When Is an Attention Head a Computational Unit? Shared Circuits Despite Orthogonal Representations

Anonymous Author(s)
Affiliation
Address
email

Abstract

Mechanistic analyses of transformers often proceed head-by-head, looking for attention maps that correspond to computational roles. But when should an attention head be interpreted as a computational unit? In principle, multi-head attention need not organize computation in such a clean way. A single computational function may be distributed across heads, and a single head may participate in several computations, with invariants appearing only after OV routing and residual-stream summation. In natural language models, this is hard to resolve because the underlying features and computations are unknown, and an apparently uninterpretable head may be polysemantic, part of a distributed circuit, or a sign that the analyst chose the wrong decomposition. To get a handle on this issue, we study a controlled setting where the target computation is analytically specified: transformers trained on factored sequence processes. These processes require independent predictive updates for multiple latent factors, represented in orthogonal residual-stream subspaces. This lets us ask whether the attention circuits implementing these independent updates are themselves factorized. We find that they need not be. Per-factor updates constrain only the aggregate routed contribution across heads, leaving substantial freedom in how computation is distributed. Heads may specialize to individual factors, compose with other heads to implement one factor, or contribute polysemantically across factor boundaries. The regime that emerges depends on the generator's spectrum, the head budget, and training dynamics. We introduce *effective subspace attention*, a scalar quantity that combines attention patterns with OV routing to recover invariant subspace-level contributions when individual maps are illegible. Our results show that independent computations can be implemented by shared attention circuits, and that individual heads may look uninterpretable even when the collective routed circuit implements a precise, theoretically predicted computation.

1 Introduction

When should an attention head be interpreted as a computational unit? Mechanistic analyses often proceed head-by-head, looking for attention maps that correspond to distinct roles. But multi-head attention need not organize computation this cleanly: a single computation may be distributed across heads Jermyn et al. [2023], and a single head may contribute to multiple computations Janiak et al. [2023]. In such cases, the invariant object is not an individual attention map, but the aggregate residual-stream update produced after OV routing and summation.

In natural-language models we cannot easily tell these cases apart, because we lack ground truth for what the underlying features are and what computations the model is meant to implement. To study the question precisely, we need a setting where the functional units of the computation are known in

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

advance and are controllable. We can then ask if and how those known functions distribute across heads, and why.

We use transformers trained on *factored sequence processes*: sequences generated by independent computational units, called factors, that evolve independently. Prior work shows that transformers trained on such processes represent each factor in its own orthogonal subspace of the residual stream Shai et al. [2026]. In this setting the computational units are known (each factor), the subspaces they occupy are known (orthogonal subspaces of the residual stream), and the per-factor update that attention must implement at each layer can be derived analytically from the data-generating process. We then ask: when the computation and corresponding representations of a network decompose into separable computational units, do the attention circuits naturally factor in the same way?

We find that they do not. Even with independent computational units assigned to orthogonal subspaces of the residual stream, individual heads can be illegible while the collective routed contribution matches theory exactly. A computation downstream from an attention head simply receives the full residual stream, and whether the information present within was written by one or multiple heads is irrelevant.

Our main contributions are:

1. We derive a **predictive theory for the collective update an attention layer must write** into each independently updated subspace of the residual stream, along with the QK/OV division of labor it requires: QK supplies the temporal weighting, OV routes token-dependent contributions into the right subspace. From this we predict the minimum number of heads needed for any given configuration.
2. We show that this subspace update admits **many head-level decompositions**. Heads can be specialized, compositional, and polysemantic, all while achieving the same loss.
3. We introduce **effective subspace attention**, a quantity which decomposes the collective contribution of multiple attention heads into each subspace when individual attention maps are illegible.

2 Background

2.1 Data-generating processes and belief state geometry

We consider training data generated by edge-emitting hidden Markov models (HMMs). An HMM is defined by a token alphabet $\mathcal{X}$, a set of hidden states $\mathcal{S}$, an initial distribution $\boldsymbol{\eta}^{(\varnothing)}$ over hidden states, and a collection of token-labeled transition matrices $(T^{(x)})_{x\in\mathcal{X}}$, where $T^{(x)}_{i,j} = \Pr(x, s_j \mid s_i)$ gives the joint probability of emitting token x and transitioning to state s_j given current state s_i.

An optimal Bayesian observer who knows the HMM structure but not the current hidden state maintains a *belief state* $\boldsymbol{\eta}$ over hidden states, updated upon observing each token x via:

$$\boldsymbol{\eta}' = \frac{\boldsymbol{\eta} T^{(x)}}{\boldsymbol{\eta} T^{(x)} \mathbf{1}}\,, \tag{1}$$

Starting from an initial state $\boldsymbol{\eta}^{(\varnothing)}$, the belief state after observing context $x_{1:\ell}$ is:

$$\boldsymbol{\eta}^{(x_{1:\ell})} = \frac{\boldsymbol{\eta}^{(\varnothing)} T^{(x_1)} \cdots T^{(x_\ell)}}{\boldsymbol{\eta}^{(\varnothing)} T^{(x_1)} \cdots T^{(x_\ell)} \mathbf{1}}\,. \tag{2}$$

For stationary processes, the optimal initial belief state is the stationary distribution $\boldsymbol{\eta}^{(\varnothing)} = \boldsymbol{\pi}$, the left eigenvector of $T = \sum_x T^{(x)}$ associated with eigenvalue 1.

Prior work shows that transformers trained on next-token prediction linearly represent the belief states induced by the data-generating process in the residual stream [Shai et al., 2024, Riechers et al., 2025].

Factored representations The Factored World Hypothesis [Shai et al., 2026] formalizes generators with independently evolving latent factors, and shows that transformers represent the corresponding per-factor beliefs in approximately orthogonal residual-stream subspaces. We call a process *independent* if each token-labeled transition operator factorizes as $T^{(x)} = \bigotimes_{n=1}^{N} T_n^{(x)}$, where factor n

has hidden state space $\mathcal{S}_n$ and transition dynamics $T_n^{(x)}$. The observed token x is determined by the joint output of latent subtokens $z^{(n)}$, though this decomposition is not exposed in the training data. Under independence, product beliefs remain product beliefs under Bayesian updating: if $\boldsymbol{\eta}^{(x_{1:\ell})} = \bigotimes_n \boldsymbol{\eta}_n^{(x_{1:\ell})}$, then $\boldsymbol{\eta}^{(x_{1:\ell+1})} = \bigotimes_n \boldsymbol{\eta}_n^{(x_{1:\ell+1})}$. Thus, the joint belief can be represented losslessly as per-factor beliefs with factor n occupying its own $(d_n - 1)$-dimensional subspace, where $d_n = |\mathcal{S}_n|$.

2.2 An architectural division of labor

The controlled setting we work in has a structurally manifest decomposition in the data generator itself, and that decomposition maps cleanly onto distinct components of the attention architecture. This lets us state up front what the architecture *must* implement, and which parts of the architecture are capable of implementing it. We describe the intuition here.

For a factored generator, each factor transition matrix $T_n^{(x)}$ has its own independent state space, so optimal beliefs evolve independently by factor. Each update requires three pieces of information: (i) *which factor* to update, set by the generator's tensor-product structure; (ii) *how strongly* to weight each temporal offset, set by the factor spectrum; and (iii) *what token-conditional vector* to write into each factor's subspace. This maps directly onto attention: QK chooses the temporal weights, while OV chooses what to write into each factor subspace. This yields two predictions:

OV carries factor identity. For any attention pattern, head h writes only in the column space of $W_O^{(h)} W_V^{(h)}$: attention reweights value vectors but cannot change their span. Thus, if the residual stream decomposes into orthogonal factor subspaces $\bigoplus_n \mathbf{V}_{F_n}$ [Shai et al., 2026], the OV maps must route contributions into these subspaces. Factor identity is therefore enforced by OV, not by the attention weights.

QK reflects temporal structure, not token identity. In our toy model, each source contribution decomposes into a token-dependent direction and a factor-specific scalar ζ_n^{d-s}. Since the direction is routed by OV, QK is left to implement only the scalar: a lag-dependent weight from source s to destination d. Thus QK should be primarily offset-shaped rather than content-routed, with different factors inducing different lag profiles.

These predictions constrain the *aggregate* layer-level circuit, not individual heads. They specify the QK/OV division of labor, but leave open how a fixed head budget distributes that work: heads may specialize, collaborate on one factor, or contribute polysemantically across factors. Thus orthogonal factor representations need not imply specialized heads. The remainder of this section makes this mechanism explicit.

2.3 Constrained belief updates

Optimal prediction requires sequential Bayesian updating: each new token refines the current predictive vector, which itself reflects all previous tokens (Eq. (1)). But attention is fundamentally parallel — at each position, it computes a weighted sum of value vectors from all source positions simultaneously. This creates a tension: how can a parallel mechanism approximate an inherently sequential computation?

The key insight of Piotrowski et al. [2025] is that attention resolves this by treating each source token as *independent evidence*. The BOS token at position 0 provides the prior $\boldsymbol{\pi}$ through the OV circuit — the baseline prediction before any evidence is observed. Each subsequent token x_s at offset $k = d - s$ then contributes an independent correction: "given the prior, how does observing x_s exactly k steps ago change my current predictions?" This correction is the geometric embedding of the difference between the prior $\Pr(S_d) = \boldsymbol{\pi}$ and the pointwise update from the observed token $\Pr(S_d | X_s = x_s) = \boldsymbol{\pi} T^{|x_s} T^{d-s}$. $T^{|x_s}$ captures the immediate evidence from the token and T^{d-s} propagates that evidence forward through $d - s$ steps of latent dynamics. Summing the prior from BOS with independent corrections from all source tokens gives the *constrained belief update*:

$$\mathbf{r}_1^{(x_{1:d})} = \underbrace{\boldsymbol{\pi}}_{\text{from BOS}} + \sum_{s=1}^{d} \underbrace{\left(\boldsymbol{\pi}\, T^{|x_s}\, T^{d-s} - \boldsymbol{\pi} \right)}_{\text{correction from source } s}. \tag{3}$$

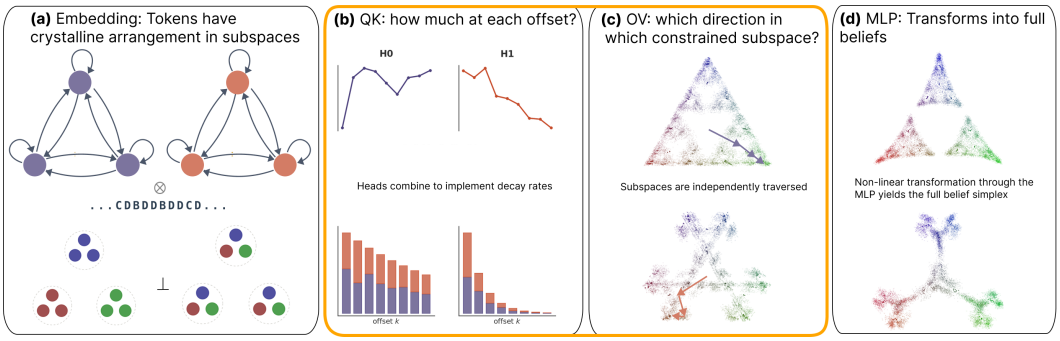

Figure 1: **Complete mechanistic description of a transformer trained on a factored process.** **(a)** The token embedding reflects the crystalline structure of the data generator: tokens arrange in factor-specific clusters within orthogonal subspaces of the residual stream. **(b)** The QK circuit determines how much each source token contributes at each offset, with heads combining to implement the temporal decay rates predicted by the spectral decomposition. **(c)** The OV circuit determines the direction and subspace of each contribution: displacement vectors within each factor's constrained predictive geometry point toward vertices of the belief simplex, with each factor's subspace traversed independently. **(d)** The MLP applies a nonlinear correction, transforming the constrained predictive geometry (intermediate fractal) into the full predictive geometry. In this work, we focus on the attention circuit (**b–c**, highlighted in orange): how the QK and OV circuits jointly implement belief updates across a factored world, the degrees of freedom in how this computation is distributed across heads, and what can be recovered when individual attention maps are not directly interpretable.

This is the best approximation to Bayesian updating achievable by summing independent pairwise contributions — precisely the functional form that attention can implement.

The spectral decomposition of the transition matrix T reveals a clean separation of roles. In our case, each factor is represented by a "Mess3" generator which has a single nontrivial decaying eigenvalue ζ of T. In this case, the corrections can be re-written through spectral decomposition as

$$\Delta \mathbf{r}_1^{(x_{1:d})} = \sum_{s=1}^{d} \zeta^{d-s} \, \boldsymbol{\pi} \, T^{|x_s} \, P_\zeta \, , \tag{4}$$

where $P_\zeta T = T P_\zeta = \zeta P_\zeta$ (see Appendix I.1 for additional details).

Equation (4) realizes the division of labor above: lag appears only in the scalar ζ^{d-s}, while token identity appears only in the displacement $\boldsymbol{\pi} T^{|x_s} P_\zeta$. The current-token term ($s = d$, $k = 0$) is the only exception, since it can also enter through the residual stream via the token embedding; see App. D.

Piotrowski et al. [2025] confirmed this correspondence in transformers trained on the Mess3 process (Appendix I) and further showed that when the eigenvalue ζ is negative, the alternating pattern ζ^{d-s} cannot be implemented by a single attention head, whose weights must be non-negative.

2.4 Per-factor constrained belief updates

For a factored process with transition operators $T^{(x)} = \bigotimes_{n=1}^{N} T_n^{(x)}$, each Mess3 factor n has its own transition matrix $T_n = \sum_{z^{(n)}} T_n^{(z^{(n)})}$, stationary distribution $\boldsymbol{\pi}_n$, and eigenvalue ζ_n. Since the factors are represented in orthogonal subspaces, the constrained belief update (Eq. (3)) decomposes into independent per-factor updates:

$$\Delta \mathbf{r}_n^{(x_{1:d})} = \sum_{s=1}^{d} \zeta_n^{d-s} \cdot \mathbf{g}_n\left(z_s^{(n)}\right) \, , \tag{5}$$

where $z_s^{(n)}$ is the sub-token for factor n at position s and $\mathbf{g}_n(z) = \boldsymbol{\pi}_n T_n^{|z} P_{\zeta_n, n}$ is a token-dependent displacement direction that is independent of offset. For a given destination d, each correction term depends only on factor n's transition dynamics and the corresponding sub-token at source s.

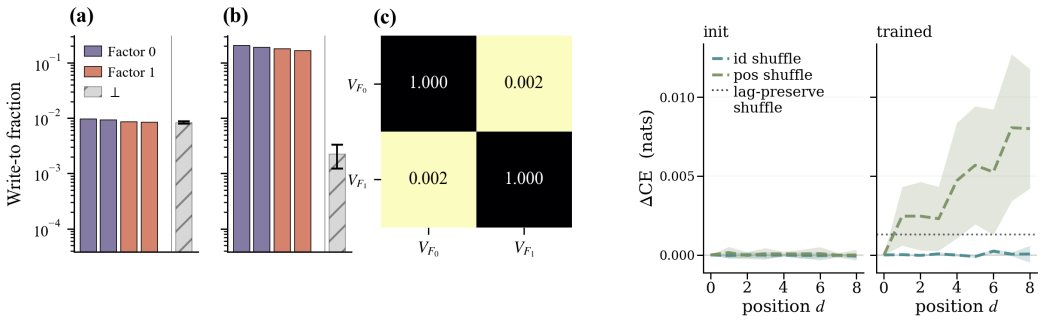

Figure 2: **OV routes factor-specific information into subspaces while QK specializes to temporal over semantic information. Left:** OV-circuit energy projected into the residual-stream subspaces associated with each factor, comparing **(a)** early training with the **(b)** end of training. Note that the bar for the perpendicular component represents an average over the remaining $d_{\text{model}} - 4$ dimensions not accounted for by $\mathbf{V}_{F_0} \oplus \mathbf{V}_{F_1}$. **(c)**: The two factor-associated subspaces are each orthogonal to one another. This is averaged over 8 different models and 2 different heads per model. **Right:** holding all else fixed, we perform causal interventions on the QK-circuit and measure the impact on cross-entropy loss (ΔCE). Interventions include shuffling token identity and position information of the embeddings entering the circuit, as well as shuffling attention weights within lag bins.

3 From specialized heads to aggregate computation

All experiments use single-layer pre-norm decoder-only transformers trained on two-factor Mess3 sequences; full hyperparameters are in Appendix L.1.

3.1 When heads are interpretable: distinct positive eigenvalues

We first train a two-head transformer on a two-factor Mess3 process with eigenvalues $\zeta_0 = 0.7$ and $\zeta_1 = 0.4$. The natural ansatz is head specialization: head n implements the temporal decay $A^{(n)}(d, s) \propto \zeta_n^{d-s}$ through QK, independent of subtoken identity and absolute position, while OV routes its output into factor subspace $\mathbf{V}_{F_n}$.

First, we use a behavioral method to find $\mathbf{V}_{F_n} \subset \mathbb{R}^{d_{\text{model}}}$, 2-d orthogonal subspaces that correspond to each factor. Once identified, we can define a write-to fraction that measures how much of the image of the OV circuit lands inside or outside of each subspace (see Appendix E for a detailed description of finding and evaluating these subspaces). Figure 2 shows that OV matrices initially write no more to the factor subspaces $\mathbf{V}_{F_n}$ than to other directions, but by the end of training place most of their write-to energy in $\mathbf{V}_{F_0} \oplus \mathbf{V}_{F_1}$, a 4-dimensional subspace of the 120-dimensional residual stream. Thus OV learns to route updates into the orthogonal factor subspaces. Separating by head shows near-specialization. Recovering the intermediate activation geometry in Figure 13 also suggests strong alignment between a specific head and a factor's belief states. However, it is not exact specialization: in Figure 3, 11% of one factor's write-to energy comes from the head primarily associated with the other factor. This leakage is consistent with the training trajectory. Early in training, both heads reduce loss on the slower-decaying factor (largest eigenvalue), which has the larger available loss reduction; only later does one head re-specialize to the faster-decaying factor. This kind of staged learning is consistent with prior work on learning dynamics in neural networks and transformers [Kunin et al., 2025, Saxe et al., 2014, Yüksel et al., 2026]. Thus, even in a setting where a specialized solution is available, gradient descent can transiently pass through collaborative decompositions.

QK shows the complementary structure predicted by the theory. The rightmost panel in Figure 2 shows token-shuffle interventions that indicate that attention is primarily lag dependent: loss increases when attention weights are shuffled across source positions, but not when shuffled only among tokens at the same position. When the position shuffles are constrained to preserve lag, the effect is significantly reduced. This indicates the QK circuits' sensitivity to the position shuffle is

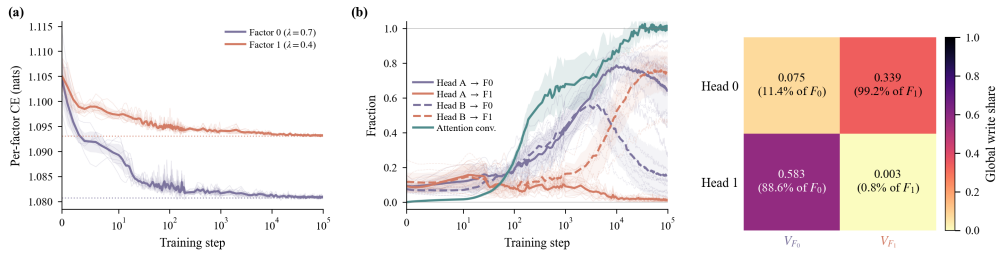

Figure 3: We measure how much the loss is reduced per-factor during training **(a)**, and for each factor, how much of each head's energy is being written into that factor as it progresses towards its final attention maps **(b)**. For $\zeta_0 = 0.7$ and $\zeta_1 = 0.4$, during training, we see that the model prefers reducing loss for the factor with the larger eigenvalue (as there is more loss to reduce). Initially, this admits a collaborative solution where both heads focus on factor 0. Eventually, the second head re-specializes to factor 1. Plots are averaged across 8 heads, and heads are aligned such that head A is the one primarily responsible for factor 0 (and vice versa). In **(right)** we examine a single seed and see that the heads are largely specialized, but there is some leakage from head 0 into factor 0, explained by the fact that both heads initially focused on that factor.

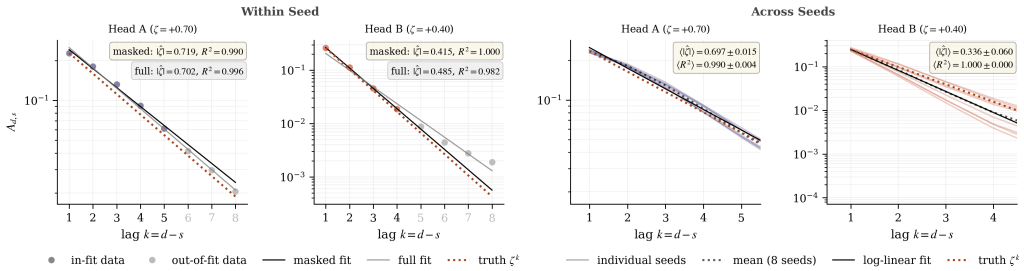

Figure 4: **Per-head attention patterns can be interpretable on their own.** In this case, for two factors with positive but different magnitude eigenvalues ($\zeta_0 = 0.70$ and $\zeta_1 = 0.40$), the decay rates of the attention profiles of each head uniquely match the temporal dynamics of one of the two factors – indicating we are in a regime of head *specialization*.

largely due to a sensitivity to lag and not the absolute position. Figure 4 shows attention weights binned by lag $k = d - s$. Since the $k = 0$ term shares a degree of freedom with the residual skip connection, we fit only off-diagonal attention weights; see Appendix D. The $\zeta_0 = 0.7$ head matches the predicted exponential decay, while the $\zeta_1 = 0.4$ head drifts at large lags, where the required corrections are small. Masking attention beyond lag τ confirms this: the slower-decaying factor remains sensitive up to $\tau = 5$, while the faster-decaying factor is sensitive only up to $\tau = 4$. We determine these thresholds using a noise floor. Lags whose ablation changes loss by less than 10^{-5}nats are deemed irrelevant. See Appendix F for details. Fitting only behaviorally relevant lags yields good agreement for both heads.

This example cleanly separates OV routing from QK temporal weighting, but also shows why individual heads are not the invariant unit of analysis. The model is constrained only by the aggregate routed update written into the residual stream, not by which head supplies it. This motivates effective *subspace* attention.

3.2 Effective subspace attention

To extract the invariant factor-level computation, let f_n map the residual-stream contribution onto the belief update for factor n in Eq. (5). We define the *effective subspace attention*

$$\boxed{\alpha_n(d,s) = \frac{\left\langle f_n\left(\sum_h A^{(h)}_{d,s}\mathbf{v}^{(h)}_s\right), \mathbf{g}_n(z^{(n)}_s)\right\rangle}{\left\|\mathbf{g}_n(z^{(n)}_s)\right\|^2}}. \tag{6}$$

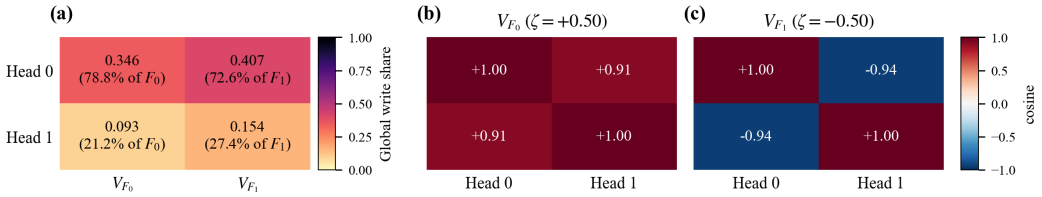

Figure 5: **In the mixed-sign case, heads are not specialized by write magnitude.** In **(a)** both heads write substantially into both factor subspaces. In **(b)** the heads write in nearly the same direction in the positive-eigenvalue subspace, but anti-parallel in the negative-eigenvalue subspace **(c)**, providing the signed OV-coupling needed for alternating decay.

This projects the summed routed contribution from source s to destination d onto the token-dependent displacement direction for factor n. Under the constrained belief-update theory,

$$\alpha_n(d, s) = \zeta_n^{d-s}.$$

By linearity, α_n decomposes across heads as $\alpha_n(d, s) = \sum_h \alpha_n^{(h)}(d, s)$, so the theory constrains only the sum:

$$\sum_h \alpha_n^{(h)}(d, s) = \zeta_n^{d-s}. \tag{7}$$

Specialization is the case where one head supplies this quantity for a factor. Collaboration is the case where several heads sum to it. Polysemanticity is the case where a head contributes to multiple factors. We quantify these regimes with the coupling matrix induced by per-head effective subspace attention: polysemy measures spread across subspaces per head, and collaboration measures spread across heads per subspace; see Appendix C. Effective subspace attention lets us distinguish head-level structure from the invariant computation implemented by the full attention layer.

3.3 When heads are not interpretable: positive/negative eigenvalues.

We apply effective subspace attention to a two-head transformer trained on a factored process with eigenvalues $\zeta_0 = 0.5$ and $\zeta_1 = -0.5$. Piotrowski et al. [2025] showed that a negative eigenvalue requires two heads: because attention weights are nonnegative, a single head cannot implement the sign-alternating decay $(-0.5)^{d-s}$. In this setting, we find that the two heads collaborate to implement the negative-eigenvalue factor, but also collaborate on the positive-eigenvalue factor.

Let us begin by validating our predictions on the OV circuit. Just like the pattern shown in Figure 2, we find that the subspaces $\mathbf{V}_{F_0}$ and $\mathbf{V}_{F_1}$ are *nearly* orthogonal to each other and carry the lion's share of variation in the residual stream of the trained network. Unlike the previous case, however, we find no evidence of head specialization when looking at the write-to fraction. Figure 5 shows that instead, both heads contribute to both factors in approximate proportion to their total write-to magnitude. However, the *direction* in which they write differs. We establish this by looking at the cosine between the two heads' write operators into $\mathbf{V}_{F_n}$, computed in matrix-Frobenius geometry (details can be found in Appendix E).

The right panel in Figure 5 reveals that while the global write share doesn't distinguish factor 0 and 1, the signed value does. The two heads write to $\mathbf{V}_{F_0}$ in nearly the same direction, but they are almost exactly anti-parallel in how they write to $\mathbf{V}_{F_1}$, providing the mechanism to fit the alternating decay rate. The top panel of Figure 6 shows a naive attempt at fitting a decay rate for each head individually for a trained model. We can see each head implements an alternating attention pattern, with one head specializing to odd k and one specializing to even k. Fits to the expected exponential decays are very poor when we look only at a single head, but the effective subspace attention allows contributions from both heads, and recovers the predicted monotone decay for ζ_0 and the predicted alternating decay for ζ_1. Recovering the geometry associated with each factor in Figure 14 also reveals that no single head is sufficient to represent the full geometry; both are required.

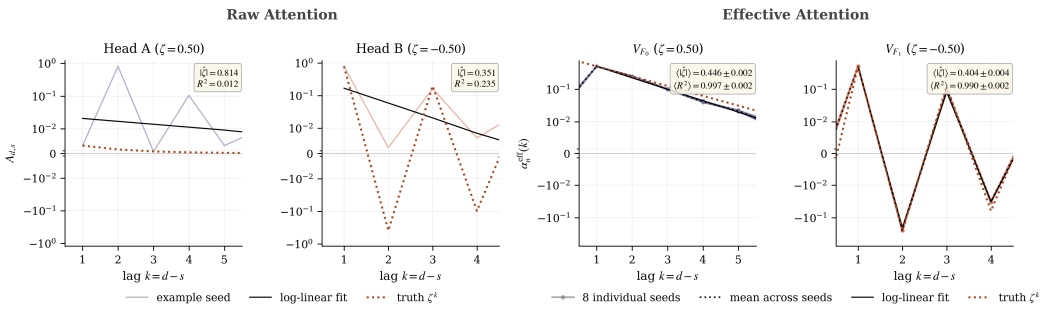

Figure 6: **Raw per-head attention profiles fail to recover the ground truth dynamics in the mixed-sign case, while effective subspace attention succeeds. Left**: neither individual head matches the predicted decay for either factor. **Right**: considering the relevant subspaces routed to by the OV vectors lets us recover the monotone positive decay and the alternating negative decay.

3.4 Minimum architecture

For a factored process, an attention head supplies a non-negative lag-dependent attention pattern, while its OV circuit determines how that pattern is signed and routed into factor subspaces. Thus the relevant architectural resource is not one head per eigenvalue or one head per factor, but the number of non-negative temporal rays needed to express the required signed lag-dependent updates after OV coupling. We refer to this as a *conic decomposition* of the update geometry.

This distinction matters in mixed-sign settings. A negative eigenvalue cannot be implemented by a single head, since attention weights are non-negative; two non-negative rays are needed to compose the alternating signed update. However, the same two non-negative rays can also be coupled with the same sign through OV to produce a monotone positive decay. Therefore, when a positive eigenvalue is present, it can sometimes be implemented "for free" by reusing the rays already required for the negative mode, reducing the head count below a naive per-eigenmode assignment. Appendix J gives the full conic argument. Table 1 lists the resulting predictions, which we test in the next section.

Table 1: Minimum head count under different two-eigenvalue cases.

Configuration	Values	Identifier	$H_{\min}$	$H_{\min}$ (modes)
$+\zeta_1, +\zeta_2,\ \lvert\zeta_1\rvert \neq \lvert\zeta_2\rvert$	+0.7, +0.4	$D^{+}_{h=2}$	2	2
$+\zeta_1, +\zeta_1$	+0.7, +0.7	$I^{+}_{h=1}$	1	1
$+\zeta_1, -\zeta_1$	+0.5, -0.5	$S^{\pm}_{h=2}$	2	2
$+\zeta_1, -\zeta_2,\ \lvert\zeta_1\rvert \neq \lvert\zeta_2\rvert$	+0.7, -0.5	$D^{\pm}_{h=2}$ / $D^{\pm}_{h=3}$	2	3
$-\zeta_1, -\zeta_1$	-0.5, -0.5	$I^{-}_{h=2}$	2	2

4 Effective attention recovers the invariant computation

Table 1 outlines the 5 different configurations of 2-factor processes we generate sequences from. Note that we omit the case of $-\zeta_1, -\zeta_2$ with $|\zeta_1| \neq |\zeta_2|$. This is because the narrow range of negative values for Mess3, $[-0.5, 0)$, means that the eigenvalues will either be so close together that the process can be approximated with fewer heads, or that one of the values will be close to 0 and therefore almost IID. The identifiers will be used in subsequent figures.

For each configuration in Table 1, we train models with $n_{\text{heads}} \in \{1, \ldots, 6\}$ across eight random seeds. Figure 7 summarizes the central empirical result: the conic head-count prediction matches the empirical loss plateau, while effective subspace attention recovers the predicted factor-level updates even when raw attention maps are not individually interpretable. First, the minimum head count predicted by our conic framework aligns with the point at which loss plateaus: adding heads below this threshold substantially improves performance, while additional heads provide little benefit. Second, raw attention maps are not reliable indicators of whether the correct computation has been learned. In several configurations, individual heads do not match any predicted decay pattern. However,

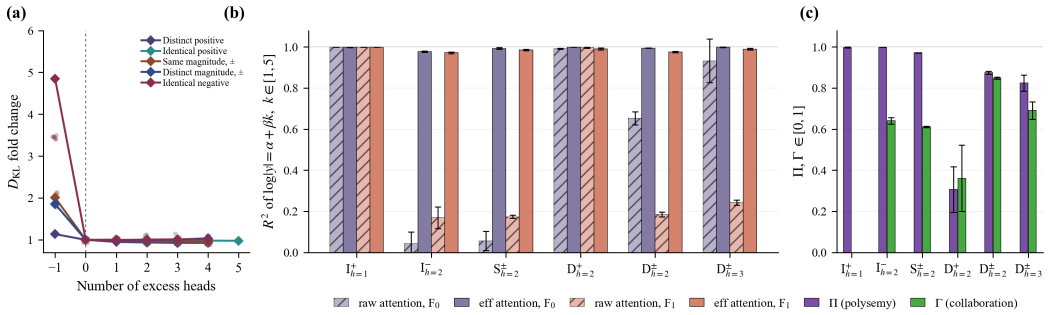

Figure 7: **(a)** D_{KL} versus the number of excess heads evaluated on 10,240 sequences. The conic minimum head count aligns with the empirical loss plateau across all five configurations. **(b)** Raw attention maps are not reliable indicators of whether the correct factor-level computation has been learned: some heads match predicted decay patterns, but many do not. Effective subspace attention, which incorporates OV routing and sums across heads, consistently recovers the predicted decay rates. **(c)** Polysemy and collaboration are common whenever multiple heads are available: heads often contribute to multiple factors, and factors are often implemented by multiple heads. Thus the invariant computation is generally the routed aggregate update, not an individual attention head.

effective subspace attention, which incorporates OV routing and sums contributions across heads, consistently recovers the theoretically predicted factor-level decay rates. Finally, polysemy and collaboration show that distributed implementations are common. Except in the single-head case, heads often contribute to multiple factors and factors are often implemented by multiple heads. Thus, the invariant computation is recovered at the routed aggregate level, not necessarily at the level of individual attention heads. In appendix C, we see that as we make more heads available than is needed, signatures of collaboration become even stronger.

5 Conclusion

We asked when an attention head should be treated as a computational unit. In our setting, the answer is: the computational unit is at the level of the routed aggregate, not the individual head. The data-generating spectrum sets a minimum head count that trained models reliably reach across all five configurations we tested. Above that floor, the same predicted update admits many head-level decompositions that loss treats as equivalent and per-head inspection cannot distinguish, regardless of whether they are specialized, collaborative, or polysemantic.

What recovers the invariant they share is effective subspace attention: it routes the post-OV, summed contribution into each factor's subspace and returns the lag-dependent scalar predicted by the constrained belief update, regardless of which heads supplied it. The unit of computation, in our setting, is the residual-stream subspace into which a layer collectively writes rather than the heads that write into it. This opens a question for natural-data interpretability: how can we find such subspaces in scaled models, and in doing so uncover their natural units of computation?

6 Limitations and future work.

Our analysis is restricted to single-layer transformers trained on processes with known ground truth, where the computation depends entirely on lag. Extending to multi-layer architectures, richer spectral structure, and processes beyond conditional independence and exclusively temporal dynamics remains important future work. More broadly, our current theory is combinatorial in the spectrum—counting distinct magnitudes and sign-changing modes—whereas the optimization problem faced by a finite model is also metric. A natural next step is to characterize how approximation quality degrades as eigenvalues move closer together or toward zero, and to replace a binary notion of "enough heads" with a quantitative tradeoff between spectral separation, exact implementability, and loss.

References

Clémentine C. J. Dominé, Nicolas Anguita, Alexandra M. Proca, Lukas Braun, Daniel Kunin, Pedro A. M. Mediano, and Andrew M. Saxe. From lazy to rich: Exact learning dynamics in deep linear networks. 2025. URL https://arxiv.org/abs/2409.14623.

Joakim Edin, Róbert Csordás, Tuukka Ruotsalo, Zhengxuan Wu, Maria Maistro, Casper L. Christensen, Jing Huang, and Lars Maaløe. Gim: Improved interpretability for large language models, 2025. URL https://arxiv.org/abs/2505.17630.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Mario Geiger, Stefano Spigler, Arthur Jacot, and Matthieu Wyart. Disentangling feature and lazy training in deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(11):113301, November 2020. ISSN 1742-5468. doi: 10.1088/1742-5468/abc4de. URL http://dx.doi.org/10.1088/1742-5468/abc4de.

Michael Hanna, Ollie Liu, and Alexandre Variengien. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=p4PckNQR8k.

Sarthak Jain and Byron C. Wallace. Attention is not explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, 2019. URL https://aclanthology.org/N19-1357/.

Jett Janiak, Chris Mathwin, and Stefan Heimersheim. Polysemantic attention head in a 4-layer transformer. *AI Alignment Forum*, 2023.

Adam Jermyn, Chris Olah, and Tom Henighan. Attention head superposition. *Transformer Circuits Thread, Circuits Updates – May 2023*, 2023.

Harish Kamath, Emmanuel Ameisen, Isaac Kauvar, Rodrigo Luger, Wes Gurnee, Adam Pearce, Sam Zimmerman, Joshua Batson, Thomas Conerly, Chris Olah, and Jack Lindsey. Tracing attention computation through feature interactions. *Transformer Circuits Thread*, 2025.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms. 2020. URL https://arxiv.org/abs/2004.10102.

Daniel Kunin, Giovanni Luca Marchetti, Feng Chen, Dhruva Karkada, James B Simon, Michael R DeWeese, Surya Ganguli, and Nina Miolane. Alternating gradient flows: A theory of feature learning in two-layer neural networks. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=t7LKc0MMW6.

Maximilian Li and Lucas Janson. Optimal ablation for interpretability. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=opt72TYzwZ.

S. E. Marzen and J. P. Crutchfield. Nearly maximally predictive features and their dimensions. *Phys. Rev. E*, 95(5):051301(R), 2017. doi: 10.1103/PhysRevE.95.051301. SFI Working Paper 17-02-007; arxiv.org:1702.08565 [cond-mat.stat-mech].

Callum Stuart McDougall, Arthur Conmy, Cody Rushing, Thomas McGrath, and Neel Nanda. Copy suppression: Comprehensively understanding a motif in language model attention heads. In *The 7th BlackboxNLP Workshop*, 2024. URL https://openreview.net/forum?id=5Hd6813x3U.

Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The hydra effect: Emergent self-repair in language model computations. 2023. URL https://arxiv.org/abs/2307.15771.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/2c601ad9d2ff9bc8b282670cdd54f69f-Paper.pdf.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=9XFSbDPmdW.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Mateusz Piotrowski, Paul M Riechers, Daniel Filan, and Adam S Shai. Constrained belief updates explain geometric structures in transformer representations. *arXiv preprint arXiv:2502.01954*, 2025.

Paul M Riechers, Thomas J Elliott, and Adam S Shai. Neural networks leverage nominally quantum and post-quantum representations. *arXiv:2507.07432*, 2025.

Cody Rushing and Neel Nanda. Explorations of self-repair in language models. 2024. URL https://arxiv.org/abs/2402.15390.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. 2014. URL https://arxiv.org/abs/1312.6120.

Adam Shai, Loren Amdahl-Culleton, Casper L. Christensen, Henry R. Bigelow, Fernando E. Rosas, Alexander B. Boyd, Eric A. Alt, Kyle J. Ray, and Paul M. Riechers. Transformers learn factored representations. *arXiv:2602.02385, ICML*, 2026.

Adam S Shai, Sarah E Marzen, Lucas Teixeira, Alexander Gietelink Oldenziel, and Paul M Riechers. Transformers represent belief state geometry in their residual stream. *NeurIPS*, 2024.

Aaditya K Singh, Ted Moskovitz, Felix Hill, Stephanie C.Y. Chan, and Andrew M Saxe. What needs to go right for an induction head? a mechanistic study of in-context learning circuits and their formation. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=O8rrXl71D5.

Eric Todd, Millicent Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AwyxtyMwaG.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, 2019.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=NpsVSN6o4ul.

Sarah Wiegreffe and Yuval Pinter. Attention is not not Explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20, 2019. URL https://aclanthology.org/D19-1002.

Keith Wynroe and Lee Sharkey. Decomposing the qk circuit with bilinear sparse dictionary learning. *AI Alignment Forum*, 2024.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, 2016. URL https://aclanthology.org/N16-1174/.

Oğuz Kaan Yüksel, Rodrigo Alvarez Lucendo, and Nicolas Flammarion. Incremental learning of sparse attention patterns in transformers. 2026. URL https://arxiv.org/abs/2602.19143.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Hf17y6u9BC.

A Related Work

A.1 Interpretability of Attention

How interpretability relates to attention depends strongly on choosing *what* attention is supposed to explain. Early work often drew equivalences between attention weights and token-level explanations: if a token was attended strongly to, that token was considered important for prediction Yang et al. [2016]. This interpretation has been heavily contested. Jain and Wallace [2019] argue that attention weights often do not correlate with other measures of feature importance, and that alternative attention distributions can often produce similar outputs. Wiegreffe and Pinter [2019] respond that attention *can* still be explanatory under some definitions, but only when the relationship between attention and the rest of the model is respected. A reasonable conclusion is that attention maps are useful diagnostic objects, but not faithful explanations by default, and that interpretation should take the rest of the model into account.

Mechanistic interpretability shifts the question away from focusing on individual tokens to instead whether attention heads implement specific, identifiable computations. This is a more complete framing because a head is not just a pattern: its behavior depends also on value vectors, output projections, and downstream effects in the residual stream [Elhage et al., 2021, Kobayashi et al., 2020]. Work on induction heads Olsson et al. [2022], the IOI circuit Wang et al. [2023], greater-than circuits Hanna et al. [2023], modular addition Nanda et al. [2023], copy suppression McDougall et al. [2024], and function vectors Todd et al. [2024] shows that some heads participate in relatively clean mechanisms. In these cases, heads are interpretable not as isolated components but as participants in larger circuits. Individual heads can be polysemantic, exhibiting multiple task-relevant attention patterns in different contexts Janiak et al. [2023]. More generally, attentional features may be represented in superposition across heads rather than localized to a single head Jermyn et al. [2023]. Recent work therefore decomposes attention computation at a finer granularity, either by tracing interactions between attention features Kamath et al. [2025] or by decomposing the QK circuit into sparse query-key feature pairs Wynroe and Sharkey [2024]. This suggests that heads are useful units of analysis, but not always the most fundamental units of explanation.

However, even successful circuit analyses complicate the claim that heads are clean. Many heads can be pruned with little effect, suggesting redundancy [Michel et al., 2019, Li and Janson, 2024, Voita et al., 2019]. In more detailed analyses, heads often collaborate, compensate, or inhibit one another Zhang and Nanda [2024]. The IOI circuit, for example, includes name-mover heads, backup heads, and negative heads, rather than a single decisive component. Similarly, work on self-repair [Rushing and Nanda, 2024, McGrath et al., 2023, Edin et al., 2025] shows that ablating one head can change the behavior of other components which then compensate, and research within training dynamics suggests that a transformer may consist of many such circuits Singh et al. [2024].

The emerging view is therefore mixed. Attention weights alone are weak explanations, and individual heads should not be presumed to correspond to clean, independent mechanisms. However, attention heads can still be meaningful units when the full circuit is accounted for.

B Effective Attention

The constrained belief update for factor n at destination position d is:

$$\mathbf{r}_n^{(z_{1:d})} = \boldsymbol{\pi}_n + \sum_{s=1}^{d} \left(\boldsymbol{\pi}_n T_n^{|z_s^{(n)}} T_n^{d-s} - \boldsymbol{\pi}_n \right) . \tag{8}$$

Each displacement can be written as:

$$\boldsymbol{\pi}_n T_n^{|z_s^{(n)}} T_n^{d-s} - \boldsymbol{\pi}_n = \zeta_n^{d-s} \cdot \mathbf{g}_n\left(z_s^{(n)}\right) , \tag{9}$$

where $\mathbf{g}_n(z) = \left(\boldsymbol{\pi}_n T_n^{|z} - \boldsymbol{\pi}_n \right) P_{\zeta_n, n}$ is the token-dependent displacement direction (independent of offset), and ζ_n is the non-trivial eigenvalue of T_n. The eigenvalue decay ζ_n^{d-s} is the scalar we wish to recover.

The displacement map f_n maps the summed per-source OV contribution into factor n's constrained belief update. For a given (destination d, source s) pair:

$$f_n\left(\sum_h A_{d,s}^{(h)} \, \mathbf{v}_s^{(h)} \right) \approx \zeta_n^{d-s} \cdot \mathbf{g}_n\left(z_s^{(n)}\right) . \tag{10}$$

To isolate the decay, we project onto the known $k{=}0$ displacement direction $\mathbf{g}_n\left(z_s^{(n)}\right)$, which corresponds to offset zero where $\zeta_n^0 = 1$:

$$\alpha_n(d, s) \;=\; \frac{\left\langle f_n\left(\sum_h A_{d,s}^{(h)} \, \mathbf{v}_s^{(h)} \right),\; \mathbf{g}_n\left(z_s^{(n)}\right) \right\rangle}{\left\| \mathbf{g}_n\left(z_s^{(n)}\right) \right\|^2} . \tag{11}$$

This is a scalar-valued function of (d, s). Under the constrained belief update theory:

$$\alpha_n(d, s) = \zeta_n^{d-s} . \tag{12}$$

The denominator normalizes out the token-dependent magnitude, leaving only the offset-dependent decay. This is well-defined whenever $\mathbf{g}_n\left(z_s^{(n)}\right) \neq \mathbf{0}$, which holds for any token z that is informative about factor n.

By linearity of f_n, the effective attention decomposes additively across heads:

$$\alpha_n(d, s) = \sum_h \alpha_n^{(h)}(d, s) , \tag{13}$$

where

$$\alpha_n^{(h)}(d, s) = \frac{\left\langle f_n\left(A_{d,s}^{(h)} \, \mathbf{v}_s^{(h)} \right),\; \mathbf{g}_n\left(z_s^{(n)}\right) \right\rangle}{\left\| \mathbf{g}_n\left(z_s^{(n)}\right) \right\|^2} . \tag{14}$$

Under head specialization, $\alpha_n^{(h)}(d, s) \approx \zeta_n^{d-s}$ for exactly one head h and ≈ 0 for all others. In the compositional specialized case, multiple heads may contribute to the same factor while remaining factor-specific. Under polysemantic attention, no individual $\alpha_n^{(h)}$ matches ζ_n^{d-s}, but their sum does.

C Measuring polysemy and collaboration

The per-head effective attention (14) also defines a natural coupling matrix $C \in \mathbb{R}^{H \times N}$ that can be used to quantify how factor information is distributed across heads:

$$C_{h,n} \;\propto\; \mathbb{E}_{z,\,k}\left[\left| \bar{\alpha}_n^{(h)}(z^{(n)}, k) \right|^2 \right] . \tag{15}$$

where

$$\bar{\alpha}_n^{(h)}(z^{(n)}, k) = \mathbb{E}_{(d,s):d-s=k,\ z_s^{(n)}=z}\left[\alpha_n^{(h)}(d, s)\right] \tag{16}$$

The average effective attention $\bar{\alpha}_n^{(h)}(z, k)$ describes the average attention head h pays to subtoken $z^{(n)}$ at a position k steps back from the query token. The average is over positions consistent with this lag, and sequences conditioned on a particular subtoken identity at this position. Since the effective attention can be negative, the coupling matrix measures the average attention power relevant to factor n within head h. From this matrix, we can define two probability distributions:

$$P(n \mid h) = \frac{C_{hn}}{\sum_{n'} C_{hn'}} \tag{17}$$

$$P(h \mid n) = \frac{C_{hn}}{\sum_{h'} C_{h'n}} \tag{18}$$

which characterize the amount a given head is responsible for a factor and the amount a given factor is handled by a particular head, respectively. We can also define relative weightings

$$w_n = \frac{\sum_h C_{hn}}{\sum_{n'h'} C_{hn'}} \tag{19}$$

$$w_h = \frac{\sum_n C_{hn}}{\sum_{n'h'} C_{hn'}} \tag{20}$$

that quantify the overall magnitude of a given head or factor's contribution to the coupling matrix. The entropy $\mathrm{H}[P]$ (not to be confused with number of heads H) of distribution P quantifies how peaked or spread a distribution is. Using the distributions in (17)- (18), we define

$$\Pi = \frac{1}{\log N} \sum_{h=1}^{H} w_h \mathrm{H}[P(n \mid h)] \quad \text{(polysemy)} \tag{21}$$

$$\Gamma = \frac{1}{\log H} \sum_{n=1}^{N} w_n \mathrm{H}[P(h \mid n)] \quad \text{(collaboration)}\,, \tag{22}$$

The polysemy score Π quantifies the extent to which a given head attends to multiple factors on average. It is zero when heads are responsible for one factor only and one when heads attend to all factors evenly. Similarly, the collaboration score Γ quantifies the fraction of total heads responsible for a particular factor on average. It is zero when each factor is handled by a single head and one when each factor is handled evenly by all heads. Finally, we define the effective number of heads for a particular factor

$$N_{\text{eff}}(n) = \exp(\mathrm{H}[P(h \mid n)]) \tag{23}$$

and its weighted average

$$\bar{N}_{\text{eff}} = \sum_n w_n N_{\text{eff}}(n) \tag{24}$$

C.1 Π, Γ, and N_{eff} as a function of H

Figure C.1 plots the metrics Π, Γ, and N_{eff} as for a varying number of total heads.

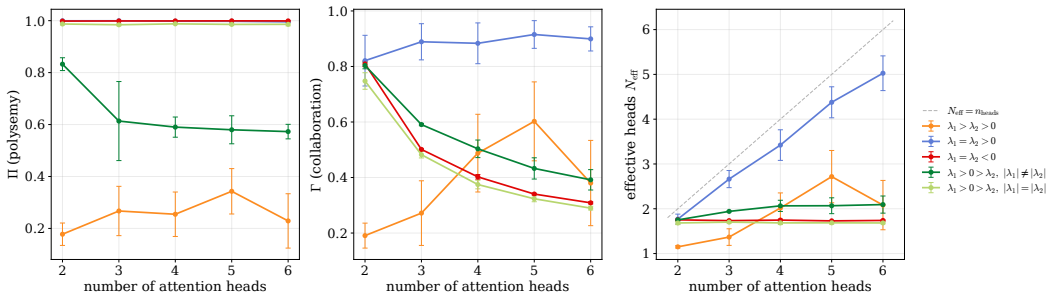

Figure 8: Polysemy Π, collaboration Γ, and effective number of heads as a function of total number of heads in the model. In each run, we fix d_{model} and vary n_{heads} which determines d_{head}. Error bars reflect the 95% confidence intervals over 8 random seeds.

D Skip-connection shares degrees of freedom with diagonal attention

The residual stream at position d after the attention layer is:

$$\mathbf{x}_{\text{mid}}^{(d)} = \underbrace{\mathbf{x}_{\text{pre}}^{(d)}}_{\text{skip connection}} + \sum_h \left[A_{d,0}^{(h)} \mathbf{v}_{\text{BOS}}^{(h)} + A_{d,d}^{(h)} \mathbf{v}_d^{(h)} + \sum_{s=1}^{d-1} A_{d,s}^{(h)} \mathbf{v}_s^{(h)} \right] . \tag{25}$$

Projecting into factor n's subspace, this must equal the constrained belief update (Eq. 4). The three terms on the right-hand side of Eq. 25 correspond to three groups of contributions:

Non-local sources ($1 \leq s < d$). These contribute the offset $k \geq 1$ displacements and enter exclusively through attention. The spectral theory applies unambiguously:

$$\sum_h A_{d,s}^{(h)} f_n(\mathbf{v}_s^{(h)}) = \zeta_n^{d-s} \cdot \mathbf{g}_n(z_s^{(n)}) . \tag{26}$$

BOS token ($s = 0$). The BOS token encodes a share of the prior $\boldsymbol{\pi}_n$ through attention to position 0:

$$\sum_h A_{d,0}^{(h)} f_n(\mathbf{v}_{\text{BOS}}^{(h)}) = \gamma_n \cdot \boldsymbol{\pi}_n \tag{27}$$

for some scalar γ_n that may vary with position d but not with token identity.

Current token ($s = d$) and skip connection. The offset $k = 0$ displacement $\mathbf{g}_n(z_d^{(n)})$ and the remaining share of the prior $(1 - \gamma_n) \cdot \boldsymbol{\pi}_n$ must be jointly delivered by two pathways:

$$f_n(\mathbf{x}_{\text{pre}}^{(d)}) + \sum_h A_{d,d}^{(h)} f_n(\mathbf{v}_d^{(h)}) = (1 - \gamma_n) \cdot \boldsymbol{\pi}_n + \mathbf{g}_n(z_d^{(n)}) . \tag{28}$$

The left-hand side has two terms but only their sum is constrained. The model is free to distribute the k=0 displacement across these two pathways without affecting the total. This is a degree of freedom that does not exist at any other offset.

D.1 Experimental test: one-hot embeddings

If the token embedding carries part of $\mathbf{g}_n(z_d^{(n)})$ through the skip connection, then removing this information pathway should force the model to compensate through diagonal attention. We test this by comparing two training conditions:

- **Learned embeddings** (standard): the embedding matrix W_E is trained jointly with all other parameters. The skip connection can carry arbitrary token-dependent information into resid_mid.

- **One-hot embeddings**: W_E is frozen as a one-hot encoding (identity matrix, zero-padded to d_{model}). The skip connection passes through a vector that identifies the token but contains no learned displacement structure. The $k{=}0$ displacement must be delivered entirely through diagonal attention.

All other parameters are trained normally in both conditions. We train 8 seeds per condition on the $\zeta_1 = 0.5,\ \zeta_2 = -0.5$ case (as we observe this setting empirically having little diagonal-attention), holding all hyperparameters fixed.

D.2 Results

Figure 9 shows the mean attention weights across lag for both conditions. With learned embeddings, $\bar{A}_{d,d}$ is lower across lag 0, but all other positions match or are higher, reflecting the degree of freedom: different seeds distribute the $k{=}0$ displacement differently between the embedding and diagonal attention. With one-hot embeddings, $\bar{A}_{d,d}$ is consistently higher.

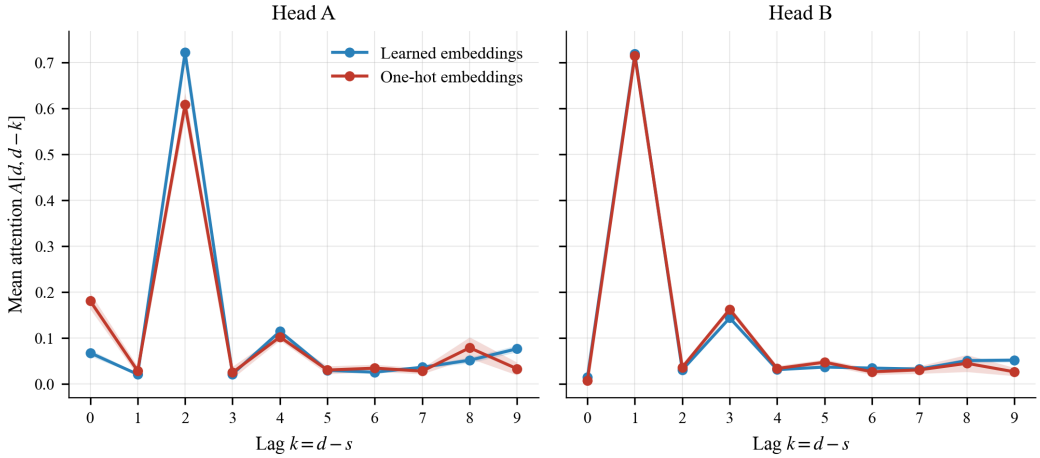

Figure 9: Mean attention-weights grouped by lag across seeds for learned vs. one-hot embeddings. Learned embeddings show lower diagonal attention across all seeds for head A, which we take to be the head implementing the $\zeta_1 = 0.5$ and the even offsets of $\zeta_2 = -0.5$. Notably, scores at all other lags are essentially an exact match. The only difference is in the diagonal attention. One-hot embeddings force higher diagonal attention because the model cannot use the embedding to store additional information. This isolates the $k = 0$ degree of freedom: the current-token displacement can be shared between the learned embedding/skip pathway and diagonal attention, while non-local attention patterns remain unchanged.

These results support Eq. 28: the skip connection and diagonal attention are interchangeable pathways for the $k{=}0$ displacement, and the model may exploit this freedom when learned embeddings are available.

E Factor Subspace Identification and Quantification

We use the vary-one routine from Shai et al. [2026] to identify subspaces.

Because our data generator is a product of two independent processes, we can sample datasets that vary one factor's subsequence while keeping the other fixed. For each factor, we fix the other to a single realization, collect activations across many realisations of the varied factor, and mean-center them per position. The top 2 principal components then form an orthonormal basis for that factor's subspace: per-position centering cancels everything that is constant across the batch leaving only the variation induced by the varied factor, and since Mess3 beliefs are 2-dimensional, two components span the belief simplex.

We define these bases as $\mathbf{V}_{F_n} \subset \mathbb{R}^{d_{\text{model}}}$. We find that these subspaces account for the vast majority of the variance in the residual stream, explaining on average 97.2% of the variation, and the subspaces

$\mathbf{V}_{F_1}$ and $\mathbf{V}_{F_2}$ are effectively orthogonal to each other, with a near-zero overlap. Once identified, we can define a write-to-fraction that measures how much of the image of the OV circuit lands inside or outside of each subspace.

Once equipped with the subspaces, $\mathbf{V}_{F_n} \subset \mathbb{R}^{d_{\text{model}}}$ we build the orthogonal-complement basis as the null space of the stacked factor bases $\mathbf{V}_{\perp} \in \mathbb{R}^{d\text{model} \times (d_{\text{model}}-4)}$ since $[\mathbf{V}_{F_0} \mid \mathbf{V}_{F_1}]$ is rank 4. Finally, we stack everything into a change-of-basis matrix:

$$\mathbf{B} = [\mathbf{V}F_0 \mid \mathbf{V}F_1 \mid \mathbf{V} \perp] \in \mathbb{R}^{d\text{model} \times d_{\text{model}}}$$

When $\mathbf{V}_{F_0}$ and $\mathbf{V}_{F_1}$ are exactly orthogonal, $\mathbf{B} \in \mathrm{O}(d_{\text{model}})$ is an orthogonal matrix and we can decompose any $x \in \mathbb{R}^{d_{\text{model}}}$ into coordinate vectors relative to those subspaces

$$y = \mathbf{B}^{\top} x = [y_0; y_1; y_{\perp}], \quad y_0 \in \mathbb{R}^2,\ y_1 \in \mathbb{R}^2, y_{\perp} \in \mathbb{R}^{d_{\text{model}}-4}\,. \tag{29}$$

In d_{model}-dim form, the orthogonal projectors are:

$$P_n = \mathbf{V}_{F_n}\mathbf{V}_{F_n}^{\top}, \quad P_{\perp} = \mathbf{V} \perp \mathbf{V} \perp^{\top} \approx I - P_0 - P_1$$

We note that the $\mathbf{V}_{F_n}$ subspaces are not completely orthogonal to each other, which is why $P_1 + P_2 + P_{\perp}$ is only approximately the identity. This allows the following approximate decomposition.

$$x \approx P_0 x + P_1 x + P_{\perp} x = \mathbf{V}_{F_0} y_0 + \mathbf{V}_{F_1} y_1 + \mathbf{V} \perp y_{\perp}$$

With these definitions in hand, we can clearly measure how much each head's OV circuit writes to different interpretable subspaces of the residual stream.

$$f_{\text{out}}(h, j) \equiv \frac{\left|\mathrm{OV}_h \mathbf{b}_j\right|_F^2}{\left|\mathrm{OV}_h\right|_F^2} \tag{30}$$

where $\mathrm{OV}_h \in \mathbb{R}^{d\text{model} \times d\text{model}}$ is head h's OV matrix and $\mathbf{b}j \in \mathbb{R}^{d\text{model}}$ is the j-th column of $\mathbf{B}$, lying in $\mathbf{V}_{F_0}$,$\mathbf{V}_{F_1}$, or $V_{\perp}$ depending on which block index j falls into. Summing per-direction within a block recovers the subspace fraction: $f_{\text{out}}(h, n) = \sum_{j \in \mathbf{V}_{F_n}} f_{\text{out}}(h, j)$ which we interpret as the percent of head h's total write-to budget that lands in factor n's subspace.

The above assumes a uniform-input distribution, but this can be replaced with an empirical average over real data that also folds in attention. We define head h's contribution to the residual stream for batch b at position d

$$r_h(b, d) = \sum_{s \le d} A_{d,s}^{(h)}(b) v_{b,s} \mathrm{W}_O^{(h)} \in \mathbb{R}^{d\text{model}} \tag{31}$$

where $v_{b,s} \in \mathbb{R}^{d_{\text{head}}}$ are the post LayerNorm inputs from the residual stream projected through $W_V^{(h)}$. From this, we can define

$$f_{\text{realized}}(h, j) = \frac{\sum_{b,d} \left(\mathbf{b}_j^{\top} r_h(b, d)\right)^2}{\sum_{b,d} |r_h(b, d)|_2^2} \tag{32}$$

and again, by summing over $j \in \mathbf{V}_{F_n}$, we can find $f_{\text{realized}}(h, n)$.

When looking for a measure that is sensitive to direction, we use $\mathbf{M}_h^{(n)} = \mathrm{OV}_h, \mathbf{V}n \in \mathbb{R}^{d\text{model}} \times 2$ which is the OV map restricted to the rank-2 output subspace of $\mathbf{V}_{F_n}$. Its squared Frobenius norm is the unnormalized write-to-$\mathbf{V}_{F_n}$ energy: $|\mathbf{M}_h^{(n)}|_F^2 = |\mathrm{OV}_h \mathbf{V}_{F_n}|F^2$ (the numerator of $f_{\text{out}}(h, n)$. We then look at the cosine between two heads' write operators, computed in matrix-Frobenius geometry:

$$s(h, h', n) = \frac{\left\langle \mathbf{M}_h^{(n)} \mathbf{M}_{h'}^{(n)} \right\rangle_F}{|\mathbf{M}_h^{(n)}|F \cdot |\mathbf{M}_{h'}^{(n)}|_F}; \tag{33}$$

where $\langle \mathbf{A}, \mathbf{B} \rangle_F$ is the the inner product of the matrices viewed as vectors in $\mathbb{R}^{d_{\text{model}} \cdot 2}$.

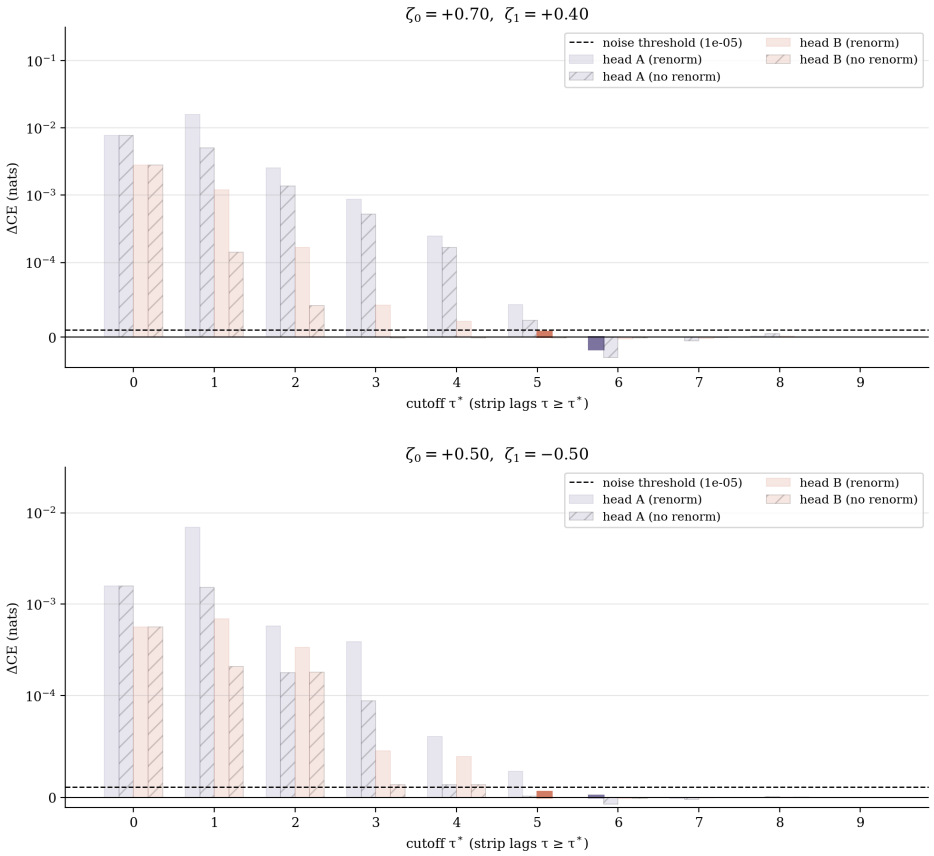

Figure 10: **The importance of attention weights to loss decay with lag.** We measure the cumulative impact on loss of ablating attention weights $A_{d,s}$ for all $\tau = d - s \geq \tau^*$. We then use this to determine the threshold τ for our attention decay fits. Bar heights are mean values over 8 model seeds.

F Additional Experiment Details

See Figs. 10 and 11 for details.

G Weight initialization and attention structure

The experiments in the main text use an initialization range of 0.02 (uniform $\mathcal{U}(-0.02, 0.02)$ for all weight matrices). Here we investigate how initialization scale affects both the final loss and the structure of the learned attention patterns.

G.1 Background: rich vs. lazy learning

The scale of weight initialization determines whether a neural network operates in the *rich* (feature learning) or *lazy* (kernel) regime Dominé et al. [2025] Geiger et al. [2020]. At small initialization, the network starts with negligible structure and the optimizer builds representations from scratch — weights move far from their initial values and the learned features bear no resemblance to the random initialization. At large initialization, the initial random structure is amplified: gradients refine rather than replace the existing weight configuration, and the network stays closer to its initialization throughout training.

For attention, this distinction has a concrete consequence. At small initialization, the QK dot products are near zero, so softmax produces *near-uniform attention patterns*. The optimizer has an easier time

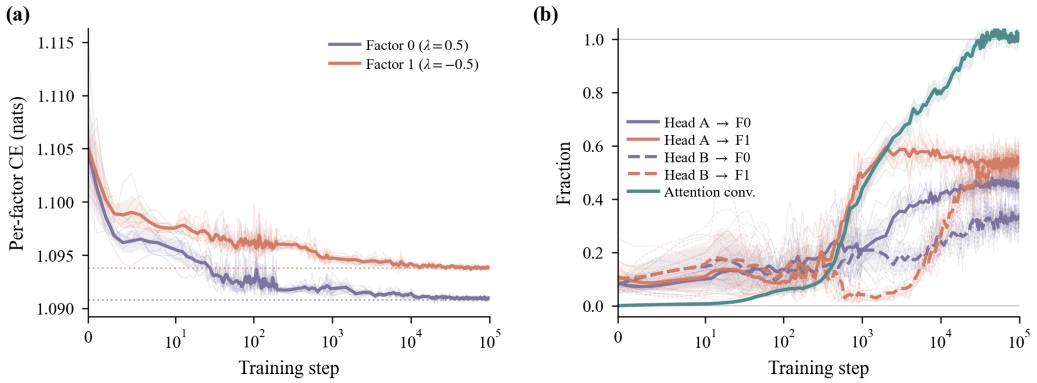

Figure 11: For $\zeta_1 = 0.5$ and $\zeta_2 = -0.5$, we see in **(a)** that loss initially reduces most for the positive factor. However, **(b)** shows that the attention heads converge to a polysemantic and collaborative solution quickly, which they must to further reduce loss for both factors.

changing these to match the spectral structure predicted by the constrained belief update framework. At large initialization, the QK dot products are large, so softmax produces *sharp, random attention patterns* from the start. The OV circuit adapts to make use of whatever temporal structure the random initialization provides, but the QK circuit — whose gradients through softmax are small when attention is already saturated — has limited ability to reshape these patterns.

G.2 Experimental setup

We compare two initialization scales across all five configurations from Table 1:

- **Small init** (init range $= 0.02$): the setting used throughout the main text.
- **Default init** (TransformerLens default): substantially larger initialization scale.

All other hyperparameters are held fixed. For each configuration and initialization scale, we train 8 models with different random seeds.

G.3 Results: loss

Table 2 compares the final KL divergence between the two initialization scales. Models trained with small initialization consistently achieve lower loss across all five configurations. The gap is most pronounced for configurations requiring compositional attention, where the QK circuit must learn structured non-monotonic patterns that the random initialization does not provide.

Table 2: Final D_{KL} (nats, mean $\pm$ std across seeds) for small vs. default initialization. H: number of heads. The ratio column shows how many times worse the large initialization is. Configurations requiring compositional attention (negative eigenvalues) are most affected. Not only is loss worse on average, the variance is much more extreme.

Configuration	H	Small init (0.02)	Default init	Ratio
Identical positive	1	0.00039 ± 0.00001	0.00045 ± 0.00001	$1.14\times$
Distinct positive	2	0.00018 ± 0.00001	0.00034 ± 0.00004	$1.96\times$
Distinct mag., $\pm$ (2H)	2	0.00030 ± 0.00001	0.00040 ± 0.00001	$1.36\times$
Distinct mag., $\pm$ (3H)	3	0.00029 ± 0.00001	0.00049 ± 0.00008	$1.69\times$
Same magnitude, $\pm$	2	0.00028 ± 0.00001	0.00091 ± 0.00026	$3.29\times$
Identical negative	2	0.00016 ± 0.00000	0.00106 ± 0.00062	$6.78\times$

G.4 Results: raw and effective attention

The loss difference is explained by the quality of the learned temporal dynamics. Figure 12 shows our predictive framework applied to higher initialization.

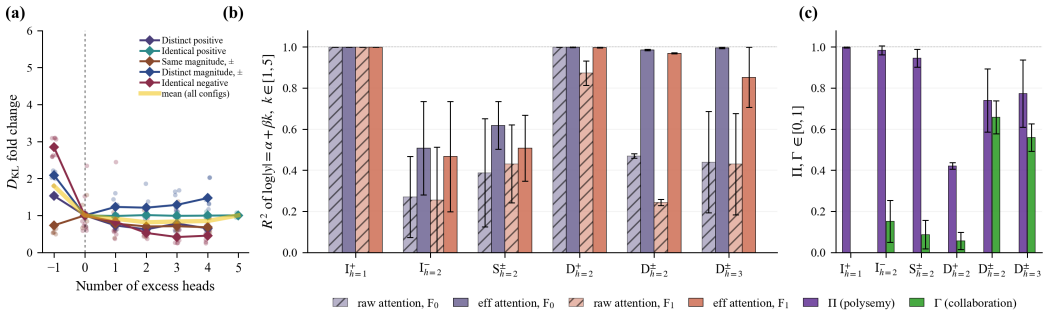

Figure 12: At larger initializations, the model fails to converge to optimal solutions. **(a)**: Compared to the main text, we here see a much more inconsistent pattern, where sometimes adding additional heads will even worsen the model, as it gets stuck in a worse setting during optimization. **(b)**: Effective-attention reveals that for some configurations, the models simply do not learn the correct patterns. We see that the cases with the worst matches are exactly the ones with the largest increase in loss from table 2. **(c)**: Our metrics show evidence of much more degenerate solutions. For the 2 worst cases, we have high polysemy yet low collaboration, indicating a dead head in the model and therefore effectively an underparameterized architecture.

With small initialization, attention patterns achieve high R^2 to the predicted candidate patterns across all configurations (consistent with the main text results). With default initialization, the R^2 is lower — the patterns show exponential-like decay but at rates that deviate from the eigenvalues of the data generator. This is consistent with the rich/lazy distinction: small initialization allows the QK circuit to learn the precise spectral structure from scratch, while large initialization locks in approximate patterns early in training that the optimizer cannot fully correct.

G.5 Interpretation

These results connect to the gauge freedom framework in two ways. First, they show that the degrees of freedom in how attention distributes computation are not merely theoretical — the initialization scale shifts which point in the solution space the optimizer finds, with small initialization favoring specialized solutions and large initialization favoring more entangled configurations where no head cleanly matches a single predicted pattern.

Second, the loss degradation at large initialization suggests that not all points in the solution space are equally accessible to gradient descent. The spectral theory predicts a family of equivalent solutions, but the QK parameterization's inductive bias — shaped by the softmax saturation at large initial weights — restricts which solutions the optimizer can reach. The gauge freedom is real in the loss landscape but constrained in practice by the training dynamics.

H Conic vs. eigenmode decomposition

The conic analysis (Appendix J) establishes that the distinct magnitude $\pm$ configuration ($\zeta_1 = +0.70$, $\zeta_2 = -0.50$) admits solutions at both $H = 2$ ($H_{\min}$, the conic minimum) and $H = 3$ ($H_{\min}$(mode), the eigenmode prediction). Both achieve equivalent loss (Figure 7a), but they implement the computation in qualitatively different ways.

H.1 Two heads: conic decomposition

At $H = 2$, the model cannot assign one head per eigenvalue. There are two eigenvalues but one is negative, requiring sign-alternating contributions that a single non-negative pattern cannot produce. Instead, the model finds a *conic* solution: both heads learn non-standard attention patterns that are

individually difficult to interpret, but whose linear combination through the OV coupling recovers the correct per-factor decay rates.

Figure 7b shows that for the 2-head case ($\mathrm{D}^{\pm}_{h=2}$), the raw attention R^2 is low for both factors. No head cleanly implements either 0.7^k or the even/odd pattern for $(-0.5)^k$. However, the effective subspace attention R^2 is high: the OV coupling successfully unmixes the heads' contributions to recover both ζ_1^{d-s} and ζ_2^{d-s}.

This is the conic decomposition in action. The two heads' patterns are non-negative elements in the feasibility cone (Definition 49), and the coupling matrix decomposes them into the target eigenvalue decays. This requires that no head is fully specialized; all heads must collaborate on all factors.

H.2 Three heads: eigenmode decomposition

At $H = 3$, the model has enough capacity to assign one head per spectral role: one head implementing the positive decay, one implementing the even-offset pattern and one implementing the odd-offset pattern.

Figure 7b confirms this: for the 3-head case ($\mathrm{D}^{\pm}_{h=3}$), the raw attention R^2 for the positive 0.7 factor is much higher, despite loss being equivalent. The model finds it useful to allocate a head primarily to serving exponential decay. There is, however, also higher variance in this metric than for the other cases. The model effectively has a head more than it needs, meaning there is a larger space of equivalent solutions it can find. It has less pressure to implement the exact attention maps. The hard requirement that 2 heads must implement the negative eigenvalue remains, and as such we need effective attention to recover the contribution to the 2nd factor.

H.3 Implications

The crucial observation is that both solutions implement the same per-factor belief update. The residual stream at resid_mid contains the same constrained predictive geometry in both cases, and both achieve equivalent loss. The difference is purely in *legibility*: the 3-head eigenmode solution admits a slightly more directly interpretable solution, while the 2-head conic solution is completely opaque just from the attention maps.

These findings reveal that the opaqueness of raw attention maps can come from multiple sources: in some cases, the model may be taking advantage of degrees of freedom and in others, there may simply exist a minimal decomposition of the mechanism that admits no clearly interpretable map.

I The structure of our training data: A process with two non-Markovian factors

Throughout this paper, we use non-Markovian training data sampled from an ergodic source composed of two independent factors to assess how the transformer would learn to represent and predict tokens from a simple "world" made of parts. Here each part is a particular three-state HMM from the parametrized Mess3 family.

I.1 A Mess3 factor

The Mess3 process Marzen and Crutchfield [2017], Shai et al. [2024], Piotrowski et al. [2025] has three hidden states and three observable tokens $\mathcal{Z} = \{0, 1, 2\}$. Since Mess3 is a 3-state HMM, its local predictive vectors live in a 2-simplex. This 2-simplex corresponds to the set of probability distributions over the three hidden states.

The Mess3 process is defined by two parameters, which can be interpreted as a latent-transition probability $p \in (0, 1/2]$ and observation fidelity $q \in [0, 1]$, with dependent quantities $\beta = (1 - q)/2$ and $y = 1 - 2p$.[1]

[1] In previous publications, p and q were denoted by x and α respectively. We use p and q here to avoid a serious internal conflict of notation.

The labeled transition matrices are:

$$T^{(0)} = \begin{bmatrix} qy & \beta p & \beta p \\ qp & \beta y & \beta p \\ qp & \beta p & \beta y \end{bmatrix} \tag{34}$$

$$T^{(1)} = \begin{bmatrix} \beta y & qp & \beta p \\ \beta p & qy & \beta p \\ \beta p & qp & \beta y \end{bmatrix} \tag{35}$$

$$T^{(2)} = \begin{bmatrix} \beta y & \beta p & qp \\ \beta p & \beta y & qp \\ \beta p & \beta p & qy \end{bmatrix} . \tag{36}$$

The net transition matrix

$$T = T^{(0)} + T^{(1)} + T^{(2)} = (q + 2\beta) \begin{bmatrix} 1-2p & p & p \\ p & 1-2p & p \\ p & p & 1-2p \end{bmatrix} \tag{37}$$

has a uniform stationary distribution $\boldsymbol{\pi} = \boldsymbol{\pi}T = \frac{1}{3}[1 \quad 1 \quad 1]$, and a stationary right eigenstate of $\mathbf{1} = T\mathbf{1} = [1 \quad 1 \quad 1]^\top$.

Besides the stationary eigenvalue of 1, the net transition matrix T has eigenvalue $\zeta = 1 - 3p$, which has (algebraic and geometric) multiplicity of 2 for $p \in (0, 1/2] \setminus \{1/3\}$. (Notice that Mess3 degenerates into a memoryless fair coin at $p = 1/3$.)

The spectral projection operator associated with stationarity is $P_1 = \mathbf{1}\boldsymbol{\pi}$. The spectral projection operator associated with ζ is $P_\zeta = I - P_1 = I - \mathbf{1}\boldsymbol{\pi}$. Note that these satisfy the eigen-relation $P_\lambda T = TP_\lambda = \lambda P_\lambda$, are orthonormal $P_\lambda P_\xi = P_\lambda \delta_{\lambda,\xi}$, and form a decomposition of the identity $\sum_{\lambda \in \Lambda_T} P_\lambda = I$. Accordingly, they form not only a decomposition of the net latent transition operator $T = \sum_{\lambda \in \Lambda_T} \lambda P_\lambda$, but also an eigen-decomposition of powers of T, such that $T^n = \sum_{\lambda \in \Lambda_T} \lambda^n P_\lambda = \mathbf{1}\boldsymbol{\pi} + \zeta^n P_\zeta$. Note that T^n corresponds to the cummulative latent transition dynamic when marginalizing over observed symbols for n time steps.

I.2 The joint dynamic

The minimal HMM capable of generating the probabilistic structure of the training sequences is a 9-state HMM obtained via the tensor product of two constituent Mess3 transition matrices:

$$T^{(x)} = T_1^{(z^{(1)})} \otimes T_2^{(z^{(2)})} \qquad \text{for } x = z^{(1)} + 3z^{(2)} . \tag{38}$$

Practically, training sequences are generated by running two HMMs in parallel and, based on the outputs of the two consituent HMMs, producing a token $x \in \mathcal{X} = \{m\}_{m=0}^{8}$ at each timestep, where each token corresponds to a unique $(z^{(1)}, z^{(2)})$ pair.

J Minimum Architectural Requirement

J.1 Minimum Architecture

For a factored process, let each factor m have stationary distribution $\boldsymbol{\pi}_m$, transition matrix T_m, and non-stationary decay eigenvalue ζ_m. In Mess3, each factor has a two-dimensional non-stationary displacement space—the plane of the 2-simplex. Since the two non-stationary eigenvalues are equal, any vector in the plane of the simplex is an eigenstate of T_m associated with eigenvalue ζ_m. Thus lag propagation acts as a scalar on the entire displacement space:

$$\boldsymbol{y}_m T_m^k = \zeta_m^k \boldsymbol{y}_m \tag{39}$$

for any non-stationary displacement $\boldsymbol{y}_m = \boldsymbol{\eta}_m - \boldsymbol{\pi}_m$ of factor m. Hence, for a source token z_s and lag $k = d - s$, the contribution to factor m can be written as

$$\zeta_m^k \, \mathbf{g}_m(z_s^{(m)}), \tag{40}$$

where $\mathbf{g}_m(z_s^{(m)})$ is the token-dependent displacement direction in the factor-m subspace.

For two factors, write these two token-dependent displacement directions induced by the source token as

$$\mathbf{u} = \mathbf{g}_1(z_s^{(1)}), \qquad \mathbf{v} = \mathbf{g}_2(z_s^{(2)}). \tag{41}$$

Because the belief readout is factorized, the residual displacement decomposes as a direct sum of factor-wise displacements. Therefore the required lag-k update has the idealized form

$$\mathbf{p}_k = \zeta_1^k \mathbf{u} + \zeta_2^k \mathbf{v} \, . \tag{42}$$

Using $(\mathbf{u}, \mathbf{v})$ as coordinates, this becomes the lag-update curve

$$\mathbf{p}_k = (\zeta_1^k, \zeta_2^k) \in \mathbb{R}^2. \tag{43}$$

We now ask how many attention heads are needed to express this family of updates under an idealized OV coupling model. The OV circuit of each head defines how strongly that head writes into the displacement subspace of each factor. In the two-factor case, collect these fixed OV couplings into a heads-by-factors matrix

$$B = \begin{pmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \\ \vdots & \vdots \\ b_{H,1} & b_{H,2} \end{pmatrix}. \tag{44}$$

The h-th row

$$\mathbf{b}_h = (b_{h,1}, b_{h,2}) \tag{45}$$

is the factor-space write direction supplied by the OV circuit of head h. Equivalently, in the original residual space, head h's relevant OV contribution is idealized as

$$b_{h,1}\mathbf{u} + b_{h,2}\mathbf{v} \, . \tag{46}$$

The attention mechanism supplies a lag-dependent nonnegative scalar $A_h(k) \geq 0$ for each head. Thus the total update expressible by H heads has factor coordinates

$$\mathbf{p}_k = B^\top A(k) = \sum_{h=1}^{H} A_h(k) \begin{pmatrix} b_{h,1} \\ b_{h,2} \end{pmatrix}, \qquad A_h(k) \geq 0. \tag{47}$$

This separates the two roles: the OV circuits supply fixed factor-write directions, while the QK/attention circuits supply the lag-dependent nonnegative coefficients. A single head can vary how much it writes as a function of lag, but its relative coupling to the two factors is fixed by its OV row.

Therefore, under this conic OV-coupling abstraction, the relevant head-count quantity is not simply the number of eigenvalues or factors. It is the minimum number of OV coupling rays whose nonnegative cone contains the finite lag-update set

$$\left\{ (\zeta_1^k, \zeta_2^k) : k \geq 0 \right\}. \tag{48}$$

Equivalently,

$$H_{\min} = \min \left\{ H : \left\{ (\zeta_1^k, \zeta_2^k) : k \geq 0 \right\} \subseteq \mathrm{cone}(b_1, \ldots, b_H) \right\}. \tag{49}$$

This is a capacity calculation for the OV geometry. It assumes that the OV maps can realize the required factor-level coupling directions, while the attention mechanism must still learn the corresponding lag-dependent coefficients $A_h(k)$. Thus the transformer need not assign one head to each factor or to each eigenvalue. Instead, it may learn a conic decomposition of the spectral update geometry. Whether the transformer prefers this solution in practice is examined in section H.

Applying the rule above to our different configurations yields table 3 (partially copied from the experiments section).

Table 3: Minimum head count under different two-eigenvalue configurations.

Configuration	Lag-update geometry	$H_{\min}$
$+\zeta_1, +\zeta_2,\ \lvert\zeta_1\rvert \neq \lvert\zeta_2\rvert$	Curve in positive quadrant	2
$+\zeta_1, +\zeta_1$	Single ray	1
$+\zeta_1, -\zeta_1$	Two parity rays in half-plane	2
$+\zeta_1, -\zeta_2,\ \lvert\zeta_1\rvert \neq \lvert\zeta_2\rvert$	Two parity branches in half-plane	2
$-\zeta_1, -\zeta_1$	One line through origin	2
$-\zeta_1, -\zeta_2,\ \lvert\zeta_1\rvert \neq \lvert\zeta_2\rvert$	Curve in alternating quadrants	3

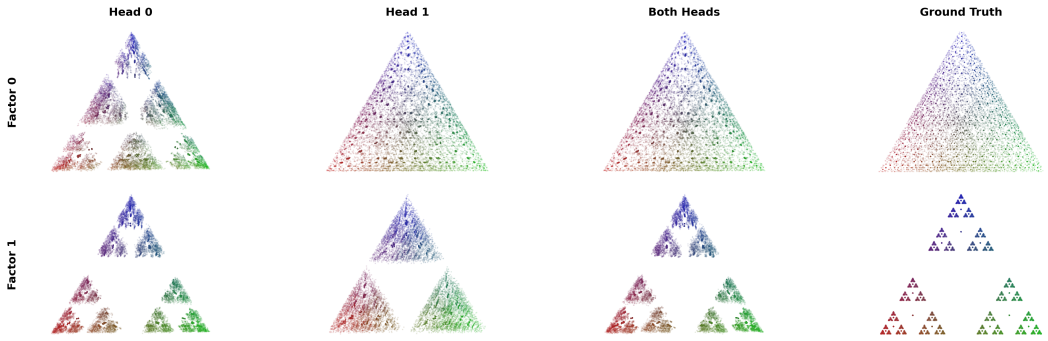

Figure 13: For $\zeta_1 = 0.7$, $\zeta_2 = 0.4$, regressing from the individual heads + the sum of both heads to the constrained belief simplex shows strong alignment between each head and a specific geometry.

K Belief State Geometry

To visualize how individual heads contribute to the constrained belief geometry, we regress from partial attention-layer outputs to the ground-truth factored constrained belief state. Let x_d^{pre} denote the residual stream before the attention layer at destination position d, and let $o_d^{(h)}$ denote the OV-routed output of head h at the same position. For a two-head model, we construct three activation sets:

$$x_d^{(1)} = x_d^{\text{pre}} + o_d^{(1)}, \qquad x_d^{(2)} = x_d^{\text{pre}} + o_d^{(2)}, \qquad x_d^{(1,2)} = x_d^{\text{pre}} + o_d^{(1)} + o_d^{(2)}.$$

These correspond to adding only head 1, only head 2, or both heads to the pre-attention residual stream.

For each activation set $S \in \{1, 2, (1, 2)\}$, we fit a linear map

$$\widehat{r}_d^S = W_S x_d^S + b_S$$

to predict the factored constrained belief state. In the two-factor Mess3 setting, each factor has a three-dimensional constrained belief vector, so the regression target is

$$r_d^{\text{fact}} = [r_d^{(0)}; r_d^{(1)}] \in \mathbb{R}^6.$$

For visualization, we plot the recovered belief for each factor separately. Each three-dimensional factor belief is mapped to RGB coordinates, so points with similar inferred beliefs have similar colors. Comparing the regressions from $x_d^{(1)}$, $x_d^{(2)}$, and $x_d^{(1,2)}$ lets us see whether the constrained belief geometry is linearly recoverable from individual head contributions or only from their sum.

For the 2 cases we walked through in the main body, $\zeta_1 = 0.7$, $\zeta_2 = 0.4$ and $\zeta_1 = 0.5$, $\zeta_2 = -0.5$, we show here the per-head and combined regressions to the belief state geometry in figures 13 and 14.

L Experiment details

L.1 Model hyperparameters

We train pre-norm, decoder-only transformers on next-token prediction with cross-entropy loss. We use a single attention layer, sequence length of 11 (including BOS token), batch size of 128,

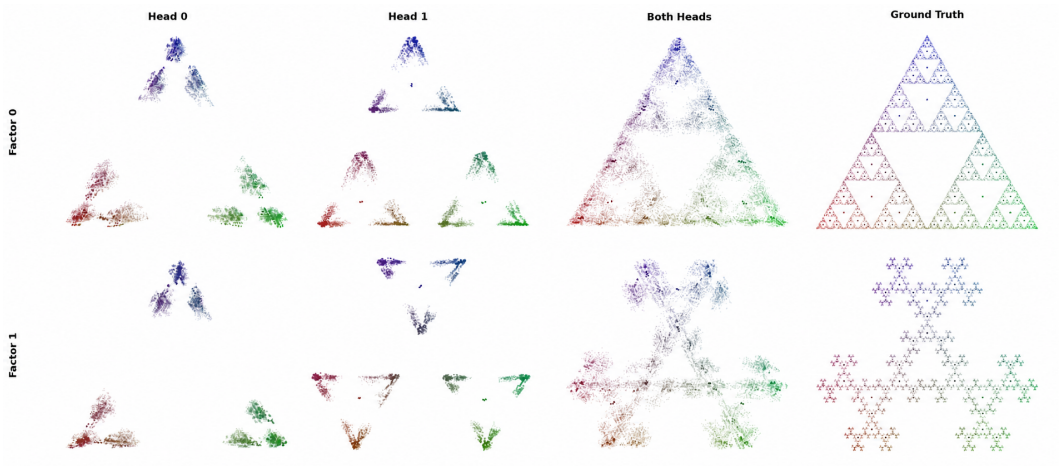

Figure 14: For $\zeta_1 = 0.5$, $\zeta_2 = -0.5$, regressing from the individual heads + the sum of both heads to the constrained belief simplex shows that no head is strongly responsible for specific ground truth geometry. In fact, both geometries require both heads.

$d_{\text{model}} = 120$, $d_{\text{head}} = d_{\text{model}} \,//\, n_{\text{heads}}$, and $d_{\text{MLP}} = 4 \cdot d_{\text{model}}$. We use the Adam optimizer with a learning rate of $5e-4$ and train for 100,000 steps.

L.2 Compute

We trained on Nvidia H100 GPUs. No model at any point during training took up more than 10GB of total GPU memory, and every model required less than 20 minutes of training.

L.3 Mess3 parameters

Table 4: Mess3 parameters for each two-factor configuration. For each factor, the transition parameter is determined by $p = (1-\zeta)/3$, and the emission parameter is fixed to $q = 0.6$.

Configuration	ζ_1	ζ_2	p_1	p_2	q_1	q_2
Distinct positive	$+0.7$	$+0.4$	0.1	0.2	0.6	0.6
Identical positive	$+0.7$	$+0.7$	0.1	0.1	0.6	0.6
Same magnitude, mixed sign	$+0.5$	-0.5	$1/6$	0.5	0.6	0.6
Distinct magnitude, mixed sign	$+0.7$	-0.5	0.1	0.5	0.6	0.6
Identical negative	-0.5	-0.5	0.5	0.5	0.6	0.6

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: **[TODO]**

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: **[TODO]**

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: **[TODO]**

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: **[TODO]**

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: **[TODO]**

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: **[TODO]**

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: **[TODO]**

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: **[TODO]**

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: **[TODO]**

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This paper is primarily theoretical and mechanistic. It studies transformers in a controlled synthetic setting where the data-generating process and target computations are analytically specified, with the goal of understanding when attention heads should or should not be interpreted as computational units. The work does not introduce a deployed system, application-specific model, dataset involving people, or new capability intended for real-world decision-making or content generation. We therefore do not identify a direct path to specific positive or negative societal impacts beyond the broad and indirect effects of foundational interpretability research.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: **[TODO]**

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification: **[TODO]**

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: **[TODO]**

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: **[TODO]**

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: **[TODO]**

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: **[TODO]**

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.