

# The Bayesian Geometry of Transformer Attention

Paper I of the Bayesian Attention Trilogy

NAMAN AGGARWAL\*, Dream Sports, USA

SIDDHARTHA R. DALAL, Columbia University, USA

VISHAL MISRA, Columbia University, USA

Transformers often appear to perform Bayesian reasoning in context, but verifying this rigorously has been impossible: natural data lack analytic posteriors, and large models conflate reasoning with memorization. We address this by constructing *Bayesian wind tunnels*—controlled environments where the true posterior is known in closed form and memorization is provably impossible. In these settings, small transformers reproduce Bayesian posteriors with  $10^{-3}$ – $10^{-4}$  bit accuracy, while capacity-matched MLPs fail by orders of magnitude, establishing a clear architectural separation.

Across two tasks—bijection elimination and Hidden Markov Model (HMM) state tracking—we find that transformers implement Bayesian inference through a consistent geometric mechanism: residual streams serve as the belief substrate, feed-forward networks perform the posterior update, and attention provides content-addressable routing. Geometric diagnostics reveal orthogonal key bases, progressive query–key alignment, and a low-dimensional value manifold parameterized by posterior entropy. During training this manifold unfurls while attention patterns remain stable, a *frame–precision dissociation* predicted by recent gradient analyses.

Taken together, these results demonstrate that hierarchical attention realizes Bayesian inference by geometric design, explaining both the necessity of attention and the failure of flat architectures. Bayesian wind tunnels provide a foundation for mechanistically connecting small, verifiable systems to reasoning phenomena observed in large language models.

## 1 Introduction

Do transformers perform Bayesian inference, or do they merely imitate it through pattern matching? Natural language offers no ground-truth posterior against which to verify predictions, and modern LLMs are too large and too entangled with their data to separate genuine probabilistic computation from memorization. Even when models *behave* Bayesianly, there is no direct way to confirm that the internal computation matches Bayes’ rule.

**Our approach.** We replace unverifiable natural data with *Bayesian wind tunnels*: controlled prediction tasks where

- (1) the *analytic posterior* is known exactly at each step,
- (2) the *hypothesis space* is so large that memorization is impossible,
- (3) in-context prediction requires *genuine probabilistic inference*.

This converts a qualitative question (“does it do Bayes?”) into a quantitative test: does the model’s predictive entropy match the analytic posterior entropy position by position?

**Two wind tunnels.** We study two settings of increasing difficulty:

- **Bijection learning:** a discrete hypothesis-elimination problem with a closed-form posterior.
- **Hidden Markov Models (HMMs):** a sequential, stochastic inference problem requiring recursive updates.

Transformers achieve machine-level Bayesian consistency in both. Capacity-matched MLPs trained identically fail catastrophically in both.

\*Currently at Google DeepMind. Work performed while at Dream Sports.

**Mechanistic discovery.** Across tasks, transformers implement Bayesian inference through a unified three-component architecture:

- (1) **Residual stream as belief state:** posterior information accumulates layer-by-layer.
- (2) **Feed-forward networks as Bayesian update:** FFNs perform the numerical posterior computation.
- (3) **Attention as routing:** QK geometry retrieves the relevant components of the belief for each update.

Geometric diagnostics reveal orthogonal key axes, progressive query–key alignment, and a one-dimensional value manifold that unfolds during training. These observations match predictions from recent gradient-based analyses of transformer learning.

**Contribution.** This paper provides the first empirical proof that transformers can realize exact Bayesian posteriors, identifies the geometric mechanism by which this occurs, and introduces Bayesian wind tunnels as a tool for probing algorithmic reasoning in small, verifiable settings.

## 2 Theoretical Framework: Cross-Entropy and Bayesian Inference

Cross-entropy training on contextual prediction tasks has a well-known population optimum: the Bayesian posterior predictive distribution. This section formalizes that connection. The theory establishes *what* the learned function should be in the infinite-data, infinite-capacity limit; the empirical sections evaluate *which architectures can approximate it* in finite settings.

### 2.1 Setup

Consider a family of tasks indexed by a latent parameter  $\theta \sim \pi(\theta)$ . For each task:

- inputs  $x$  are drawn from some distribution (possibly adversarial or chosen by the experimenter),
- labels are drawn according to  $y \sim p(y | x, \theta)$ ,
- the model observes a context  $c = \{(x_i, y_i)\}_{i=1}^k$  and must predict  $y$  for a new query input.

We train a model  $q(y | x, c)$  by minimizing population cross-entropy:

$$\mathcal{L}(q) = \mathbb{E}_{\theta \sim \pi} \mathbb{E}_{c, (x, y) \sim p(\cdot | \theta)} [-\log q(y | x, c)] . \quad (1)$$

### 2.2 Cross-entropy minimizes to the Bayesian posterior predictive

**THEOREM 1 (POPULATION OPTIMUM OF CROSS-ENTROPY).** *The minimizer of (1) is the Bayesian posterior predictive distribution*

$$q^*(y | x, c) = \int p(y | x, \theta) p(\theta | c) d\theta, \quad (2)$$

where

$$p(\theta | c) \propto \pi(\theta) \prod_{(x_i, y_i) \in c} p(y_i | x_i, \theta). \quad (3)$$

**PROOF.** Fixing  $(x, c)$  and taking expectation over  $y \sim p(\cdot | x, c)$ ,

$$\arg \min_q \mathbb{E}[-\log q(y | x, c)] = p(y | x, c),$$

which equals  $\int p(y | x, \theta) p(\theta | c) d\theta$  by Bayes' rule and the factorization  $(y \perp c) | (x, \theta)$ .  $\square$

**Remark 1.** This result is *architecture-agnostic*: it defines the Bayes-optimal function but not whether any particular architecture can represent or learn it. Our experiments address this realizability question directly.

### 2.3 Application to the bijection wind tunnel

In the bijection task, each  $\theta$  is a bijection  $\pi : \{1, \dots, V\} \rightarrow \{1, \dots, V\}$ . A training sequence reveals  $k - 1$  input--output pairs. Let  $O_{k-1}$  be the set of outputs already observed. Because each input appears at most once per sequence, the current query  $x_k$  has never been seen before, so Bayes' rule reduces to:

$$p(\pi(x_k) = y \mid c) = \begin{cases} \frac{1}{V - k + 1}, & y \notin O_{k-1}, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

Hence the analytic posterior entropy is

$$H_{\text{Bayes}}(k) = \log_2(V - k + 1), \quad (5)$$

producing a monotone staircase that shrinks by one bit whenever a mapping is revealed.

This closed-form posterior allows direct, position-by-position comparison between model entropy and Bayesian entropy; memorization is impossible because the hypothesis space size  $V!$  is enormous.

### 2.4 Application to the HMM wind tunnel

In the HMM task, each  $\theta$  consists of:

- a transition matrix  $T \in \mathbb{R}^{S \times S}$ ,
- an emission matrix  $E \in \mathbb{R}^{S \times V}$ ,
- an initial state distribution  $\pi_0$ .

After observing  $o_{1:t}$ , the true Bayesian posterior over hidden states is given by the forward algorithm:

$$\alpha_t(s) = p(s_t = s \mid o_{1:t}) = \frac{E(o_t \mid s) \sum_{s'} T(s \mid s') \alpha_{t-1}(s')}{\sum_{s''} E(o_t \mid s'') \sum_{s'} T(s'' \mid s') \alpha_{t-1}(s')}. \quad (6)$$

The analytic posterior entropy is therefore

$$H_{\text{Bayes}}(t) = - \sum_{s=1}^S \alpha_t(s) \log_2 \alpha_t(s). \quad (7)$$

Because every training sequence is generated from a freshly sampled  $(T, E)$ , the hypothesis space is massive and memorization is impossible. The model must learn to (i) parse the header encoding  $T$  and  $E$ , and (ii) implement a recursive Bayesian update.

### 2.5 Implications for model evaluation

The theoretical results above imply a sharp operational test: *a model performs correct Bayesian inference if and only if its predictive entropy tracks (5) or (7) at every position.*

Evaluating the **entropy calibration error**

$$\text{MAE} = \frac{1}{L} \sum_k |H_{\text{model}}(k) - H_{\text{Bayes}}(k)| \quad (8)$$

therefore provides a direct, bit-level measure of Bayesian correctness, independent of accuracy or perplexity.

In later sections we show that transformers achieve near-perfect calibration, while matched MLPs do not.

### 3 Experimental Design

We evaluate whether small transformers can realize exact Bayesian inference by placing them in two controlled “Bayesian wind tunnels” where memorization is impossible and the analytic posterior is known in closed form. The two tasks—bijection learning and Hidden Markov Model (HMM) state tracking—probe different inference structures. Bijections require discrete hypothesis elimination; HMMs require recursive integration of stochastic transitions and emission likelihoods.

Across both settings, the evaluation criterion is simple: *does the model’s predictive entropy  $H_{\text{model}}$  match the analytic posterior entropy  $H_{\text{Bayes}}$  at every position?*

We measure this using mean absolute entropy error (MAE),

$$\text{MAE} = \frac{1}{L} \sum_{t=1}^L |H_{\text{model}}(t) - H_{\text{Bayes}}(t)|, \quad (9)$$

where  $L$  is the number of supervised prediction positions. Because each training instance uses a fresh bijection or a fresh HMM, memorization is infeasible; the model must perform genuine in-context inference.

#### 3.1 Task 1: Bijection Learning

Each sequence is derived from a new random bijection  $\pi : \{1, \dots, V\} \rightarrow \{1, \dots, V\}$  with  $V = 20$ . At position  $k$ , the model has observed  $k - 1$  distinct input–output pairs and must predict  $\pi(x_k)$ . Because inputs never repeat, the Bayes-optimal posterior over  $\pi(x_k)$  is uniform over the  $V - k + 1$  unseen values.

*Bayesian ground truth.* Let  $O_{k-1}$  be observed outputs. Then

$$p(\pi(x_k) = y \mid \text{context}) = \begin{cases} \frac{1}{V-k+1}, & y \notin O_{k-1}, \\ 0, & y \in O_{k-1}, \end{cases}$$

with entropy  $H_{\text{Bayes}}(k) = \log_2(V - k + 1)$ .

*Evaluation.* We compute MAE over a held-out set of 2,000 bijections. Because  $20! \approx 2.4 \times 10^{18}$  possible bijections exist and training uses only  $10^5$  samples, no bijection is seen twice; the task enforces true hypothesis elimination.

*Sequence format.* Each training example is tokenized as

$$[x_1, y_1, \text{SEP}, x_2, y_2, \text{SEP}, \dots, x_{19}, \text{SEP}],$$

with teacher forcing at every  $y_k$  position.

#### 3.2 Task 2: Hidden Markov Model State Tracking

The second wind tunnel probes a qualitatively different inferential structure: recursive belief updating. Each sequence is derived from a fresh HMM with  $S = 5$  hidden states and  $V = 5$  observation symbols. Transition rows and emission rows are drawn independently from a symmetric Dirichlet distribution with all concentration parameters equal to 1 (i.e.,  $\text{Dirichlet}(1, 1, 1, 1, 1)$ ), ensuring diverse and non-degenerate dynamics.

*Sequence format.* Each sequence contains:

- a 10-token **header** encoding flattened  $T$  and  $E$ , and
- $K$  observation—prediction pairs, each consisting of:
  - the observed symbol  $o_t$ ,
  - a supervised prediction at that same position for  $p(s_t \mid o_{1:t})$ .

*Bayesian ground truth: forward algorithm.* For each HMM and for each time  $t$  we compute

$$\alpha_t(s) \propto E(o_t | s) \sum_{s'} T(s | s') \alpha_{t-1}(s'), \quad (10)$$

normalized to  $\sum_s \alpha_t(s) = 1$ . The exact posterior entropy is

$$H_{\text{Bayes}}(t) = - \sum_{s=1}^S \alpha_t(s) \log_2 \alpha_t(s).$$

*Evaluation lengths.* Models are trained on sequences with  $K = 20$  prediction positions and evaluated on:

- $K = 20$  (validation: within training horizon),
- $K = 30$  (1.5× training length),
- $K = 50$  (2.5× training length).

This tests whether the model has learned a position-independent recursive algorithm or has merely memorized a finite-horizon computation.

*Why memorization is impossible.* Each sequence uses new  $T$ ,  $E$  matrices and new stochastic emission trajectories. The space of possible HMMs exceeds  $10^{40}$  even under coarse discretization, ensuring that learned behavior cannot rely on recall of any particular HMM.

### 3.3 Architectures

*Transformers.* We use small but realistic transformer stacks:

- **Bijection transformer (2.67M):** 6 layers, 6 heads,  $d_{\text{model}} = 192$ ,  $d_{\text{ffn}} = 768$ .
- **HMM transformer (2.68M):** 9 layers, 8 heads,  $d_{\text{model}} = 256$ ,  $d_{\text{ffn}} = 1024$ .

Both use learned token embeddings, learned absolute positional embeddings, pre-norm residual blocks, and standard multi-head self-attention.

*Capacity-matched MLP baselines.* To isolate the role of attention, we train MLPs with:

- 18–20 layers,
- width 384–400,
- residual connections and layer normalization,
- identical embedding layers and training protocol as the transformers.

Parameter counts match transformers within 1%. These MLPs serve as controls testing whether hierarchical attention is *architecturally necessary*.

### 3.4 Training Protocol

Training is identical across architectures for each task.

*Optimization.* AdamW with  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ , weight decay 0.01, gradient clipping at 1.0.

*Learning rates.*

- Bijections: constant  $10^{-3}$ .
- HMMs:  $3 \times 10^{-4}$  with 1000-step warmup and cosine decay.

*Data sampling.* Every batch draws fresh bijections or fresh HMMs; sequences never repeat.

*Teacher forcing.* Cross-entropy loss is applied at each supervised prediction position.

*Ablation stability.* Layer-wise and head-wise ablations are reported as averages over three random seeds; the HMM length-generalization results are also evaluated across multiple seeds to ensure robustness.

#### 4 Results: Transformers Track the Bayesian Posterior

We evaluate whether transformers lie on the analytic Bayesian manifold using two behavioral tests: (1) pointwise calibration—does  $H_{\text{model}}(t)$  match  $H_{\text{Bayes}}(t)$  at every position? (2) generalization—does the learned computation extend to unseen bijections, unseen HMMs, and longer sequences?

We present results for bijections and HMMs in parallel, followed by MLP controls and multi-seed robustness.

##### 4.1 Bijection Wind Tunnel: Exact Hypothesis Elimination

A 2.67M-parameter transformer converges to the analytic posterior with near machine precision. Figure 1 shows the predictive entropy

$$H_{\text{model}}(k) = - \sum_y p_{\text{model}}(y \mid x_k, \text{context}) \log_2 p_{\text{model}}(y \mid x_k, \text{context})$$

overlaid on  $H_{\text{Bayes}}(k) = \log_2(V - k + 1)$ . The curves coincide across all positions, including late steps where only 2–4 hypotheses remain.

Quantitatively, the transformer achieves

$$\text{MAE} = 3 \times 10^{-3} \text{ bits},$$

averaged over 2,000 held-out bijections. This error is smaller than single-precision numerical noise in the analytic posterior.

*Per-sequence evidence.* Aggregate calibration could hide averaging artifacts. Figure 2 plots eight individual entropy trajectories. Each displays the characteristic staircase pattern: entropy drops discretely whenever a new input–output pair eliminates hypotheses, and collapses to near zero when an input repeats and the mapping is known. The model performs stepwise Bayesian elimination; it is not matching the curve in expectation but reproducing it sequence by sequence.

*Inside-model consistency.* Layer-wise ablations (Figure 3) show that removing any block increases error by more than an order of magnitude, confirming a deeply compositional computation. Head-wise ablations (Figure 4) identify a single Layer 0 hypothesis-frame head whose removal is uniquely destructive, consistent with the geometric analysis in Section 5.

##### 4.2 HMM Wind Tunnel: Recursive Bayesian State Tracking

The 2.68M-parameter transformer also learns the forward algorithm for HMM inference.

*Within training horizon ( $K=20$ ).* At  $t \leq 20$ , model entropy tracks the exact forward-recursion entropy with

$$\text{MAE} = 7.5 \times 10^{-5} \text{ bits}.$$

The two curves are visually indistinguishable (Figure 5).

*Beyond training horizon ( $K=30, K=50$ ).* To test algorithmic generalization, we roll the model out to  $1.5\times$  and  $2.5\times$  training length. The transformer remains remarkably close to the analytic posterior:

$$\text{MAE}(K = 30) = 1.25 \times 10^{-2}, \quad \text{MAE}(K = 50) = 2.88 \times 10^{-2}.$$

Errors increase smoothly with  $t$ , with *no* discontinuity at  $t = 20$  (the training boundary). This is strong evidence of a position-independent recursive algorithm rather than a finite-horizon memorized computation.

*Per-position calibration.* Figure 6 shows absolute error  $|H_{\text{model}}(t) - H_{\text{Bayes}}(t)|$ . Three patterns emerge:

- (1) early positions are slightly noisier (uncertain initial state);
- (2) mid-sequence positions achieve near-zero error at all lengths;
- (3) late positions degrade smoothly with sequence length, consistent with accumulated numerical drift.

*Per-sequence dynamics.* Figure 7 shows the model tracking sequence-specific fluctuations: entropy dips when emissions strongly identify states and rises when observations are ambiguous. The transformer captures these dynamics exactly.

*Semantic invariance under hidden-state relabeling.* Hidden-state indices are purely symbolic: permuting the labels corresponds to the same latent process. We sample a random permutation  $\sigma$  of  $\{1, \dots, S\}$  and apply it to the HMM parameters by permuting rows and columns of  $T$  (i.e.,  $T'_{\sigma(i), \sigma(j)} = T_{i,j}$ ) and permuting rows of  $E$  (i.e.,  $E'_{\sigma(i), o} = E_{i,o}$ ). We then recompute the analytic posterior under  $(T', E')$  and evaluate the model on sequences generated from the permuted HMM. If the model implements Bayesian filtering rather than associating meaning with specific state IDs, its entropy calibration should be unchanged up to numerical noise. Figure 8 shows MAE before vs. after permutation lies on the diagonal, with  $\Delta\text{MAE}$  concentrated near zero.

### 4.3 Length Generalization Requires Late-Layer Attention

To identify which components support stable rollout, we train a variant transformer in which attention is disabled in the top two layers but FFNs and residuals remain intact.

The no-late-attention model fits the training horizon reasonably well ( $1.57 \times 10^{-3}$  bits), but breaks down under rollout:

$$\text{MAE}(K = 30) = 5.55 \times 10^{-1}, \quad \text{MAE}(K = 50) = 1.79.$$

The degradation factor grows from  $21\times$  (at  $K = 20$ ) to  $62\times$  (at  $K = 50$ ), demonstrating that late-layer attention is not required for fitting  $K = 20$  but *is essential* for stable long-horizon Bayesian updates (Figure 9).

### 4.4 MLP Controls: Architectural Necessity of Attention

Capacity-matched MLPs trained under identical conditions fail in both wind tunnels.

*Bijections.* The MLP achieves  $\text{MAE} \approx 1.85$  bits—about  $618\times$  worse than the transformer— and shows no improvement from 100k to 150k steps. Its entropy curve remains nearly flat, indicating it learns only the marginal output distribution.

*HMMs.* The MLP achieves  $\text{MAE} \approx 0.40$  bits at all lengths (Table 1), showing no sign of recursive computation. The flat per-position error profile (Figure 10) indicates collapse to a position-averaged approximation rather than belief tracking.

These failures cannot be attributed to optimization, data, or capacity. They reflect the absence of content-addressable routing and residual compositionality— key geometric ingredients supplied by attention.

#### 4.5 Multi-Seed Consistency

To ensure that Bayesian tracking is not an artifact of initialization or optimization noise, we repeated all HMM experiments across **five independent random seeds**. Per-position error curves for all seeds (Figure 11) nearly overlap at  $K = 20$ ,  $K = 30$ , and  $K = 50$ .

The seed-to-seed variability is negligible compared to the gap between transformer and MLP performance, confirming that the learned Bayesian algorithm is robust to initialization and training noise.

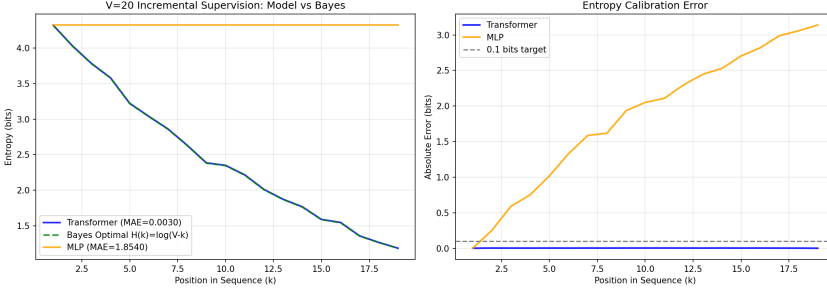


Fig. 1. **Bijection wind tunnel: transformer matches the Bayesian posterior; MLP cannot.** Entropy trajectories at 150k training steps. The transformer lies essentially on top of the analytic Bayes curve across positions, while the capacity-matched MLP barely reduces uncertainty and fails to implement hypothesis elimination. This is the comparison summarized quantitatively in Table 1 and discussed in Section 4.1.

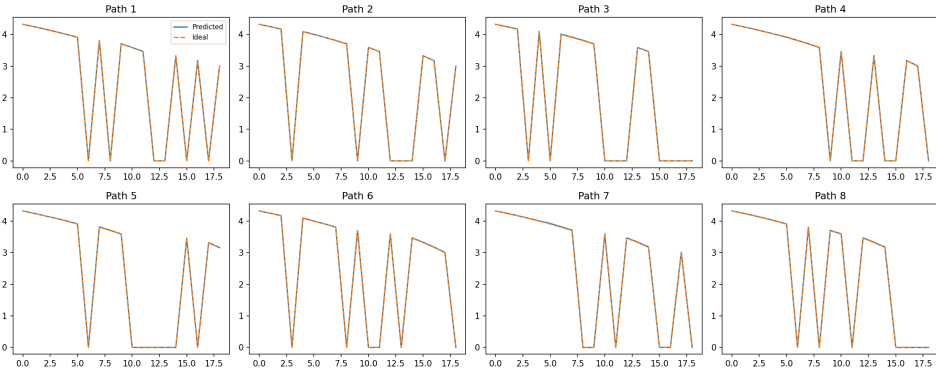


Fig. 2. **Bijection wind tunnel: per-sequence entropy dynamics.** Eight randomly chosen bijections from the test set. Each panel shows transformer entropy (solid) and analytic Bayes entropy (dashed) as a function of position. The sawtooth pattern—discrete drops when mappings are revealed and collapses to (near) zero when previously seen inputs reappear—confirms that the transformer is performing stepwise hypothesis elimination, not merely matching the Bayes curve in aggregate.

### 5 Mechanism: How Transformers Realize Bayesian Inference

The behavioral results in Section 4 demonstrate that small transformers track analytic Bayesian posteriors with sub-bit precision across two distinct wind-tunnel tasks. We now examine *how* this computation is implemented internally. Evidence from ablations, QK geometry, probe dynamics, and

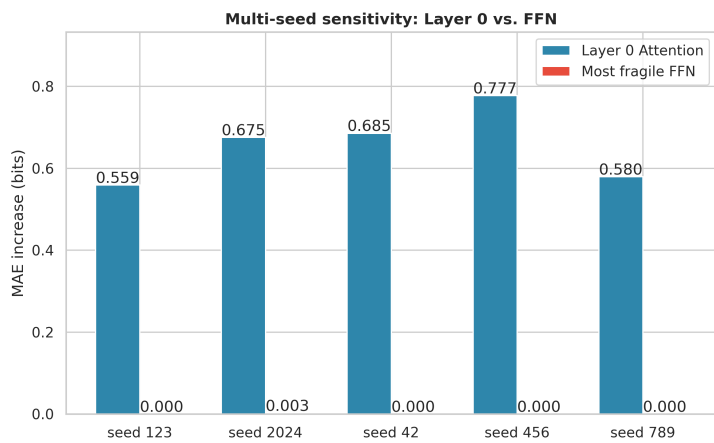


Fig. 3. **Bijection wind tunnel: layer-wise ablation.** Mean absolute entropy error (bits) when ablating each layer (attention+FFN) in turn, averaged over seeds. Removing any single layer increases calibration error by more than an order of magnitude, showing that the Bayesian computation is genuinely hierarchical and compositional rather than shallow or redundant.

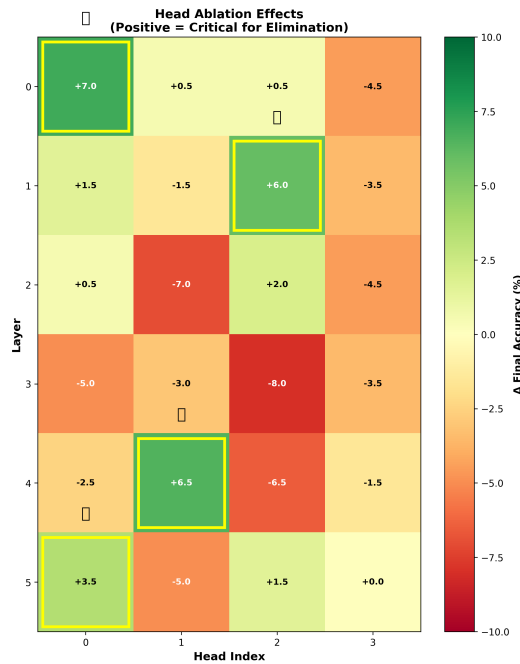


Fig. 4. **Head-wise ablation.** Change in mean absolute entropy error when ablating individual attention heads. A single Layer-0 “hypothesis-frame head” plays a uniquely important role, while many later heads are partially redundant. This supports the three-stage picture in Section 6: foundational binding, progressive elimination, and value-manifold refinement.

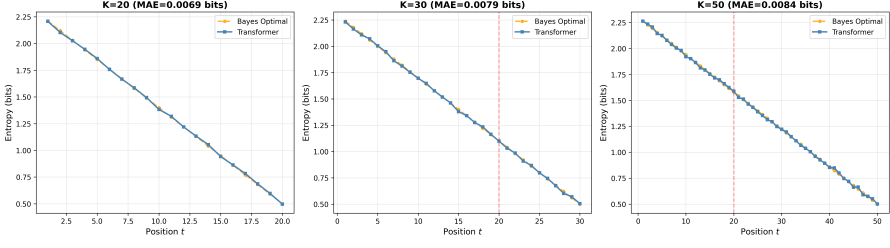


Fig. 5. **HMM wind tunnel: calibration across sequence lengths.** Transformer predictive entropy  $H_{\text{model}}(t)$  (solid) versus analytic  $H_{\text{Bayes}}(t)$  (dashed) at the training length  $K = 20$  and at  $K = 30$  and  $K = 50$ . At  $K = 20$  the trajectories overlap almost perfectly; for longer sequences the error grows smoothly with position and shows no kink at the training boundary, indicating a position-independent recursive algorithm rather than finite-horizon memorization.

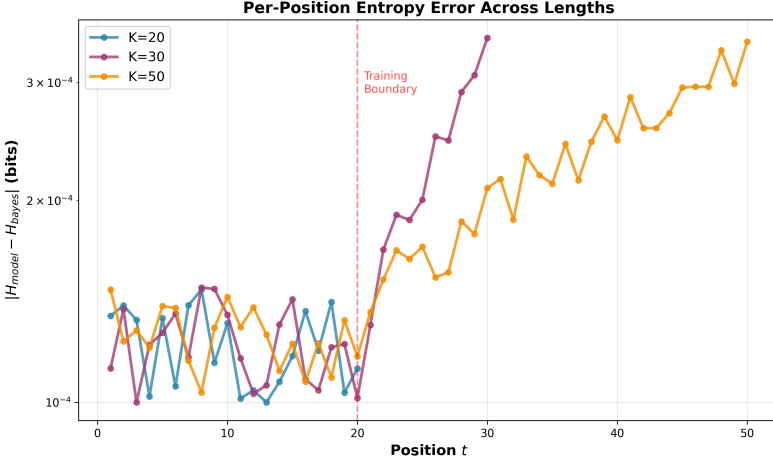


Fig. 6. **HMM wind tunnel: per-position calibration.** Absolute entropy error  $|H_{\text{model}}(t) - H_{\text{Bayes}}(t)|$  as a function of position for  $K = 20$ ,  $K = 30$ , and  $K = 50$ . Errors are tiny at the training length and increase gradually with  $t$  for extended lengths, again with no discontinuity at  $t = 20$ .

Table 1. **HMM wind tunnel: transformer vs MLP calibration across lengths.** Mean absolute entropy error (bits) between model entropy and analytic Bayes entropy. The transformer achieves near-perfect calibration and degrades gracefully with length; the capacity-matched MLP fails catastrophically, with errors  $\sim 0.4$  bits at all positions and lengths.

Model	$K = 20$ (training)	$K = 50$ ( $2.5 \times$ length)
Transformer (2.68M)	$7.5 \times 10^{-5}$	$2.88 \times 10^{-2}$
MLP (2.70M)	$4.09 \times 10^{-1}$	$4.02 \times 10^{-1}$
Degradation factor	$5,467 \times$	$14 \times$

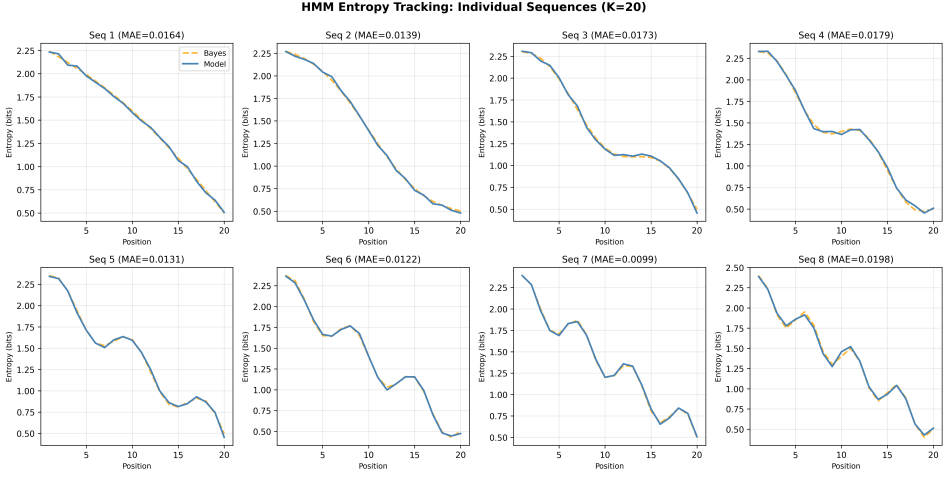


Fig. 7. **HMM wind tunnel: per-sequence entropy dynamics.** Entropy trajectories  $H_{\text{model}}(t)$  and  $H_{\text{Bayes}}(t)$  for eight randomly chosen  $K = 20$  test HMMs. The transformer tracks sequence-specific rises and drops in uncertainty, reflecting the stochastic interplay of transitions and emissions.

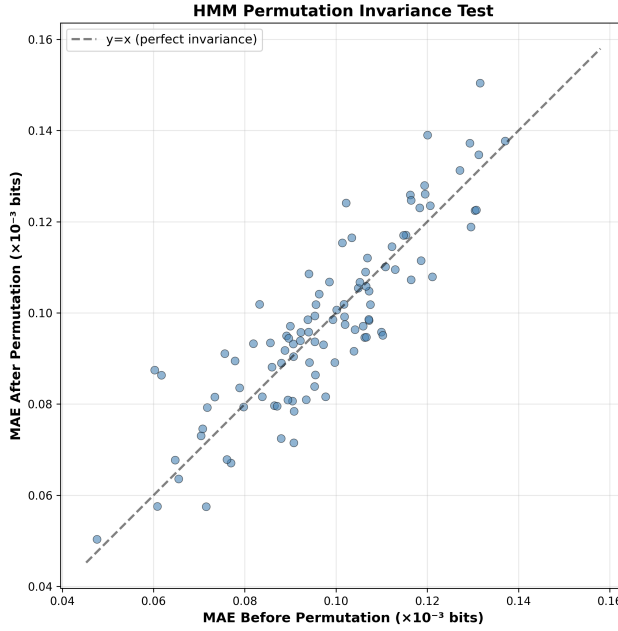


Fig. 8. **Semantic invariance under hidden-state relabeling.** Mean absolute entropy error before vs. after randomly permuting hidden-state labels in the HMMs. Points lie on the diagonal and the distribution of  $\Delta\text{MAE}$  is tightly concentrated near zero, confirming that the transformer's computation is invariant to arbitrary relabelings of the hidden state space.

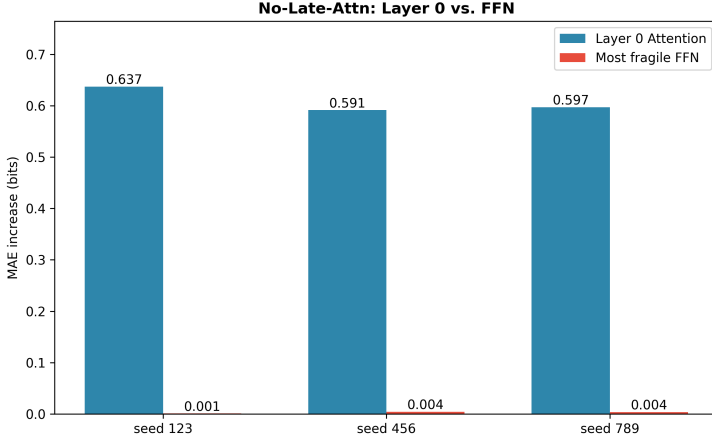


Fig. 9. **Late-layer attention and length generalization.** Mean absolute entropy error as a function of sequence length for the full transformer and a variant with attention disabled in the top two layers. The no-late-attention model is only modestly worse at the training length but its error explodes on longer sequences, with the degradation factor growing from  $\sim 21\times$  at  $K = 20$  to over  $60\times$  at  $K = 50$ . Late attention is therefore crucial for stable rollout beyond the training horizon.

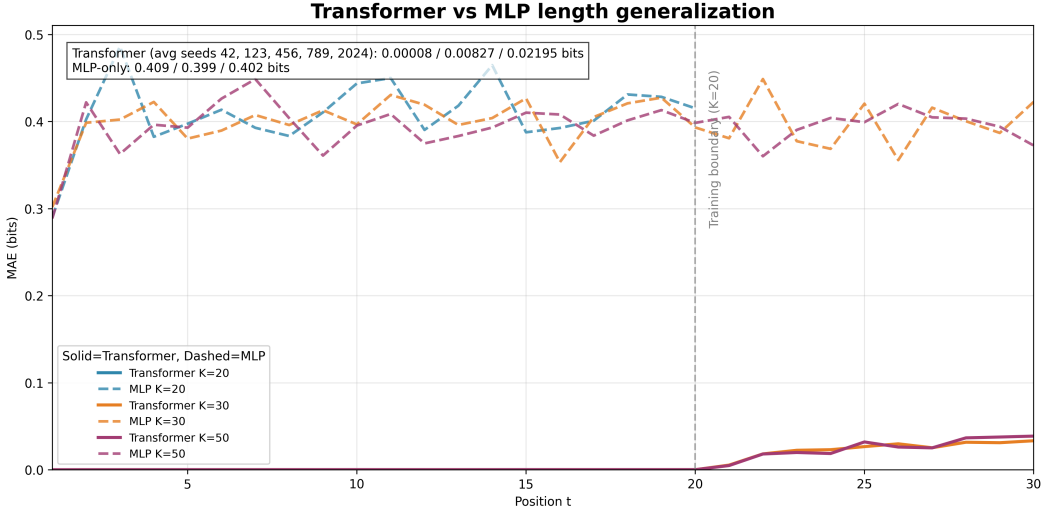


Fig. 10. **HMM wind tunnel: transformer vs MLP length generalization.** Per-position mean absolute entropy error for the transformer (solid) and capacity-matched MLP (dashed) at  $K = 20$  and  $K = 50$ . The vertical gray line marks the training boundary at position  $t = 20$ . The transformer shows near-zero error at the training length and smooth degradation beyond it; the MLP maintains flat  $\sim 0.4$ -bit error across positions, indicating failure to learn recursive Bayesian updates.

training trajectories reveals a consistent architectural mechanism: transformers perform Bayesian inference by constructing a representational frame, executing sequential eliminations within that frame, and progressively refining posterior precision across layers.

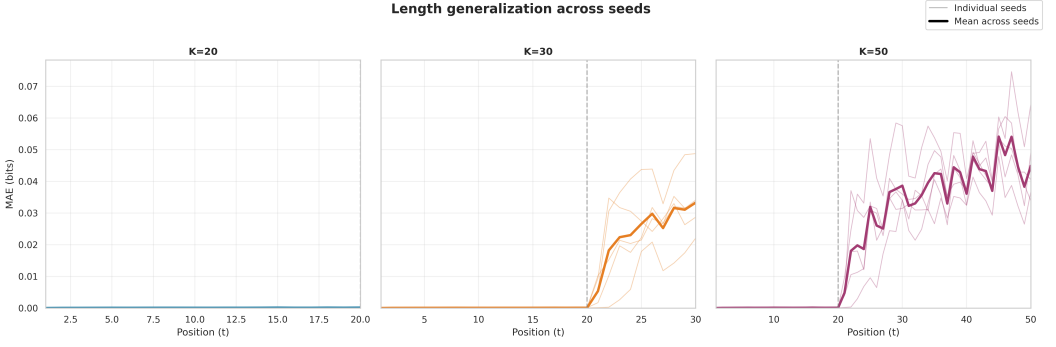


Fig. 11. **Multi-seed robustness of HMM length generalization.** Overlay of per-position transformer MAE curves across five random seeds for  $K = 20$ ,  $K = 30$ , and  $K = 50$ . Seed-to-seed variability is negligible relative to the transformer—MLP gap, showing that the learned Bayesian algorithm is robust to initialization and optimization noise.

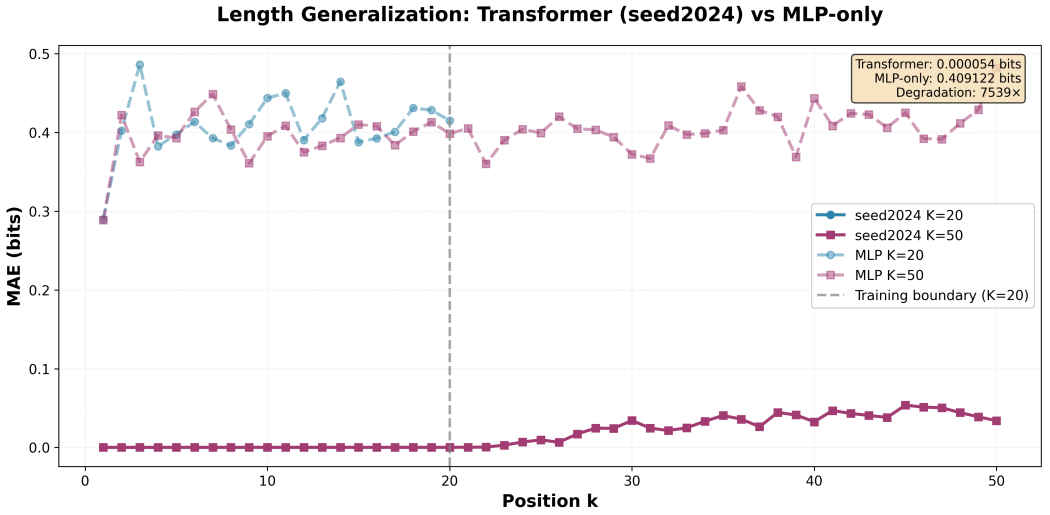


Fig. 12. **Representative single-seed trajectory.** Per-position MAE for one representative seed (2024) closely matches the multi-seed average in Figure 11, further confirming that the length generalization pattern is not an artifact of a particular initialization.

### 5.1 Layer 0 Creates the Hypothesis Frame

The computation begins with a structural operation: Layer 0 attention constructs the *hypothesis space* in which all subsequent inference takes place. Keys at this layer form an approximately orthogonal basis over input tokens (Figure 14), providing a coordinate system over which posterior mass can be represented and manipulated.

Head-wise ablations confirm the indispensability of this step. A single Layer 0 “hypothesis-frame head” dominates the layer’s contribution (Figure 4), and ablating this head alone severely disrupts calibration. Here “hypothesis-frame head” means the head whose keys span the near-orthogonal basis over hypothesis tokens and whose values instantiate the corresponding per-hypothesis slots

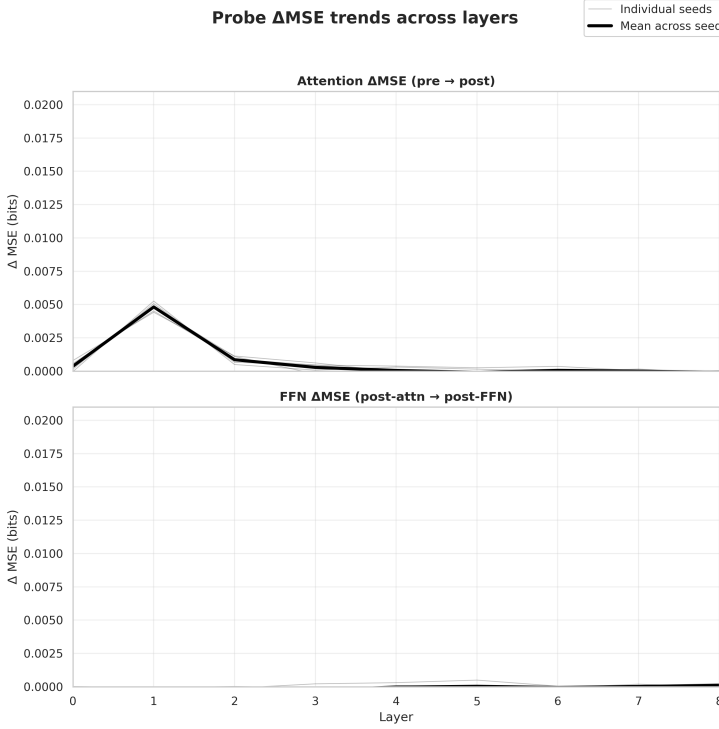


Fig. 13. **Block-wise probe deltas for entropy prediction.** For each transformer block we train a linear probe on the *pre*-sublayer residual stream to predict the analytic posterior entropy, then evaluate the same probe on the *post*-sublayer residual. The plotted quantity is the change in mean-squared error (MSE) when moving from pre- to post-sublayer, i.e.,  $\Delta\text{MSE} = \text{MSE}(\text{probe on post-residual}) - \text{MSE}(\text{probe on pre-residual})$ , so negative values mean the block improves an entropy-linear probe. Positive values indicate that the block reduces probe error. FFN layers account for the largest reductions in MSE, showing that they implement most of the numerical Bayesian update, while attention primarily provides routing rather than performing the heavy probabilistic computation.

in the residual stream. No other attention head exhibits comparable sensitivity. This identifies a structural bottleneck: forming the hypothesis frame is a prerequisite for any later Bayesian computation.

Once established, this frame remains stable through training. Attention maps at Layer 0 change little across checkpoints, even as the value manifold and calibration improve substantially. The model therefore learns the geometry of the inference problem early, and subsequently refines numerical precision within this fixed frame.

## 5.2 Sequential Bayesian Elimination Across Depth

With the hypothesis frame in place, the middle layers perform a layer-by-layer process that mirrors Bayesian elimination.

*Progressive QK sharpening.* As depth increases, queries align more strongly with the subset of keys consistent with the observed evidence (Figure 15). Early layers attend broadly; deeper

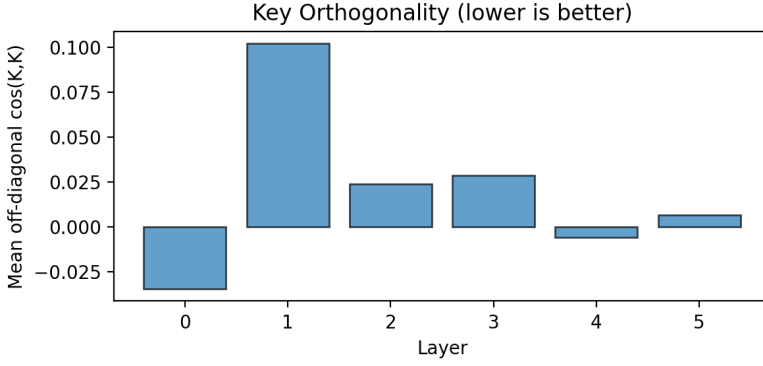


Fig. 14. **Key orthogonality in Layer 0.** Cosine similarity matrix of key vectors for all input tokens in the bijection model at 150k steps. Off-diagonal entries cluster near zero, showing that distinct inputs occupy nearly orthogonal directions and form an explicit hypothesis basis.

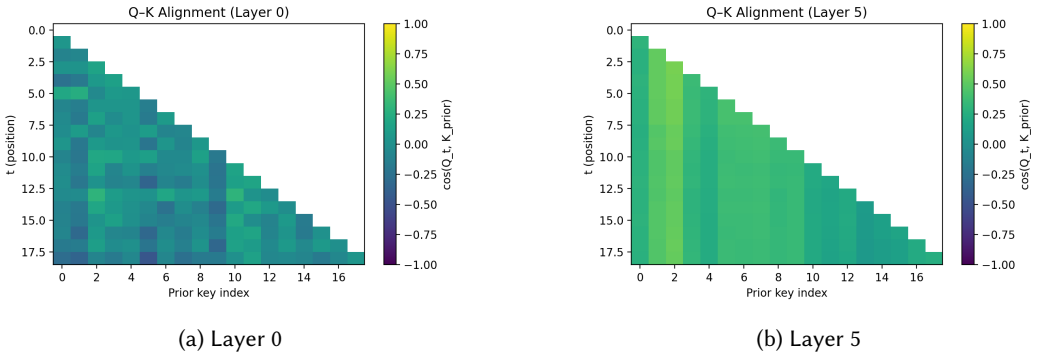


Fig. 15. **Progressive query–key alignment across depth.** Cosine similarity between queries and keys at an early layer (left) and a deep layer (right) of the bijection transformer. For each *sequence position*  $t$  on the horizontal axis, we plot the cosine similarities  $\cos(q_t, k_j)$  between the query at position  $t$  and all key vectors  $k_j$  along the vertical axis. Here  $t$  indexes the query-token positions in the serialized input sequence (i.e., the positions where the model must predict), not the token identity; separator/header tokens are included only insofar as they occupy sequence positions. In Layer 0, attention is diffuse over many keys; by Layer 5 it concentrates sharply on the remaining feasible hypothesis keys, making sequential elimination visible as geometric focusing in Q–K space.

layers concentrate attention almost exclusively on the feasible hypotheses. This geometric focusing parallels analytic Bayesian conditioning, where inconsistent hypotheses receive vanishing weight.

*Hierarchical compositionality.* Layer-wise ablations (Figure 3) show that removing any single layer (attention + FFN, as implemented) increases calibration error by more than an order of magnitude. This demonstrates that the computation is not shallow or redundant. Each layer provides a distinct and non-interchangeable refinement step, forming a sequential, compositional realization of Bayesian updates.

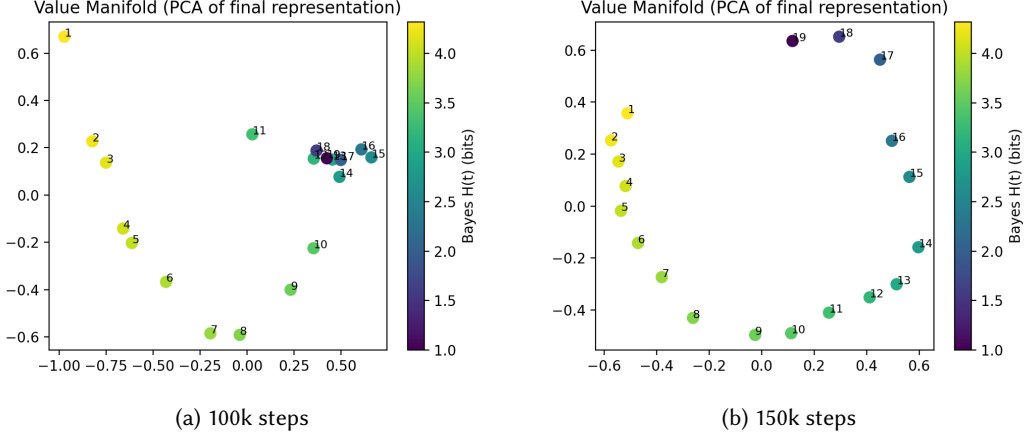


Fig. 16. **Value manifold unfurling during training.** PCA projection of attention outputs in the bijection model, colored by analytic posterior entropy. At 100k steps, low-entropy states are tightly clustered; by 150k, they lie along a smooth one-dimensional curve parameterized by entropy, enabling fine-grained encoding of posterior states. Each point is an attention output (head output or block attention output — whichever you used) at a supervised prediction position; PCA is fit on the pooled outputs and then plotted, colored by analytic posterior entropy.

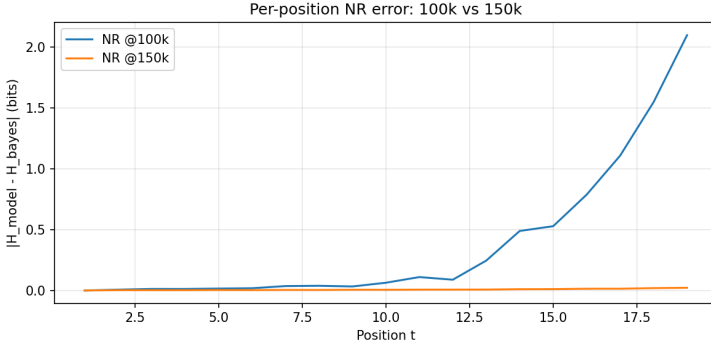


Fig. 17. **Per-position calibration improves as the value manifold unfurls.** Absolute entropy error as a function of position in the bijection task at 100k and 150k training steps. The dominant improvements occur at late positions, matching the geometric unfurling of low-entropy states in Figure 16.

Together, these observations indicate that transformers implement Bayesian elimination not via a single transformation, but through a depth-wise sequence of projections and refinements within the Layer 0 frame.

### 5.3 Attention as Content-Addressable Routing

Across all depths, attention serves a consistent geometric role: it retrieves the components of the belief state relevant for the next update.

Three observations support this routing interpretation:

- **Orthogonal keys** (Figure 14) provide a basis for content-addressable lookup of hypotheses.

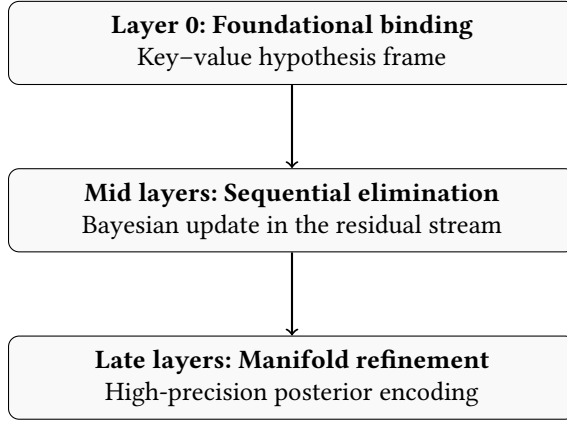


Fig. 18. Three-stage architectural mechanism for Bayesian inference. Layer 0 constructs a key-value hypothesis frame, mid layers implement sequential Bayesian updates in the residual stream, and late layers refine the representation on a low-dimensional posterior manifold.

- **Sharpened QK alignment across depth** (Figure 15) routes residual-stream information toward the feasible hypothesis subspace.
- **Stable routing during late refinement** (Figures 16 and 17) shows that once the frame is correct, attention maps change minimally even as calibration improves.

Routing is also essential for maintaining stable recursive inference. In the HMM task, disabling attention only in the top two layers leaves performance within the training horizon largely intact, but long-horizon inference collapses (Figure 9). Thus attention is required both for forming the initial hypothesis frame and for sustaining stable belief updates under extended rollout.

#### 5.4 Value-Space Manifolds and Precision Refinement

After routing stabilizes, the final layers refine the *precision* of the posterior representation. Figures 16 and 17 show that:

- At intermediate checkpoints, value representations of low-entropy states are nearly collapsed and cannot reliably encode distinctions among small remaining hypothesis sets.
- By the final checkpoint, these states lie along a smooth *one-dimensional manifold* parameterized by posterior entropy.

This geometric unfurling enables fine-grained encoding of posterior confidence and accounts for late-position improvements in calibration. Importantly, this refinement occurs while attention maps remain nearly unchanged, producing a clear *frame-precision dissociation*: attention defines where information flows, while downstream transformations refine how precisely beliefs are encoded.

#### 5.5 Synthesis: A Three-Stage Architectural Mechanism

Across both wind tunnels, the evidence aligns into a three-stage mechanism (Figure 18):

- (1) **Foundational binding (Layer 0)**. Construct an orthogonal hypothesis frame. (Key geometry; catastrophic Layer 0 head ablations.)
- (2) **Progressive elimination (middle layers)**. Sequentially suppress inconsistent hypotheses through sharpening QK alignment. (Layer-wise compositionality; geometric focusing.)
- (3) **Precision refinement (late layers)**. Encode posterior entropy on a smooth value manifold while keeping routing fixed. (Value-manifold unfurling; frame-precision dissociation.)

This structure mirrors the analytic decomposition of Bayesian conditioning: define a hypothesis space, update beliefs with evidence, and refine confidence as uncertainty decreases.

## 5.6 Relation to Gradient-Dynamics Predictions

These empirical observations match predictions from recent analyses of gradient dynamics, which show that attention scores tend to stabilize once the correct routing structure has formed, while value and residual representations continue to refine precision. The observed stability of attention maps together with the unfolding of the value manifold provides direct evidence for this *differential convergence* of routing and precision.

## 6 Analysis and Discussion

The wind-tunnel experiments demonstrate that small transformers, trained with standard optimization and without architectural modifications, implement Bayesian inference with striking fidelity. In this section we discuss the broader implications of these results for interpretability, architectural necessity, and the connection between controlled wind tunnels and the behavior of large language models.

### 6.1 Why Hierarchical Attention Implements Bayes

Across the bijection and HMM settings, the internal geometry uncovered in Section 5 reveals a consistent computational pattern. Transformers realize Bayesian conditioning through a stacked sequence of geometric operations:

- (1) **Foundational binding (Layer 0).** Orthogonal keys create a hypothesis frame. The catastrophic effect of ablating the Layer 0 hypothesis-frame head (Figure 4) demonstrates that this frame is structurally indispensable.
- (2) **Progressive elimination (middle layers).** QK-alignment sharpens across depth (Figure 15), mirroring the multiplicative suppression of ruled-out hypotheses in analytic Bayesian updates. Layer-wise ablations (Figure 3) show that each layer contributes a non-interchangeable refinement step.
- (3) **Precision refinement (late layers).** Once routing stabilizes, value representations unfold into a low-dimensional manifold parameterized by posterior entropy (Figure 16), improving calibration particularly at late positions (Figure 17). This frame–precision dissociation reflects a division of labor: attention establishes where information flows, while subsequent transformations refine the numerical precision of the belief.

This hierarchy parallels Bayes’ rule: define a hypothesis space, integrate evidence, and refine the posterior. The transformer implements these steps using attention geometry and residual-stream representations.

### 6.2 Depth as Compositional Necessity

A central conclusion from the ablation studies is that depth is not redundant. In both wind tunnels, removing any individual layer increases calibration error by more than an order of magnitude (Figure 3). This shows that Bayesian reasoning is expressed as a sequence of compositional projections, each layer refining the belief state in a way that cannot be collapsed into a single transformation.

This stands in contrast to wide, shallow architectures: even with comparable parameter counts and identical training, MLPs fail to perform hypothesis elimination or state tracking (Section 4.4). Bayesian inference requires *hierarchical refinement*, and transformers supply the appropriate inductive bias through depth and residual composition.

### 6.3 From Wind Tunnels to Natural Language

While the wind tunnels are deliberately simplified, they capture the essential structure of probabilistic inference: integrating evidence over time to update latent beliefs. Large language models operate in a far more complex setting, with high-dimensional latent spaces and ambiguous, multi-modal evidence. Yet the geometric ingredients observed here—orthogonal hypothesis axes, depth-wise refinement, and stable routing—are structural rather than task-specific.

The results therefore suggest that the probabilistic behaviors exhibited by LLMs may arise not only from scale or data richness but also from architectural geometry. Wind tunnels provide a verifiable lower bound: they show that transformers *can* implement Bayesian inference exactly when the posterior is known.

### 6.4 Architectural Necessity and MLP Failure

The capacity-matched MLP controls clarify which architectural components are essential. Even with similar parameter counts and identical data exposure, MLPs fail catastrophically in both wind tunnels, with entropy errors on the order of 0.4 bits (Table 1). These failures are not due to optimization difficulties: the tasks are simple, gradients are well-behaved, and training converges smoothly.

Instead, the gap reflects the absence of:

- content-addressable retrieval of hypotheses,
- compositional refinement through depth, and
- stable routing structures that support long-horizon inference.

Transformers succeed because attention supplies the geometric mechanisms—orthogonal bases, selective routing, and progressive focusing—that MLPs lack. The failure of matched MLPs therefore serves as a clean demonstration that attention is not merely useful but architecturally necessary for Bayesian structure learning in context.

### 6.5 A Lower Bound for Reasoning in LLMs

The wind tunnels establish a principled baseline for mechanistic reasoning in transformers. If a model cannot implement Bayes in a setting with a closed-form posterior and impossible memorization, it offers little evidence of genuine inference capability in natural language. Conversely, the fact that small, verifiable transformers succeed here—with interpretable geometric mechanisms—suggests that similar structures may underpin reasoning in large models.

This provides a concrete research direction: search for the same geometric signatures in frontier LLMs. The diagnostics used here—key orthogonality, QK sharpening, value-manifold structure, and routing stability—offer testable predictions for analyzing pretrained language models.

## 7 Related Work

### 7.1 Bayesian Interpretations of Deep Learning

A long line of work interprets neural networks through a Bayesian lens, from classical analyses of predictive uncertainty [10, 12] to variational or stochastic approximations of posterior inference [3, 7]. Recent papers argue that, in large-data limits, minimizing cross-entropy implicitly targets the Bayesian posterior predictive [15, 16]. These results concern what training *should* produce at the population level. Our contribution is complementary: a controlled setting in which the true posterior is known, memorization is impossible, and one can directly test whether a finite transformer *actually* realizes this Bayesian computation.

## 7.2 In-Context Learning and Algorithmic Generalization

Transformers have been shown to perform algorithmic tasks in context, including arithmetic [6], synthetic induction [5], and more general pattern extrapolation [2, 13]. Behaviorally, these models often resemble Bayesian learners, an observation formalized by recent explanatory theories [15, 16]. However, prior work cannot distinguish true Bayesian computation from learned heuristics or memorized templates, because the ground-truth posterior is unknown for natural language tasks.

Our wind-tunnel methodology solves this identification problem: by constructing tasks with closed-form analytic posteriors and combinatorially large hypothesis spaces, we obtain a direct pointwise comparison between model predictions and Bayes' rule. This moves the discussion from correlation to mechanism.

## 7.3 Mechanistic Interpretability and Attention Geometry

Mechanistic studies of transformers have revealed specialized attention heads for induction, copying, and retrieval [4, 11]. Other work has examined QKV spaces, circuit decomposition, and sparse structures that arise during training [13]. These studies provide qualitative and circuit-level insight into model behaviors.

Our contribution is to link these geometric structures directly to *Bayesian inference* in a setting where the posterior is known. We show that keys form near-orthogonal hypothesis axes, queries sharpen onto feasible hypotheses across depth, and value representations unfurl into a one-dimensional entropy manifold. This connects mechanistic interpretability to probabilistic computation in a rigorous way: the internal geometry needed for Bayesian reasoning becomes directly visible.

## 7.4 Architectural Comparisons

Alternative sequence models—state-space architectures [8, 9], convolutional variants [14], and deep MLPs—often match transformers in perplexity on natural text. But perplexity conflates modeling and inference capability. Our results provide a finer test: whether an architecture can reproduce an analytic Bayesian posterior under strict non-memorization constraints. The capacity-matched MLP controls clarify that attention-based routing is not merely beneficial but *necessary* for Bayesian structure learning in context.

## 7.5 Training Dynamics

Finally, concurrent work analyzes the gradient dynamics that create these structures during training [1]. They show that attention and value updates follow coupled laws that produce a stable routing frame and a progressively refined value manifold. Our empirical findings align with this picture: attention stabilizes early, while value vectors continue to encode the posterior with increasing resolution. Together, these perspectives connect the optimization trajectory to the geometric structure that implements Bayesian inference.

## 8 Limitations and Future Work

Our experiments are intentionally small-scale: they use controlled Bayesian wind tunnels with analytic posteriors, modest vocabulary sizes, and transformers with 2–3M parameters. This regime is what makes mechanistic verification possible, but it naturally abstracts away from the full complexity of natural-language inference. Several limitations therefore remain, which point directly toward future extensions.

*Scale and richness of inference tasks.* Bijections and HMMs capture essential elements of Bayesian computation—discrete elimination and recursive state tracking—but they represent only a narrow

slice of the inference problems encountered by large language models. Future wind tunnels could incorporate richer latent-variable structures, including Kalman filtering, hierarchical Bayesian models, or causal graphical models, all of which have closed-form posteriors and allow precise verification.

*Dimensionality of hypothesis spaces.* Although the hypothesis spaces in both tasks are large enough to prevent memorization, their representational dimensionality is modest (e.g., five hidden states in HMMs). Larger systems with high-dimensional latent variables would test whether the geometric mechanisms we observe—orthogonal hypothesis axes, progressive Q–K sharpening, and value-manifold refinement—scale smoothly with dimensionality.

*Connection to large pretrained models.* Our geometric diagnostics (key orthogonality, score-gradient structure, value manifolds) are testable predictions for frontier LLMs. Whether similar Bayesian manifolds arise in large models trained on natural text remains an open question. Applying these tools directly to pretrained transformer layers is a natural next step and may reveal how approximate Bayesian structure manifests in more complex settings.

*Architectural generality.* The experiments here use standard transformers. It remains unclear whether alternative architectures—state-space models, deep MLPs with more sophisticated gating, or hybrid recurrent-attention systems—can form comparable Bayesian manifolds. Wind-tunnel evaluations could provide a principled benchmark for comparing architectures in terms of inference fidelity rather than perplexity alone.

*Training dynamics and phase transitions.* A notable empirical phenomenon is the frame—precision dissociation: attention maps stabilize early while value manifolds continue to unfurl and refine posterior precision. A systematic study of these phases—how early the frame forms, how quickly precision improves, and how these dynamics depend on depth, width, and data complexity—could lead to a more general theory of representation formation in transformers.

*Towards natural-language wind tunnels.* Ultimately, we aim to understand how the exact Bayesian reasoning demonstrated here relates to the approximate reasoning observed in natural language tasks. Wind tunnels provide a lower bound: they establish that transformers *can* implement Bayesian updates when the problem is well specified. The next challenge is to design controlled tasks embedded within naturalistic language data that preserve analytic structure while introducing real-world ambiguity.

## 9 Conclusion

We introduced Bayesian wind tunnels—controlled experimental settings with analytic posteriors and combinatorially large hypothesis spaces—to test whether transformers genuinely implement Bayesian inference rather than merely mimicking it. Across two fundamentally different inference problems, discrete bijection elimination and sequential state tracking in Hidden Markov Models, small transformers converge to the exact Bayesian posterior with sub-bit calibration error, even at sequence lengths well beyond those seen in training. Capacity-matched MLPs fail catastrophically in both settings, demonstrating that this behaviour arises from the geometry of attention rather than model size or optimization.

Geometric diagnostics provide a unified explanation. Keys form an approximately orthogonal basis over hypotheses; queries progressively align with the feasible region of that basis; and value vectors organize along a low-dimensional manifold parameterized by posterior entropy. Training sculpts this manifold: attention patterns stabilize early, while value representations continue refining posterior precision—a frame—precision dissociation predicted by concurrent gradient-dynamics

analysis. These mechanisms together implement the essential components of Bayesian conditioning: binding, elimination, and refinement, expressed as a sequence of structured linear transformations across depth.

The wind-tunnel regime is intentionally simplified, but it establishes a clear lower bound: if a model cannot implement Bayes in settings where the posterior is known and memorization is impossible, it cannot do so in natural language. Conversely, our results show that transformer geometry is sufficient for exact Bayesian inference when the task permits verification. This provides a principled foundation for studying approximate reasoning in larger models and offers concrete, testable predictions—orthogonal hypothesis axes, progressive Q–K sharpening, and value-manifold structure—for analysing pretrained LLMs.

Transformers succeed here not because they are large, but because their architecture furnishes the right inductive bias: residual streams that carry evolving belief states, attention that routes information selectively, and feed-forward layers that implement local Bayesian updates. Together, these components carve a Bayesian manifold inside the model’s representation space. Understanding how this manifold emerges, scales, and ultimately degrades in real-world language remains an important direction for future work.

## References

- [1] Naman Aggarwal, Vishal Misra, and Siddhartha R. Dalal. 2025. Gradient Dynamics of Attention: How Cross-Entropy Sculpted Bayesian Manifolds. *arXiv:arXiv:2512.XXXXX [cs.LG]* <https://arxiv.org/abs/2512.XXXXX> Paper II of the Bayesian Attention Trilogy.
- [2] Ekin Akyürek and Jacob Andreas. 2022. What Learning Algorithms Does In-Context Learning Learn? Investigations with Linear Models. *arXiv preprint arXiv:2209.11895* (2022).
- [3] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. Weight Uncertainty in Neural Networks. In *Proceedings of the 32nd International Conference on Machine Learning*. 1613–1622.
- [4] 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. 2021. A Mathematical Framework for Transformer Circuits. *Transformer Circuits Thread, Anthropic*. <https://transformer-circuits.pub/2021/framework/index.html>
- [5] Jeffrey L Elman. 1990. Finding structure in time. *Cognitive Science* 14, 2 (1990), 179–211.
- [6] Shivam Garg, Dimitris Tsipras, Percy S. Liang, and Gregory Valiant. 2022. What Can Transformers Learn In-Context? A Case Study of Simple Function Classes. In *Advances in Neural Information Processing Systems*, Vol. 35. 29881–29895.
- [7] Alex Graves. 2011. Practical Variational Inference for Neural Networks. In *Advances in Neural Information Processing Systems*, Vol. 24.
- [8] Albert Gu and Tri Dao. 2023. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *arXiv preprint arXiv:2312.00752* (2023).
- [9] Albert Gu, Karan Goel, and Christopher Ré. 2022. Efficiently Modeling Long Sequences with Structured State Spaces. In *International Conference on Learning Representations*.
- [10] David J. C. MacKay. 1992. A Practical Bayesian Framework for Backpropagation Networks. *Neural Computation* 4, 3 (1992), 448–472.
- [11] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress Measures for Grokking via Mechanistic Interpretability. *arXiv preprint arXiv:2301.05217* (2023).
- [12] Radford M. Neal. 2012. *Bayesian Learning for Neural Networks*. Lecture Notes in Statistics, Vol. 118. Springer.
- [13] 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, et al. 2022. In-Context Learning and Induction Heads. *Transformer Circuits Thread, Anthropic*. <https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html>
- [14] Michael Poli, Stefano Massaroli, et al. 2023. Hyena Hierarchy: Towards Larger Convolutional Language Models. In *International Conference on Machine Learning*.
- [15] Johannes von Oswald, Christian Henning, Adrià Garriga-Alonso, Massimo Caccia, Frederik Träuble, Benjamin F. Grewe, Bernhard Schölkopf, Claudia Clopath, and Johanni Brea. 2023. Transformers as Meta-Learners for Bayesian Inference. *arXiv preprint arXiv:2305.14034* (2023).

- [16] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An Explanation of In-Context Learning as Implicit Bayesian Inference. In *International Conference on Learning Representations*.