TRACING INFORMATION PROCESSING IN TRANSFORMERS WITH STOCHASTIC NORMALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Representational similarity is foundational to analyses of deep networks, yet distances between point-valued representations are not intrinsically tied to downstream function: nearby states may produce different behaviors, while distant states may behave similarly. We instead give representations volume, turning similarity into statistical distinguishability. Overlapping stochastic representations necessarily induce overlapping downstream distributions, grounding latent comparison in model function and bringing it under information-theoretic tools such as the data-processing inequality. We realize this idea in pretrained transformers through a light-touch modification to normalization: at each residual-stream read, normalize the state, add isotropic Gaussian noise, and renormalize. During distillation fine-tuning, one learned allocation parameter per read distributes a fixed global rate budget across the processing stack. The resulting model can be viewed as transformer blocks reading the residual stream with learned finite precision. This stochastic construction supports complementary views of transformer computation: tracing which input distinctions remain accessible to downstream components, and tracking how variability introduced at different locations propagates and mixes through the network. Across vision and language transformers, these views reveal structured, component-specific patterns of information preservation and processing. Together, our results establish finite-precision stochastic representations as a functionally grounded lens on transformer computation.

1 INTRODUCTION

Deep networks are ordinarily trained on point-valued representations: each example constrains the computation at a measure-zero set of internal states, while behavior in the surrounding representational volume is left implicit. Many interpretability methods impose geometric notions of similarity on point-valued representations – for example, through pairwise similarity measures (Kornblith et al., 2019) or the reconstruction distortion used to fit sparse surrogate models (Huben et al., 2024; Bricken et al., 2023) – even though this geometry is not directly grounded by the model's functional objective. The disconnect between geometry and functionality when assessing point-based representations has been increasingly recognized in recent years (Ding et al., 2021; Cui et al., 2022; Davari et al., 2023).

When point-valued representations are replaced by stochastic ones, they acquire volume: behavior over neighborhoods of internal states becomes part of the training objective, coupling representational geometry to model functionality. The resulting stochastic representations define explicit channels through which quantities such as information transmission and pairwise distinguishability can be measured. Without a constraint on representation scale, however, a model can trivially overcome stochasticity by increasing the separation of its representations, recovering an effectively deterministic channel. Prior approaches therefore impose explicit capacity constraints, often through learned encoder-decoder bottlenecks (Kingma & Welling, 2014; Alemi et al., 2017; Murphy, 2026). Doing so at many internal locations requires substantial architectural modification.

Here, we propose a light-touch modification to transformer architectures that exploits the intrinsic scale constraint imposed by LayerNorm (Ba et al., 2016) or RMSNorm (Zhang & Sennrich, 2019). Additive noise is applied to every normalization step in a transformer, before the learned affine transformation, with one optimized magnitude per location such that only $\sim 2L$ additional scalar parameters are introduced. In pre-norm transformers, each modification can be interpreted as a

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

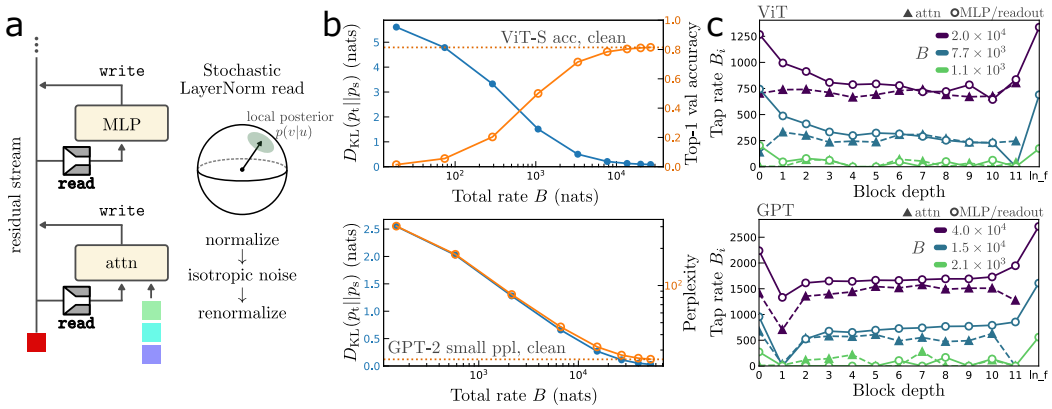

Figure 1: **Injecting stochasticity into residual stream reads. (a)** We add noise to every location in the network where the residual stream is read. The noise is injected inside the LayerNorm operation, which maps representations to the surface of a hypersphere. **(b)** For fine-tuned ViT-S (top) and GPT-2 small (bottom), the imposed read rate B sets the distillation loss relative to the base (deterministic) model, $D_{\mathrm{KL}}(\text{teacher}||\text{student})$, as well as the modality-specific validation performance. **(c)** Per-tap rates provide a macroscopic view of prioritized information protection. For each attention block and MLP block, and for the final read before the unembed operation, a single σ is learned during optimization while fine-tuning all model weights.

noisy read of the residual stream (Elhage et al., 2021): every attention and MLP block pays for measurement precision when accessing the representation carried by the residual stream.

Prior bottleneck approaches interpret useful computation upward from a low-information limit (Murphy, 2026). Here, we begin from a pretrained deterministic solution and move toward lower-fidelity internal channels, jointly adapting the model so that stochastic neighborhoods become part of the learned computation. Because the model weights are optimized jointly with the stochastic channels, the resulting neighborhoods are not merely tolerated by the pretrained computation; they are incorporated into the computation itself.

In experiments on ViT-S and GPT-2 small, we use the fine-tuned models as approximations to their pretrained bases, enabling information-theoretic analysis of internal processing. Across vision and language, we trace how distinctions between perturbed inputs are preserved, propagated, and selectively accessed across depth and attention components. In GPT-2, the most sensitive key projections include several high-induction heads, as well as heads whose selectivity is weak or absent in canonical attention-pattern statistics. Stochastic normalization thus provides a functionally grounded view of transformer information processing that complements methods based on the geometry of point-valued representations.

2 Method

A transformer processes input elements x_i through distinct residual streams, whose representations u_i^l are updated by successive attention and multilayer perceptron (MLP) sublayers. Attention transmits information between streams, whereas the MLP transforms each stream independently. In a pre-norm transformer, every sublayer begins by *reading* its residual stream through a normalization operation. We introduce stochasticity at these reads.

In the following, we restrict our focus to LayerNorm (Ba et al., 2016), though the construction is nearly identical for RMSNorm (Zhang & Sennrich, 2019), which does not mean-center. LayerNorm acts as $\mathrm{LN}(u) = \gamma \odot n_\epsilon(u) + \beta$, where $n_\epsilon(u) = (u - \langle u \rangle)/\sqrt{\mathrm{Var}(u) + \epsilon}$ and γ and β are learned affine parameters (Ba et al., 2016). Because ϵ serves only as a numerical safeguard, we set it to zero in the geometric analysis. The normalized state then has component mean zero and component variance one, implying a Euclidean norm of $\sqrt{d}$. LayerNorm therefore maps representations to the $(d-2)$-sphere $\mathcal{M}_d = z \in \mathbb{R}^d : \langle z \rangle = 0,\ \|z\|_2 = \sqrt{d}$.

We describe the modification at a single tap, suppressing stream and tap indices except on the tap-dependent noise scale σ_t. Given residual-stream state u, we compute $z = n_\epsilon(u)$, draw $\eta \sim \mathcal{N}(0, \sigma_t^2 I)$, and set $v = n_\epsilon(z + \eta)$. The usual LayerNorm affine transformation is then applied to v before the attention or MLP sublayer. We apply this construction at every residual read, with independent noise draws and a separate noise scale for each tap.

For a particular input and realization of all upstream noise, u and z are individual points. Let U denote the raw residual-stream state when viewed as a random variable over inputs and upstream noise, let $Z = n_\epsilon(U)$ be its normalized form, and let V be the output of the local stochastic channel. By the data-processing inequality, no subsequent computation using this read can contain more information about U, through this read, than is available in V. We quantify this fundamental local limitation using Shannon mutual information, $I(U; V)$ (Cover & Thomas, 1991). Because Z is a deterministic function of U and the channel depends on U only through Z, $I(U; V) = I(Z; V)$.

The maximum-entropy distribution on $\mathcal{M}_d$, relative to its natural surface measure, is the uniform distribution. Letting A_d denote the surface area of $\mathcal{M}_d$, its entropy is $\log A_d$. The transmitted information can therefore be upper-bounded as $I(U; V) = h(V) - h(V \mid U) \leq \log A_d - h(V \mid U) \equiv R_d(\sigma_t)$. Rotational symmetry makes the conditional entropy, and hence the rate bound $R_d(\sigma_t)$, a function only of the noise scale and representation dimension. The gap in the bound is $R_d(\sigma_t) - I(U; V) = D_{\mathrm{KL}}\left(p_V \,\|\, \mathrm{Unif}(\mathcal{M}_d)\right)$, where p_V is the marginal distribution of noisy reads induced by the distribution of U and the local noise. The bound is tight when the measured states are uniform over the LayerNorm sphere.

A global hyperparameter B fixes the total rate bound across all taps. Per-tap rates are learned subject to this fixed total, with allocation logits a_t defining $B_t = B\,\mathrm{softmax}(a)_t$. For a given representation dimension, we precompute the monotonic relationship between σ and $R_d(\sigma)$, approximated using a matched vMF channel (Appx. A,C) and use this lookup table to convert each allocated rate into its corresponding noise scale.

We install these stochastic reads in a pretrained model and fine-tune the resulting student by distillation from the original deterministic model. Specifically, we minimize $\mathcal{L}_{\mathrm{distill}} = \mathbb{E}_x\left[D_{\mathrm{KL}}\left(p_{\mathrm{teacher}}(\cdot \mid x) \,\|\, p_{\mathrm{student}}(\cdot \mid x)\right)\right]$. Unlike information-restriction methods that impose a soft rate penalty (Tishby et al., 2000; Murphy, 2026), our objective contains no rate-loss term. The total rate upper bound is instead enforced directly through the fixed allocation constraint and the calibrated conversion from rates to noise scales.

For paired stochastic computations, we use common random numbers (Glasserman & Yao, 1992) to couple upstream noise realizations; details and their relationship to effective posterior overlap are given in Appx. A.

2.1 Tracing distinguishability

After optimization, the raw residual-stream state remains point-valued, but it affects the downstream sublayer only through a rate-limited stochastic read. The functionally relevant object at a tap is therefore not the point alone, but the conditional distribution of states that the tap can observe. We refer to this distribution as the tap's *local posterior*.

The implemented channel—isotropic Gaussian perturbation followed by renormalization—induces a projected-normal distribution on the LayerNorm sphere. For tractable evaluation, we approximate this distribution with a von Mises–Fisher (vMF) distribution (Davidson et al., 2018). We rescale the LayerNorm sphere to unit radius for this approximation and omit the rescaling from the notation below. Its effective ambient dimension is $p = d - 1$, since mean-centering removes one degree of freedom. For a normalized residual state with mean direction $\mu(u)$, the local posterior is approximated as

$$p(v \mid u) \approx C_p(\kappa) \exp\left(\kappa\, \mu(u)^\top v\right),$$

where $C_p(\kappa)$ is the vMF normalizing constant and κ controls concentration (Mardia & Jupp, 2009). The correspondence between the Gaussian noise scale σ and vMF concentration κ is calibrated numerically for the relevant dimension and stored in a lookup table. We evaluate the quality of this approximation in Appendix C.

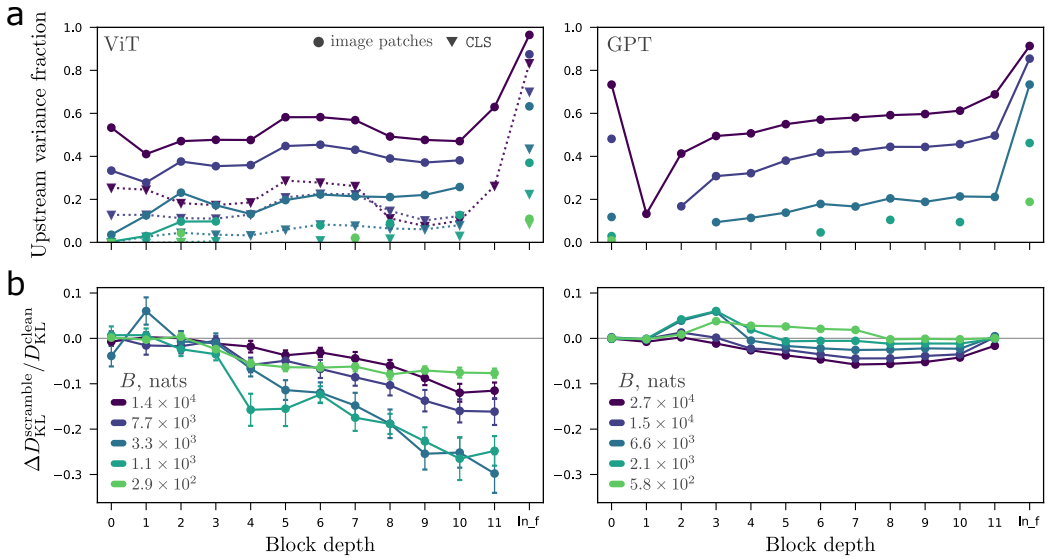

Figure 2: **The nature of the compounded stochasticity. (a)** The upstream variance estimated at each MLP tap, relative to the total (upstream and local posterior added variance), for ViT (left) and GPT2 (right) models. For ViT, the `CLS` stream has different spread than the image patch representations. The markers for collapsed taps in the lower information models are excluded. **(b)** Change in distillation loss as a result of scrambling joint residual stream structure at layer l. For multiple passes on the same input, the residual stream states are scrambled, breaking any joint structure in the effective posterior across streams. The fractional change in the distillation loss D_{KL} shows that scrambling improves the performance of ViT models and has a more muted effect on GPT models.

Given two raw residual-stream states u_i and u_j at the same tap, we quantify the overlap between their local posteriors using the Bhattacharyya coefficient (Bhattacharyya, 1943)

$$\mathrm{BC}(p_i, p_j) = \int_{\mathcal{S}^{p-1}} \sqrt{p(v \mid u_i)\, p(v \mid u_j)}\, dv.$$

The coefficient ranges from zero for disjoint distributions to one for identical distributions; thus, lower BC indicates greater distinguishability. For vMF distributions with a common concentration κ, integrating the geometric mean of their densities yields the closed form

$$\mathrm{BC}(p_i, p_j) = \frac{C_p(\kappa)}{C_p\left(\kappa \sqrt{\frac{1+\mu_i^\top \mu_j}{2}}\right)},$$

where $\mu_i = \mu(u_i)$ and $\mu_j = \mu(u_j)$.

Processing cannot increase distinguishability. In particular, the squared Hellinger distance, $H^2(P, Q) = 1 - \mathrm{BC}(P, Q)$, is an f-divergence and satisfies the data-processing inequality (Ali & Silvey, 1966; Csiszár, 1967). Equivalently, applying a deterministic function or stochastic channel can only increase BC. For an MLP sublayer, which transforms each stream independently, the BC between two output distributions must therefore be at least as large as the BC between their reads. An attention sublayer instead operates jointly on multiple streams: its output distinguishability is bounded by that of the joint distribution of all visible reads, but not necessarily by the marginal distinguishability of any single read.

The linear query, key, and value projections allow us to trace which distinctions in a residual-stream read remain available to each attention head. Under the unit-radius convention adopted above, let

$V \sim p(v \mid u)$ denote the local posterior associated with raw residual state u. For a query, key, or value matrix A, the corresponding head component observes $Y_A = A(\sqrt{d}\,\gamma \odot V + \beta)$. We denote its induced distribution by $p_A(y \mid u)$ and define the projected overlap between two residual states as

$$\mathrm{BC}_A(u_i, u_j) = \mathrm{BC}\left(p_A(\cdot \mid u_i), p_A(\cdot \mid u_j)\right).$$

Because the projection is generally noninvertible, the data-processing inequality gives $\mathrm{BC}_A(u_i, u_j) \geq \mathrm{BC}(p(\cdot \mid u_i), p(\cdot \mid u_j))$: a head component may discard distinctions present in its residual-stream read, but cannot create new ones. Using analytically-derived projected density (Appx. D), we can estimate BC_A via Monte Carlo.

3 Experiments

For both vision and natural language, optimization consisted of fine-tuning a point-based (deterministic) model with solely a distillation loss, $\mathcal{L} = \mathbb{E}_{x\sim p(x)}[D_{\mathrm{KL}}(p_{\text{teacher}}(y|x)||p_{\text{student}}(y|x))]$, with $p(x)$ corresponding to the training data. The noise level was ramped from $\sigma_g = 0.05$ (the uniform-equivalent per-tap noise level defining the budget) to its target value during the first 6 of 25 epochs, equivalently decreasing the rate budget from its near-deterministic initial value. For vision, we fine-tuned ViT-S (Dosovitskiy et al., 2021) from `timm` with `vit_small_patch16_224` weights on ImageNet-1k images (Deng et al., 2009); for language we used Hugging Face's GPT-2 small (Radford et al., 2019) with `gpt2` weights and the OpenWebText (OWT) dataset (Gokaslan & Cohen, 2019). More specifics can be found in Appx. A.

Both vision and language models exhibited a knee near $B \sim 10^4$ nats of total read-rate: above it, behavior remained close to the base model, whereas below it, distillation loss and validation performance steadily degraded (Fig. 1b). This behaviorally anchored transition motivates viewing the rate sweep as a probe of the base model's operating resolution. The optimized per-tap allocation provides a macroscopic view of information processing inside the model (Fig. 1c). For both models, the allocation was relatively flat across depth, with exceptions near the beginning and end; attention taps were generally assigned lower resolution than MLP taps in the same layer. Consistently, the final readout was the most protected. As information was restricted further, a subset of taps reached the minimum realizable rate. In GPT-2, both layer 1 reads and the layer 11 attention read reached the rate floor first ($B = 1.5 \times 10^4$ nats), presumably preserving rate for more functionally important reads.

3.1 Tracing stochastic processing streams

Noise introduced at each read propagates through subsequent blocks, producing compounding variation along the depth of the model. We first characterized this accumulation by decomposing the variance of repeated draws at each MLP tap into locally introduced and upstream-induced components (Fig. 2a). In the ViT, patch streams accumulated substantially more upstream-induced variance than the `CLS` stream, whose upstream variation remained smaller than the noise introduced by its own reads. At the final tap, the precise local read causes inherited variation to account for a correspondingly large fraction of the total variance.

We next probed dependencies across residual streams by independently shuffling each position across a batch of repeated draws. This approximately replaces samples from the joint distribution $p(u_1, \ldots, u_T \mid \vec{x})$ with samples from the product of its position-wise marginals, while preserving both each marginal and the full input sequence $\vec{x}$. The resulting representation combines streams from different upstream noise trajectories and is therefore not an alternative available to the model within a single forward pass.

The effect was strongly model- and depth-dependent. In the ViT, shuffling reduced the distillation loss, and increasingly so at later depths. In GPT-2, early shuffling produced small, budget-dependent degradations, whereas later shuffling was mildly beneficial. One possible explanation lies in the interaction between attention structure and data modality. The ViT aggregates many partially redundant noisy measurements, for which removing cross-stream noise dependence can improve aggregation. Causal language modeling instead builds directed, nested dependencies among non-redundant prefix representations, making coherent stochastic trajectories particularly important early in processing.

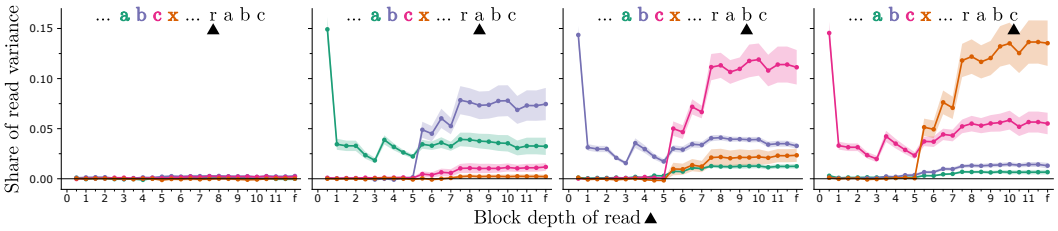

Figure 3: **Variance decomposition, induction task.** In a simple induction setup, a sequence of randomly sampled tokens repeats three in a row (marked `a b c`) after a gap of 64. The fraction of the variance at a read, calculated as the trace of the covariance matrix over samples, can be decomposed by origin. The black triangle indicates the read location, where the variance is measured, and then the colors indicate the share arising from each of the earlier streams (e.g., green for the first occurrence of `a`, purple for `b`, and so on). Shaded regions are SEM over 8 random samples that share the repeated structure.

Finally, the injected stochasticity provides endogenous tracers of how information moves between token streams. In a synthetic induction sequence, ...`abcx`...`rabc`, we decompose the variance at each repeated-token read according to its upstream source for the GPT-2 model with total budget $B = 1.5 \times 10^4$ nats (Fig. 3). Before the repeat, the earlier `abc` contribute negligibly. When `a` reappears, variation from its first occurrence is immediately visible by the block 0 mlp read; at block 5, variation from the original `b` begins to enter as well, matching the next-token retrieval structure characteristic of induction. The source-specific pattern continues through the subsequent repeated tokens. Thus, repeated samples of the stochastic model expose position-to-position processing along native forward passes, without activation replacement or other off-path interventions.

3.2 Vision resolution: Response curves to graded augmentations

In a ViT, the `CLS` stream is central to prediction yet identical for all images at initialization, making it a natural location to monitor where image distinctions emerge. We applied continuously graded augmentations and measured the distinguishability of the resulting `CLS` reads from those of the unmodified image. In Fig. 4a, progressively blurring an example image produces a corresponding decay in BC across depth.

For 200 images from the ImageNet-1k validation set, we applied four augmentations: blurring, hue rotation, additive Gaussian noise, and fading via contrast reduction. Indexing their magnitudes by the total Euclidean pixel displacement $||\Delta x||_2$ allows their effects to be compared at matched input-space distances. At both the layer 6 MLP and the final readout, the model was most sensitive to blurring and least sensitive to fading (Fig. 4b). The relative ordering was broadly preserved at the final readout, while all augmentations became more distinguishable in absolute terms. These response curves trace augmentation-specific slices through a fuzzy resolution neighborhood around each image, relating graded changes in input space to when they become resolved during processing.

3.3 Natural language: Attention head selectivity

In contrast to vision, textual perturbations are discrete in nature, and there is no single reference stream to compare across examples. To monitor the emergence of distinguishability, we perturbed a text sequence at a single location and left the remainder identical. An example is shown in Fig. 5, where the pair of sentences `A {stranger/visitor} suddenly appeared beside the tall iron front gate` differs only in the second token position. The processing of both sentences cannot differ before the location of the perturbation, and the location of the perturbation is trivially distinguishable from the outset. All subsequent tokens' streams initialize at identical embeddings, allowing us to monitor the distinguishability between the pairs as a function of depth in processing. The block-level taps show a gradually increasing distinguishability (quantified via increasing Bhattacharyya distance, $\mathrm{BD} = -\log \mathrm{BC}$) with depth. For the model with the

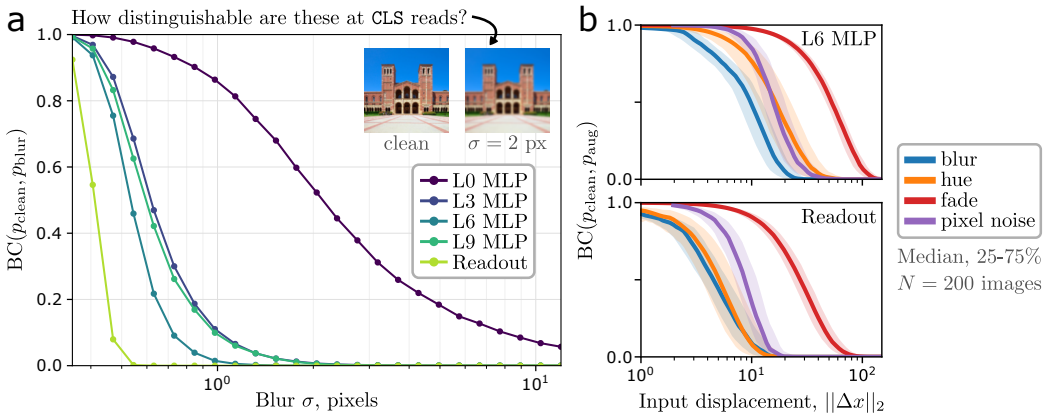

Figure 4: **Distinguishability zone via continuous augmentations (a)** For the ViT-S with $B = 3.3 \times 10^3$ nats total budget, the BC of an image and its blurred version smoothly decreases—indicating increasing distinguishability—with the blur magnitude, and varies significantly across depth. **(b)** The model is more sensitive to some augmentations than others, shown here over 200 images via median curves with interquartile range shaded.

larger information budget ($B = 1.5 \times 10^4$ nats), the distinguishability is high early on in processing, whereas the other model processes the pair more similarly until the last layers.

In Fig. 5 (right), the Q and K projections in layer 5 show head-specific sensitivity to the perturbation. Because these projections are noninvertible, they discard information; the relevant question is therefore whether a given distinction is preserved in the subspace selected by each head, potentially revealing functional specialization (Olsson et al., 2022; Wang et al., 2023).

We generated 80 minimally perturbed pairs by replacing nouns in OpenWebText samples with randomly selected nouns from the same category. We retained swaps producing at most 50 nats of Bhattacharyya distance at the layer 5 attention read immediately following the perturbation. This controls perturbation severity: larger changes produce widespread distinguishability and obscure head-to-head selectivity.

Figure 6 shows the fraction of upstream distinguishability retained by each K projection in the GPT-2 model operating at $B = 1.5 \times 10^4$ nats. Random 64-dimensional projections preserve a nonzero fraction and provide a baseline near which most learned K projections operate. Seven heads in the middle layers stand out above this baseline. Five coincide with high-induction-score heads in the GPT-2 Small induction mosaic of Nanda (2022). Among the remaining two, `L5H8` was described as a "fuzzy induction head" in the canonical indirect object identification analysis; `L5H5` and `L6H9` were identified there as canonical induction heads (Wang et al., 2023). Thus, six of the seven K-selective heads have an independently observed connection to induction.

The correspondence is also reflected, though incompletely, in prefix-matching scores computed for the stochastic model from attention maps on repeated random-token sequences (Olsson et al., 2022). Attention maps indicate which positions a head weights, whereas K distinguishability identifies which perturbation-induced distinctions are exposed to its attention routing. Notably, `L5H8` and `L4H7` retain strong K selectivity despite having nearly zero prefix-matching scores. The former's previous characterization as a fuzzy induction head illustrates how distinction preservation can reveal induction-adjacent structure that is weak or absent in the canonical attention statistic. For all highlighted heads except `L4H7`, this selectivity is specific to K: the corresponding Q and V projections remain near the random-projection baseline (Fig. 6b). `L4H7` instead preserves unusually high distinguishability across Q, K, and V, suggesting broader sensitivity to the perturbation. Finally, for the layer 5 heads, excess K selectivity decays with distance from the perturbed token toward the network-wide baseline (Fig. 6c).

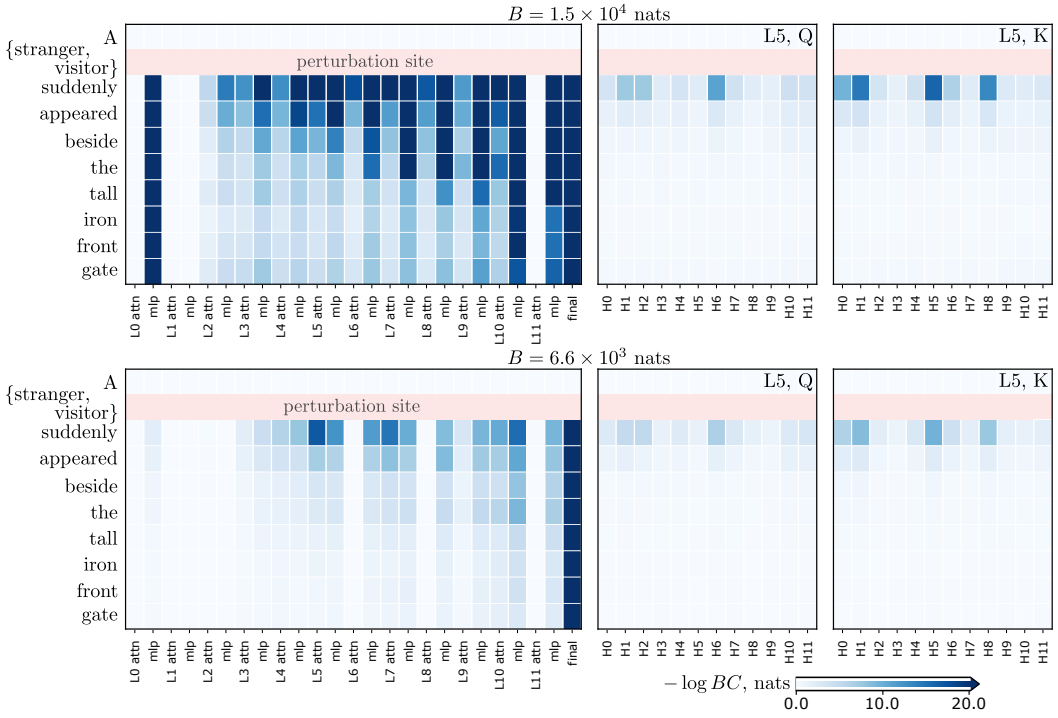

Figure 5: **Perturbation distinguishability across the sequence and in specific heads.** For the sentence shown at left, the perturbation consists of switching the *stranger* token for *visitor*. After the perturbation, the distinguishability (here displayed by Bhattacharyya distance, $-\log BC$) is location-dependent. Attention heads' projections tap into the distinguishability (*right*); shown are the query (Q) and key (K) projections for the twelve heads of layer 5.

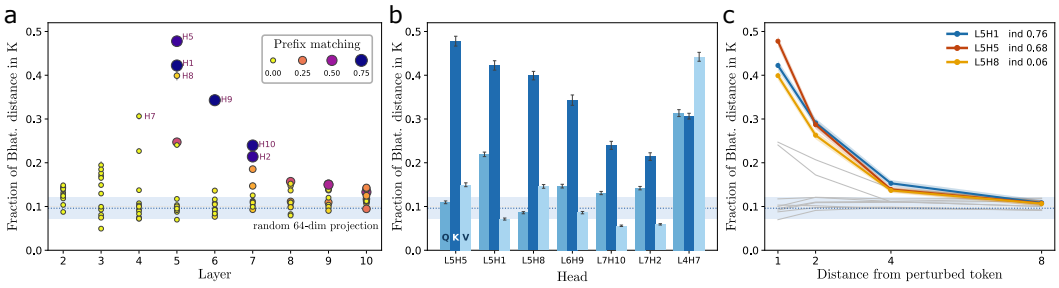

Figure 6: **Key sensitivity to single-token perturbations.** **(a)** The fraction of the Bhattacharyya distance preserved from each attention read into its subsequent K projections is shown for all heads between layers 2 and 10, for the $B = 1.5 \times 10^4$ nats GPT-2 model. For reference, random 64-dimensional projections preserve around 0.1 of the upstream Bhattacharyya distance (shaded band is 5-95% spread). Heads' key projections are colored and sized according to the head's prefix-matching score (Olsson et al., 2022). **(b)** The same fraction for the top heads in **a**, evaluated at the query (Q) and value (V) projections as well. **(c)** The same fraction for all heads in layer 5 as a function of distance from the perturbation.

4 Related Work

Representational similarity analysis characterizes neural or learned representations through the geometry of responses across inputs (Kriegeskorte et al., 2008; Kriegeskorte & Wei, 2021; Kornblith

et al., 2019). However, geometric similarity measures need not agree, and their relationship to functional behavior depends on the invariances built into the measure (Ding et al., 2021).

Psychophysics has long characterized perceptual resolution through just-noticeable differences (Robinson, 2010). Signal detection theory subsequently recast discrimination in terms of overlapping distributions of internal responses, making distinguishability graded rather than binary (Green & Swets, 1966). Here we have taken the same premise — that states accessible to a noisy physical information-processing system are better characterized by distributions than by perfectly resolved points (Pohl et al., 2026) — to engineer a modified system whose distinguishability volumes are directly readable. Designed perturbations therefore serve as standardized probes of distinguishability across models, analogous to the use of shared stimulus sets in RSA and model-discriminating "controversial stimuli" in systems neuroscience (Kriegeskorte et al., 2008; Golan et al., 2020). In contrast to systems neuroscience, we can optimize systems at different resolution scales (Fig. 1) and directly access their computation for probing and intervention (Fig. 6).

There is a broader theme of using optimized information loss as a route to understanding. Information-bottleneck methods optimize task-relevant representations under constraints on retained information (Tishby et al., 2000; Alemi et al., 2017; Achille & Soatto, 2018). Information Bottleneck Attribution instead installs a post-hoc, spatially allocated bottleneck to identify information required for a particular prediction (Schulz et al., 2020; Hong et al., 2025). Related probing work uses minimum description length to quantify not only whether a property is decodable, but how economically it can be extracted from a representation (Voita & Titov, 2020).

Complementary theoretical work studies how bounded numerical precision constrains the representational and computational capacity of neural language models (Merrill & Sabharwal, 2023; Nowak et al., 2024; Li & Cotterell, 2026b;a). Our notion of precision is instead stochastic and local to residual-stream reads, providing a tool for analyzing information processing in pretrained models rather than bounding expressivity.

5 Discussion

For unconstrained point representations processed by an expressive function class, geometric proximity alone provides no general guarantee of similar downstream behavior. We introduce stochasticity during optimization so that transformer blocks instead learn to operate on finite-resolution distributions of residual-stream reads. This makes the stochastic variation informative: overlap between read distributions gives an operational notion of distinguishability, while repeated stochastic trajectories expose how variation propagates and mixes through the network. These views remain partial. High overlap constrains two inputs to be processed similarly, whereas low overlap says only that a distinction is available; likewise, propagated variance establishes dependency without specifying the computation performed.

The construction should not be taken to imply that an unmodified transformer possesses an intrinsic probabilistic geometry. Rather, it produces a controlled, finite-precision extension of the pretrained model: with few added parameters and modest fine-tuning, the model can remain within hundredths of a nat of the original behavior while every residual-stream read is stochastic and rate-limited. The lower-rate boundary of this behaviorally anchored regime thereby probes the model's operating resolution.

Several approximations remain. The information-rate expression is an upper bound whose tightness depends on the distribution of states over the LayerNorm sphere. Moreover, our vMF analysis describes local posteriors, while upstream stochasticity induces effective posteriors that are generally broader mixtures. Common random numbers enable lower-variance paired comparisons but do not eliminate the difficulty of characterizing these full distributions.

Our experiments on ViT-S and GPT-2 small should therefore be viewed as a proof of concept. Further work should test whether the observed patterns persist at larger scales, better characterize effective posteriors and joint stream structure, and determine when conclusions from rate-limited models transfer to their unconstrained counterparts. More broadly, stochastic normalization turns internal representations from isolated points into measurable processing distributions, revealing both which distinctions remain available and how variation is routed through computation. It does not provide a complete mechanistic account, but offers a functionally grounded substrate for more detailed analysis.

AI USE STATEMENT

In this work, we used generative AI tools for literature scans, to implement the fine-tuning scripts, derive the BC estimation method in the attention-head projected subspaces, and to run numerous analyses, most of which did not make it into the submission. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Alessandro Achille and Stefano Soatto. Information dropout: Learning optimal representations through noisy computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40 (12):2897–2905, 2018.

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=HyxQzBceg.

Syed Mumtaz Ali and Samuel D Silvey. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1): 131–142, 1966.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Anil Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distribution. *Bulletin of the Calcutta Mathematical Society*, 35:99–110, 1943.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Thomas M Cover and Joy A Thomas. *Elements of information theory*, volume 1. wiley, 1991.

Imre Csiszár. On information-type measure of difference of probability distributions and indirect observations. *Studia Sci. Math. Hungar.*, 2:299–318, 1967.

Tianyu Cui, Yogesh Kumar, Pekka Marttinen, and Samuel Kaski. Deconfounded representation similarity for comparison of neural networks. *Advances in Neural Information Processing Systems*, 35:19138–19151, 2022.

MohammadReza Davari, Stefan Horoi, Amine Natik, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. Reliability of CKA as a similarity measure in deep learning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=8HRvyxc606.

Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. Hyperspherical variational auto-encoders. *arXiv preprint arXiv:1804.00891*, 2018.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. Grounding representation similarity through statistical testing. *Advances in neural information processing systems*, 34:1556–1568, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

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.

Paul Glasserman and David D Yao. Some guidelines and guarantees for common random numbers. *Management Science*, 38(6):884–908, 1992.

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019.

Tal Golan, Prashant C Raju, and Nikolaus Kriegeskorte. Controversial stimuli: Pitting neural networks against each other as models of human cognition. *Proceedings of the National Academy of Sciences*, 117(47):29330–29337, 2020.

David Marvin Green and John A Swets. *Signal detection theory and psychophysics*, volume 1. Wiley New York, 1966.

Jung-Ho Hong, Ho-Joong Kim, Kyu-Sung Jeon, and Seong-Whan Lee. Comprehensive information bottleneck for unveiling universal attribution to interpret vision transformers. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25166–25175, 2025. doi: 10.1109/CVPR52734.2025.02343.

Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=F76bwRSLeK.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMLR, 2019.

Nikolaus Kriegeskorte and Xue-Xin Wei. Neural tuning and representational geometry. *Nature reviews neuroscience*, 22(11):703–718, 2021.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A Bandettini. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:249, 2008.

Jiaoda Li and Ryan Cotterell. Characterizing the expressivity of local attention in transformers. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 37485–37507, 2026a.

Jiaoda Li and Ryan Cotterell. Characterizing the expressivity of fixed-precision transformer language models. *Advances in Neural Information Processing Systems*, 38:159510–159554, 2026b.

Kanti V Mardia and Peter E Jupp. *Directional statistics*. John Wiley & Sons, 2009.

William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545, 2023.

Kieran A Murphy. From independent patches to coordinated attention: Controlling information flow in vision transformers. *arXiv preprint arXiv:2602.04784*, 2026. URL https://arxiv.org/abs/2602.04784.

Neel Nanda. Induction mosaic. https://www.neelnanda.io/mosaic, 2022. Accessed: 2026-08-31.

Franz Nowak, Anej Svete, Alexandra Butoi, and Ryan Cotterell. On the representational capacity of neural language models with chain-of-thought reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12510–12548, 2024.

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.

Stephan Pohl, Edgar Y Walker, David L Barack, Jennifer Lee, Rachel N Denison, Ned Block, Florent Meyniel, and Wei Ji Ma. Clarifying the conceptual dimensions of representation in neuroscience. *Nature Reviews Neuroscience*, 27(5):357–372, 2026.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

George H. Robinson. *Fechner's Law*, pp. 1–1. John Wiley & Sons, Ltd, 2010. ISBN 9780470479216. doi: https://doi.org/10.1002/9780470479216.corpsy0350. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470479216.corpsy0350.

Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1xWh1rYwB.

Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 183–196, 2020.

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.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in neural information processing systems*, 32, 2019.

Algorithm 1 Noisy LayerNorm Read

(Replaces every `nn.LayerNorm`).

function NOISYLN($x;\ \gamma, \beta, \sigma$)

$\hat{x} \leftarrow \text{NORMALIZE}(x)$, $\text{NORMALIZE}(u) := \dfrac{u - \text{mean}(u)}{\sqrt{\text{var}(u) + \epsilon}}$ ▷ statistics over the width axis

$z \sim \mathcal{N}(0, I_D)$ ▷ drawn independently per token/patch position and per sample

$\hat{x} \leftarrow \text{NORMALIZE}(\hat{x} + \sigma z)$ ▷ re-normalise: the read stays on $\mathcal{S}^{D-2}$

return $\gamma \odot \hat{x} + \beta$ ▷ learned affine applied *after* the channel

end function

A IMPLEMENTATION

Code is included in the supplemental material. Excepting the analysis, the crux is the simple modification to LayerNorm, presented in Alg. 1.

A.1 THE CHANNEL AND ITS CALIBRATION

Every `nn.LayerNorm` in the network is replaced by a noisy read: the input is normalised, isotropic Gaussian noise of scale σ is added, the result is re-normalised, and only then are the learned gain and bias applied. Because a LayerNorm output has zero mean and unit RMS, each read lives on the sphere $\mathcal{S}^{D-2}$ inside the mean-zero hyperplane of $\mathbb{R}^D$. We approximate each read as $\text{vMF}(\mu, \kappa)$ and take the concentration κ as the rate knob:

$$\text{rate}(\kappa) = \text{KL}(\text{vMF}(\mu, \kappa) \,\|\, \text{Unif}) = \kappa\, A_p(\kappa) + \log C_p(\kappa) - \log C_p(0), \tag{1}$$

in closed form, and the sampling scale is matched to it through the mean resultant length $\rho = A_p(\kappa) = I_{p/2}(\kappa)/I_{p/2-1}(\kappa)$ by $\sigma = \sqrt{1/\rho^2 - 1}$, so that the mean read-cosine of the add-Gaussian-then-renormalise channel equals ρ (verified numerically against the analytic ρ, and the closed-form rate against a Wood/Ulrich Monte-Carlo estimate). This matching is what makes the channel reparameterisation-clean: gradients flow to both the weights and the rate allocation with no rejection sampler in the loop. Training uses a precomputed strictly-monotone 4000-point geometric grid in κ (up to 10^6) with the rate and σ columns exposed as differentiable one-dimensional linear interpolants.

Both ends of that grid are set by floating-point limits. All Bessel evaluations use the exponentially scaled modified Bessel function of the first kind, $\widetilde{I}_v(z) = I_v(z)\, e^{-z}$ for real $z > 0$ (SciPy's `scipy.special.ive`): the unscaled I_v overflows to infinity at the top of the grid for every p we use, whereas the scaling cancels in the ratio A_p and enters the log-normaliser additively as $\log I_v(\kappa) = \log \widetilde{I}_v(\kappa) + \kappa$. At the bottom of the grid the opposite failure appears, and it is *not* an artefact of the scaling: in double precision $I_{p/2-1}(\kappa)$ underflows to exactly zero for $\kappa \lesssim 20$ at $p = 767$, scaled or not, which makes $\rho = A_p(\kappa)$ a $0/0$. The ViT runs therefore keep the default floor $\kappa_{\min} = 5$, which stays representable for $p \leq 383$, while for GPT-2 the floor is found by bisection as the smallest κ whose $\widetilde{I}_v$ clears a conservative 10^{-250}, giving $\kappa_{\min} = 76.3$ at $p = 767$. Since the interpolants clamp their argument to the grid, this floor also caps the largest representable per-tap noise: $\sigma \leq 76.6$ at $p = 383$ and 10.1 at $p = 767$. The cap is reached in practice, as the learned allocation drives the most heavily compressed taps down onto it: for GPT-2, $5/25$ taps sit at the cap at $\sigma_g = 1$, rising to $13/25$ at $\sigma_g = 2$ and $22/25$ at $\sigma_g = 8$ (ViT-Small, $1/25$ and $18/25$ at $\sigma_g = 1$ and 8).

A.2 RATE BUDGET AND ALLOCATION

A run is specified by a single dimension-free knob, the uniform-equivalent noise level σ_g. This is converted to κ_g, then to a per-tap rate, and the *total* budget is fixed at $B = n \cdot \text{rate}(\kappa_g)$ over the n taps; the network learns only how to spend it, via unconstrained logits a passed through a softmax,

$$r_i = B \cdot \text{softmax}(a)_i, \qquad \textstyle\sum_{i=1}^{n} r_i = B \ \text{ by construction.} \tag{2}$$

Conserving the budget in *rate* (rather than penalising it) removes the degenerate solution where the model compresses every tap and pays the penalty once, and no penalty term appears in the loss at

all: the distillation KL alone drives a water-filling allocation. Both architectures contain 25 taps, although the rate associated with a fixed σ_g depends on the representation dimension.

A.3 THE NOISE RAMP

We ramp the noise budget: the run starts at a near-clean $\sigma_g = 0.05$ and interpolates to the target over the first 6 of 25 epochs ($\sim$24% of training), updated **every optimiser step** rather than per epoch. Two interpolation shapes are implemented: linear in $\log$-rate and linear in $\log \sigma_g$. The GPT-2 sweep used the former, and ViT the latter. Everything else is shared: AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$) with two parameter groups — network weights at the run's learning rate with weight decay 5×10^{-3}, allocation logits at 6×10^{-3} with **no** decay — cosine decay after a 0.2-epoch linear warmup, global grad-norm clipping at 1.0 over weights and logits jointly, `bfloat16` autocast on A100 MIG partitions. Every run is a *finetune* arm whose loss is $\mathrm{KL}(\text{teacher} \,\|\, \text{noisy student})$ at $T = 1$ against a frozen copy of the pretrained weights, with the student initialised at the teacher; distilling from the clean model rather than from labels keeps the distortion axis measured in the same units across both modalities.

A.3.1 COMMON RANDOM NUMBERS AND VARIANCE DECOMPOSITION

At a given target tap, upstream stochasticity makes the effective posterior a mixture of local posteriors. Let $\epsilon_{1:L}$ denote the Gaussian noise draws at the L upstream taps. The effective posterior for input x is

$$\bar{p}(v \mid x) = \mathbb{E}_{\epsilon_{1:L}}[p(v \mid x, \epsilon_{1:L})].$$

When comparing processing pathways, we couple their evaluations using common random numbers (CRN) (**?**). In particular, noise draws may be shared up to a chosen processing breakpoint and sampled independently thereafter.

For distinguishability comparisons, sharing the complete upstream noise sequence gives

$$\mathrm{BC}(\bar{p}(\cdot \mid x_1), \bar{p}(\cdot \mid x_2)) \geq \mathbb{E}_{\epsilon_{1:L}}[\mathrm{BC}(p(\cdot \mid x_1, \epsilon_{1:L}), p(\cdot \mid x_2, \epsilon_{1:L}))]\,,$$

by joint concavity of the Bhattacharyya coefficient. Thus, fully coupled evaluations provide a lower bound on the overlap between the corresponding effective posteriors.

The same coupling provides a decomposition of stochastic variation. Let Y and Y' denote two realizations of a residual-stream state that share noise up to breakpoint k and use independent noise thereafter. Conditional independence of the suffixes gives

$$\mathrm{Cov}(Y, Y') = \mathrm{Cov}(\mathbb{E}[Y \mid \epsilon_{1:k}])\,.$$

Thus, $\mathrm{TrCov}(Y, Y')$ measures the portion of residual variance attributable to stochasticity shared before the breakpoint. For nested breakpoints, differences between successive cross-covariance traces give the incremental variance associated with each intervening partition; normalizing by the total variance yields the shares reported in Fig. 3.

A.4 INDUCTION SEQUENCE

For Fig. 3, we created simple randomized token sequences in the spirit of Olsson et al. (2022). Tokens are drawn uniformly without replacement from GPT-2 token ids 1000–39999. The first token in the sequence is EOS, followed by 32 + 4 + 64 random tokens. These are followed by a repeat of the first three out of four that follow the initial 32-token padding. For the GPT-2 model with total budget $B = 1.5 \times 10^4$ nats, we sampled more than 5×10^5 passes through the model with common random numbers up to specific breakpoints in the processing. The total variance at a read, given by $\mathrm{Tr}C$, can then be decomposed by the share contributed by earlier partitions.

A.5 GPT-2

GPT-2-small (124M, $D = 768$; `ln_1`/`ln_2` per block plus `ln_f`, giving 25 taps) is finetuned on OpenWebText (Gokaslan & Cohen, 2019). Documents get an explicit EOS appended before packing, so GPT-2's own context-reset convention sits at every document boundary instead of unrelated documents being spliced together; blocks are non-overlapping 512-token windows, which removes

any need for padding or attention masks. Validation is a held-out contiguous 5000-document tail of the same stream. Runs use batch 32×512 tokens with `limit_train_batches` $= 5000$, i.e. 81.9M tokens per epoch and ≈ 2.05B tokens over 25 epochs — under a quarter of one pass, so no example is seen twice. Learning rate $1 \times 10^{-4} \to 1 \times 10^{-6}$ cosine. All three dropout probabilities (residual, embedding, attention) are forced to 0.0: dropout is an uncontrolled second noise source that would sit underneath the vMF channel during training and confound the rate measurement. The loss is a manually shifted cross-entropy, and the distillation KL is chunked in blocks of 128 positions along the sequence — the full $[B, L, V]$ `float32` softmax at vocabulary size 50257 is $\sim$3.3 GB per tensor, enough to decide whether a run fits its MIG slice — with identical `float32` arithmetic, not a precision trade. Evaluation is full-validation perplexity plus next-token top-1 accuracy under the noise (never a clean or mean pass). Note that a GPT-2 "epoch" here is a fixed 5000 optimiser steps, not a pass over the corpus.

A.6 ViT

The vision runs finetune timm's `vit_small_patch16_224` ($D = 384$) on full ImageNet-1k at 224^2. Preprocessing is timm's standard ViT config — bicubic interpolation, `crop_pct` 0.9, mean/std 0.5 — with deliberately *finetuning*-weight augmentation held fixed across the whole sweep so that cross-σ_g and cross-model comparisons are not confounded by an augmentation change: RandAugment `rand-m7-mstd0.5-inc1`, RandomResizedCrop with the scale floor raised to 0.4 (no extreme micro-crops), horizontal flip 0.5, no random erasing, and drop-path 0. Mixup/CutMix are disabled in the distillation arm — the loss is the teacher KL, so soft-target cross-entropy and label smoothing do not apply. Batch 512, learning rate $4 \times 10^{-4} \to 1 \times 10^{-5}$ cosine, 25 epochs (a 50-epoch arm was also run and did not change the picture). Validation accuracy is computed on a fixed 40-batch (20,480-image) subsample of the held-out split, with the channel *on* at the run's learned allocation. An ImageNet epoch here is one full pass over the training set.

B Additional ViT distinguishability

Fig. 7 extends the results of Fig. 4 in the main text to more locations and multiple information levels. As information decreases (σ_g increases), the distinguishability curves shift right toward larger pixel displacements. Along the direction of processing, the order of the augmentations changes: at the block 0 MLP read, pixel noise is the least and hue the most distinguishable of the augmentations at matching overall pixel displacement. From block 3 onward, fade becomes the least distinguishable and blur the most distinguishable.

C Gaussian to von Mises-Fisher approximation

In Fig. 8, we compare the angular spread induced by the noisy LayerNorm operation to that of an approximating vMF distribution. Due to rotational symmetry, the comparison reduces to a single scalar: the angular deviation from the distribution mean, μ.

D Pushforward density under query, key, and value projections

Let

$$\mathcal{H} = \{z \in \mathbb{R}^d : \mathbf{1}^\top z = 0\}$$

denote the mean-zero LayerNorm subspace, and let $p = d-1$ be its dimension. Choose an orthonormal basis $E \in \mathbb{R}^{d \times p}$ for $\mathcal{H}$. Under the unit-radius convention used for the vMF approximation, a stochastic read $V \in \mathcal{H}$ can be written as $V = EX$ for some $X \in \mathcal{S}^{p-1}$. The corresponding unscaled LayerNorm state is $\sqrt{d}EX$.

For a query, key, or value projection A, including the LayerNorm affine parameters γ and β, the projected state is

$$Y_A = A\left(\sqrt{d}\,\gamma \odot EX + \beta\right) = B_A X + b_A,$$

where

$$B_A = \sqrt{d}\,A\,\mathrm{diag}(\gamma)E, \qquad b_A = A\beta.$$

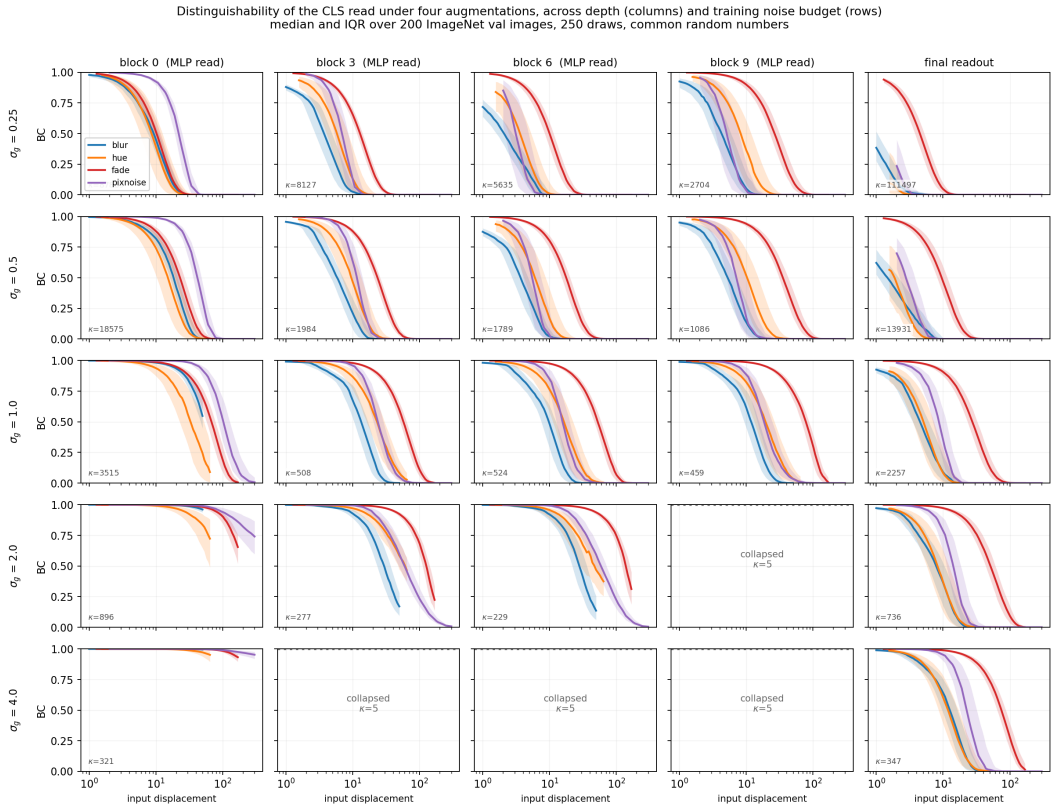

Figure 7: **Distinguishability of augmentations inside ViT-S, across depth and information rate.**

The shift b_A is common to all local posteriors and does not affect their BC.

If B_A is rank deficient, we restrict its codomain to $\mathrm{im}(B_A)$ and express the output in orthonormal coordinates on this image. The resulting matrix has full row rank; we continue to denote it by B_A and let m denote its output dimension. Define

$$S_A = (B_A B_A^\top)^{1/2}, \qquad P_A = S_A^{-1} B_A.$$

Then

$$B_A = S_A P_A, \qquad P_A P_A^\top = I_m.$$

Thus, P_A has orthonormal rows and S_A is invertible. Writing $W_A = P_A X$, we have $Y_A - b_A = S_A W_A$. Translation by b_A and transformation by S_A are both invertible on the support of the projected distributions, so

$$\mathrm{BC}\left(p_{Y_A|u_i}, p_{Y_A|u_j}\right) = \mathrm{BC}\left(p_{W_A|u_i}, p_{W_A|u_j}\right).$$

Because P_A has orthonormal rows and X has unit norm, the support of W_A is the unit ball

$$\mathbb{B}^m = \{w \in \mathbb{R}^m : \|w\|_2 \le 1\}.$$

We next derive the density on this ball. At a fixed tap, consider a local posterior

$$X \sim \mathrm{vMF}_p(\mu, \kappa), \qquad p(x \mid \mu, \kappa) = C_p(\kappa) \exp(\kappa \mu^\top x).$$

Let $q = p - m$. Complete the rows of P_A with an orthonormal basis $D \in \mathbb{R}^{q \times p}$ for its null space, so that the matrix formed by stacking P_A and D is orthogonal. Every point in the spherical preimage of $w \in \mathbb{B}^m$ can then be written as

$$x = P_A^\top w + D^\top \sqrt{1 - \|w\|_2^2}\, \xi, \qquad \xi \in \mathcal{S}^{q-1}.$$

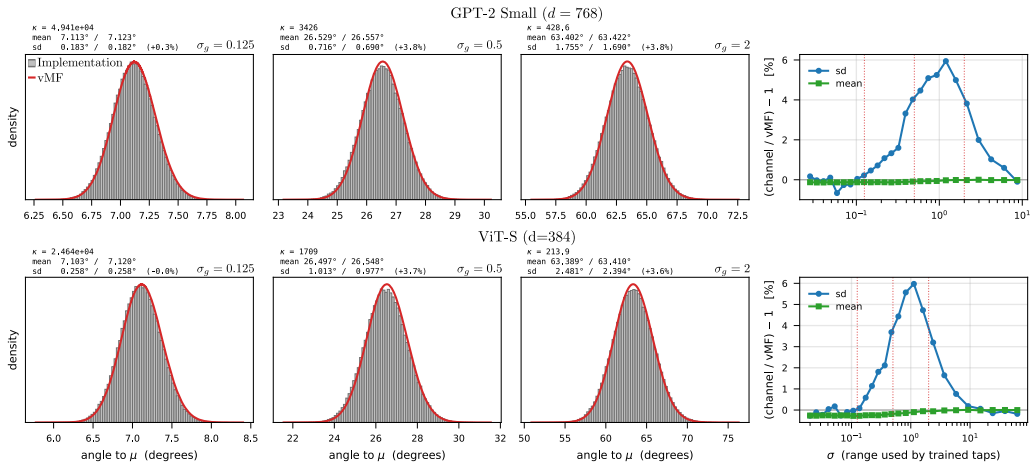

Figure 8: **Calibration of the von Mises-Fisher fit to the noisy LayerNorm channel.** For several levels of noise, we show the histogram of angular deviations from the mean, μ (gray), and the analytic vMF density (red) used in our rate and Bhattacharyya coefficient calculations. The two means and spreads are printed above the three examples at the GPT-2 (top) and ViT-S (bottom) dimensionalities. On the right, the difference in the mean and spread of angular deviations is measured across a range of noise levels.

The spherical surface measure decomposes as

$$d\omega_{p-1}(x) = \left(1 - \|w\|_2^2\right)^{q/2-1} dw\, d\omega_{q-1}(\xi).$$

Define

$$a = P_A\mu, \qquad \rho = \|D\mu\|_2 = \sqrt{1 - \|a\|_2^2}.$$

For a point in the preimage of w,

$$\mu^\top x = a^\top w + \sqrt{1 - \|w\|_2^2}\,(D\mu)^\top \xi.$$

Integrating over ξ and using the vMF normalizing constant in dimension q gives the pushforward density

$$g_\mu^A(w) = \frac{C_p(\kappa)}{C_q\left(\kappa\rho\sqrt{1 - \|w\|_2^2}\right)} \exp(\kappa a^\top w)\left(1 - \|w\|_2^2\right)^{q/2-1}, \qquad w \in \mathbb{B}^m.$$

For local posteriors centered at μ_i and μ_j, let g_i^A and g_j^A denote their respective pushforward densities. Their projected BC is

$$\mathrm{BC}_A(u_i, u_j) = \int_{\mathbb{B}^m} \sqrt{g_i^A(w) g_j^A(w)}\, dw.$$

We estimate this integral using the equally weighted proposal

$$r_A(w) = \frac{1}{2}\left[g_i^A(w) + g_j^A(w)\right].$$

Samples from r_A are obtained by selecting i or j with equal probability, drawing X from the selected vMF local posterior, and computing $W_A = P_A X$. Importance sampling then gives

$$\widehat{\mathrm{BC}}_A(u_i, u_j) = \frac{1}{N}\sum_{n=1}^{N} \frac{2\sqrt{g_i^A(w_n) g_j^A(w_n)}}{g_i^A(w_n) + g_j^A(w_n)}, \qquad w_n \sim r_A.$$

This estimator is unbiased under exact sampling. Moreover, the arithmetic–geometric mean inequality gives

$$0 \leq \frac{2\sqrt{g_i^A(w) g_j^A(w)}}{g_i^A(w) + g_j^A(w)} \leq 1,$$

so every Monte Carlo summand is bounded.