Not Even Long

An Information-Theoretic Perspective on Precision in Neural Networks

Weyl AI Research

January 2026

Abstract

We propose that neural network precision requirements are determined by information content, not numerical convention. The **natural precision** of a tensor is the minimum bits required to preserve task-relevant information; additional precision has no Landauer lower bound for reversible reparameterizations but is computationally wasteful on real hardware. We formalize this using Landauer's principle: only bit erasure has irreducible thermodynamic cost. Representation changes that preserve information—bijections on the realized support—are **epilogues**: gauge transformations with zero Landauer lower bound. This framework explains why quantization "works": it fails when it erases task-relevant bits, and incurs no information-theoretic penalty when it doesn't. We show that successful quantization methods (GPTQ, AWQ, SmoothQuant) implicitly satisfy the injectivity condition by transforming tensors until their realized support fits within the codebook structure.

1 The Approximation-Error Frame

Standard quantization theory treats precision reduction as approximation:

$$\hat{x} = Q(x) \approx x \quad \text{with error } \epsilon = x - \hat{x}$$

This frames quantization as a tradeoff: fewer bits means more error, and error degrades performance. The goal is to minimize $\|\epsilon\|$ subject to bit budget constraints.

This view is incomplete. It treats all bits as equal and all errors as harmful. Neither is true.

2 Information-Theoretic Reframing

Not all bits carry task-relevant information. Consider:

- A weight tensor with values in $\{-1, 0, +1\}$ stored in fp32
- A 16-bit activation whose bottom 8 bits are noise
- An embedding table with 50,000 entries stored in fp64

In each case, the *stored precision* exceeds the *information content*. The excess bits have no Landauer lower bound for erasure—no task-relevant information is lost. (Note: while logical bijections have zero Landauer cost, practical hardware still dissipates energy for memory movement and switching; we distinguish the information-theoretic lower bound from implementation costs.)

Definition 1 (Realized Support)**.** *For a tensor X observed during inference/training, the* ***realized support*** $\text{supp}_{\mathcal{D}}(X)$ *is the set of values actually taken by X under deployment distribution $\mathcal{D}$. This is finite for any finite computation. All probabilities and supports throughout are with respect to $\mathcal{D}$.*

Definition 2 (Task-Aware Equivalence)**.** *Let G be the downstream subnetwork (the remainder of the model after X). For tolerance τ and metric ℓ, define $z \sim_\tau z'$ if $\ell(G(z), G(z')) \leq \tau$. Two values are* ***task-equivalent*** *if collapsing them does not change downstream behavior beyond tolerance.*

Definition 3 (Natural Precision)**.** *The* ***natural precision*** *of tensor X is:*

$$b^*(X) = \lceil \log_2 |supp(X)| \rceil$$

The minimum bits to uniquely identify each realized value. When task-equivalence is considered:

$$b^*_\tau(X) = \lceil \log_2 |supp(X)/\sim_\tau| \rceil$$

The minimum bits to distinguish task-inequivalent values.

Proposition 1. *Any representation change that is a bijection on $supp(X)$ preserves all information about X.*

Proof. A bijection $f : \text{supp}(X) \to \text{supp}(X)$ is invertible. Given $f(x)$, we can recover $x = f^{-1}(f(x))$ exactly. No information is lost. □

3 Epilogues as Gauge Transformations

Definition 4 (Epilogue)**.** *An* ***epilogue*** *is a representation change $E : \mathcal{X} \to \mathcal{Y}$ that is:*

1. *Bijective on the realized support: $E|_{supp(X)}$ is one-to-one*
2. *Potentially non-bijective on the full domain: E may collapse unrealized values*

Epilogues are **gauge transformations**: they change the representation without changing the information content. Like coordinate changes in physics, they affect how we describe the system but not the system itself.

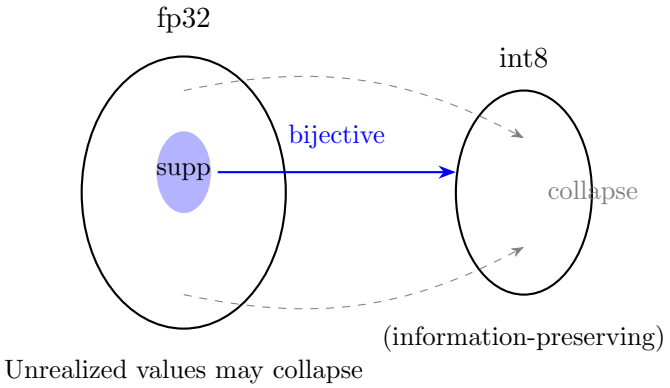

Figure 1: An epilogue is bijective on the realized support (blue) but may collapse unrealized values (gray). Only the bijection matters for information preservation.

4 Landauer's Principle

Landauer's principle [1] states that erasing one bit of information requires dissipating at least $kT \ln 2$ of energy, where k is Boltzmann's constant and T is temperature.

Theorem 1 (Landauer Lower Bound)**.** *The minimum thermodynamic cost of a computation is bounded below by the bits irreversibly erased.*

Reversible operations—bijections—have **no Landauer lower bound**. They can in principle be performed without the irreducible dissipation that accompanies erasure. (Practical implementations still incur switching and memory-movement costs, but these are engineering constraints, not fundamental limits.)

Applied to quantization:

- If quantization is bijective on $\text{supp}(X)$: zero information erased, no Landauer lower bound
- If quantization collapses distinct realized values: information erased, irreducible thermodynamic cost

The question "how much precision do we need?" becomes: "how many bits are in the realized support?"

5 The Codebook Injectivity Condition

Let $d(\cdot, \cdot)$ denote the metric defining nearest-neighbor assignment (typically L_2 per-channel or per-tensor). This induces a **Voronoi partition**: each codebook entry $c \in C$ owns the region of inputs closer to c than to any other entry.

Definition 5 (Codebook)**.** *A **codebook** C for quantization is a finite set of reconstruction values. Quantization maps each input to its nearest codebook entry under metric d.*

Proposition 2 (Injectivity Condition)**.** *Quantization with codebook C is an epilogue if and only if no two values in supp(X) map to the same codebook entry.*

Proof. If distinct $x_1, x_2 \in \text{supp}(X)$ both map to $c \in C$, then $Q(x_1) = Q(x_2) = c$, violating injectivity. Conversely, if all realized values map to distinct codebook entries, $Q|_{\text{supp}(X)}$ is injective and hence bijective onto its image. □

Corollary 1. *Quantization is information-preserving if and only if $|C| \geq |\text{supp}(X)|$ and the Voronoi cells of C each contain at most one realized value.*

Proposition 3 (Relaxed Injectivity)**.** *Strict injectivity is sufficient but not necessary. Quantization preserves task-relevant information at tolerance (τ, δ) if:*

1. ***Probabilistic:*** $\Pr[Q(X_1) = Q(X_2), X_1 \neq X_2] \leq \delta$ *under deployment distribution, or*
2. ***Task-aware:*** *Q merges only pairs (z_1, z_2) with $\ell(G(z_1), G(z_2)) \leq \tau$*

The strict condition ($|C| \geq |\text{supp}_{\mathcal{D}}(X)|$) is the idealization. In practice, successful methods achieve the relaxed conditions by transforming tensors until collisions are rare or task-irrelevant.

Definition 6 (Voronoi Margin)**.** *For $x \in \text{supp}_{\mathcal{D}}(X)$ with nearest codeword c, the **margin** is $\gamma(x) = \min_{c' \neq c}[d(x, c') - d(x, c)]$. If $\gamma(x) > \eta$ for all realized x, injectivity survives perturbations $\|\delta\| \leq \eta/2$. Methods that increase margins are robust to distribution shift and calibration noise.*

Key clarification: The injectivity condition is not assumed to hold *a priori* for raw tensors—it typically does not. Rather, we argue that **successful quantization methods work precisely because they transform tensors until the condition holds**. The methods below shrink, reshape, or factorize tensors until their realized support fits within the codebook's Voronoi structure with adequate margin.

6 Explaining Successful Quantization

Modern quantization methods implicitly achieve the injectivity condition through various transformations:

6.1 SmoothQuant

Xiao et al. [3] observe that activation outliers break quantization. Their fix: migrate magnitude from activations to weights via a diagonal scaling matrix.

Information-theoretic interpretation: Outliers expand the realized support beyond what 8 bits can biject. Smoothing shrinks the support back into the codebook's Voronoi structure. The transformation is an epilogue—bijective on the (smoothed) support.

6.2 AWQ: Activation-Aware Weight Quantization

Lin et al. [4] protect "salient" weights—those with high activation magnitude—from quantization error.

Information-theoretic interpretation: Salient weights have larger effective support (more distinct activation×weight products). AWQ allocates precision to high-information directions, reducing the risk of erasing task-relevant distinctions.

6.3 GPTQ

Frantar et al. [5] use Hessian information to guide quantization, minimizing output perturbation.

Information-theoretic interpretation: The Hessian identifies which weight perturbations affect the output. Weights in flat Hessian directions have low information content (many weight values produce the same output). GPTQ preferentially quantizes low-information weights—implicit Landauer minimization.

6.4 SVDQuant and Nunchaku

Li et al. [6] absorb outliers into low-rank components, enabling 4-bit quantization of the residual.

Information-theoretic interpretation: Outliers are low-rank structure—a few directions carry most of the range. Factoring them out reduces the residual's support to fit in 4-bit codebooks while preserving the full information in the low-rank factors.

7 The Residual Stream as Carrier

In transformers, the residual stream enables aggressive attention quantization:

Information-theoretic interpretation: The residual stream's information passes through without quantization (or at higher precision). The attention branch adds a *delta* whose information

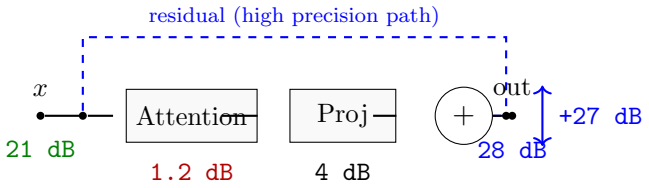

Figure 2: The residual stream carries information through the high-precision path. Attention can collapse to 1.2 dB SNR; the sum recovers to 28 dB. The residual is the carrier wave; attention is the modulation.

content may be lower than its precision suggests. The architecture separates high-information (residual) from low-information (delta) paths.

This is **coherent detection** from RF engineering [7]: the carrier (residual) provides a reference against which the modulation (attention) is decoded. The sum recovers information that neither branch alone contains.

SNR measurement note: The dB values shown are illustrative. When combining signals, SNR arithmetic must be performed in the linear (power) domain: $\text{SNR}_{\text{out}} = P_{\text{signal,combined}}/P_{\text{noise,combined}}$, then converted to dB. The +27 dB recovery occurs because signal adds coherently while uncorrelated quantization noise adds in quadrature.

Measurement protocol: For a layer with residual path R and attention path A:

1. Compute FP32 baseline output y_{fp}; signal power $S = \|y_{\text{fp}}\|_2^2$
2. Quantize only attention $\to y_{\text{qa}}$; noise $N_A = \|y_{\text{qa}} - y_{\text{fp}}\|_2^2$
3. Quantize only residual $\to$ noise N_R; quantize both $\to N_{A+R}$
4. Report linear SNRs $S/N_\bullet$, then convert to dB
5. Verify uncorrelated noise: report $\text{corr}(y_{\text{fp}}, y_{\text{qa}} - y_{\text{fp}})$ and $\text{corr}(y_{\text{fp}}, y_{\text{qr}} - y_{\text{fp}})$

This makes the "carrier/modulation" claim reproducible and testable.

8 Operational Natural Precision

In practice, we cannot enumerate $\text{supp}(X)$ for large tensors. We define an operational proxy:

Definition 7 (Operational Natural Precision)**.**

$$\hat{b}^*_\Delta(X) = H_\Delta(X) + \epsilon$$

*where $H_\Delta(X)$ is the **discretized entropy** at bin width Δ and ϵ accounts for rare events. For continuous X, discretized entropy relates to differential entropy $h(X)$ via $H_\Delta(X) = h(X) - \log_2 \Delta + o(1)$ in the high-resolution regime.*

For well-behaved distributions:

- Gaussian with std σ: $H_\Delta(X) \approx \log_2(\sigma/\Delta) + \frac{1}{2}\log_2(2\pi e) \approx \log_2(\sigma/\Delta) + 2.05$
- Uniform on $[a, b]$: $H_\Delta(X) = \log_2((b-a)/\Delta)$
- Sparse (many zeros): $H_\Delta(X)$ reduced by sparsity entropy

Refinement: For tensors where arbitrarily close values should be considered distinct, an ϵ-covering number $N(\text{supp}(X), \|\cdot\|, \epsilon)$ provides a more robust notion than raw cardinality, connecting to rate-distortion theory [8].

Jacobian proxy for task-relevance: Let $J_G(z)$ be the Jacobian of downstream network G at z. The projected quantization error

$$e_{\text{task}}(z) = \|J_G(z)(Q(z) - z)\|_2$$

measures how much quantization noise propagates to task-relevant outputs. Minimizing $\mathbb{E}[e_{\text{task}}(Z)]$ at fixed bit rate is an operational surrogate for preserving task-relevant information. This connects to GPTQ's Hessian criterion: flat Hessian directions have small $\|J_G\|$ and tolerate coarse quantization.

KL divergence as kernel trick: Computing J_G is expensive. For language models, a practical alternative is the KL divergence between output distributions:

$$D_{\text{KL}}(p_{\text{fp}} \| p_Q) = \sum_v p_{\text{fp}}(v) \log \frac{p_{\text{fp}}(v)}{p_Q(v)}$$

where p_{fp} and p_Q are next-token distributions from the full-precision and quantized models. This captures task-relevant information loss without explicit Jacobian computation:

- $D_{\text{KL}} = 0$: quantization is an epilogue (no task-relevant information lost)
- $D_{\text{KL}} > 0$: some task-relevant distinctions collapsed
- Per-layer KL (with frozen downstream): localizes information loss

This is the same quantity minimized by knowledge distillation, providing a unified view: **distillation teaches a smaller model to be an epilogue of the larger one**.

9 Implications

For quantization: The goal is not "minimize error" but "preserve information." A 4-bit quantization that is bijective on the realized support is perfect. A 16-bit quantization that collapses distinct realized values is lossy.

For mixed precision: Different tensors have different natural precision. Uniform precision wastes bits on low-information tensors and starves high-information ones. Optimal allocation matches precision to information content.

For architecture design: Architectures that separate high-information paths (residuals) from low-information paths (attention deltas) enable aggressive quantization of the latter.

For theory: The continuous/discrete distinction is less important than the information/non-information distinction. A tensor's "precision requirement" is not about numerical accuracy but about preserving the bijection on realized support.

10 Relation to Classical Theory

The framework connects to classical results:

- **Rate-distortion theory** [9, 8]: The Shannon lower bound gives the minimum bits to achieve distortion D. Our $b^*_\tau(X)$ is the finite-sample, task-specific analogue.

- **Entropy-constrained quantization**: Lloyd-Max and entropy-constrained VQ minimize distortion at fixed rate. Our claim is that successful neural quantization implicitly solves this for task-relevant distortion.
- **Companding** [13]: Non-uniform quantization (μ-law, A-law) allocates precision to high-density regions. SmoothQuant/AWQ are learned companders for neural activations.
- **Information bottleneck**: Tishby's framework compresses representations while preserving task-relevant information. Epilogues are the "free" compressions that lose no task-relevant bits.

Our contribution is the unifying lens: **successful quantization methods are support-shaping transformations that achieve (relaxed) injectivity on the task-relevant quotient space**.

11 Limitations

- **Distribution dependence:** $\text{supp}(X)$ and task-equivalence depend on the deployment distribution $\mathcal{D}$. Calibration sets may not capture rare events or distribution shift. Safety margins (e.g., Lipschitz bounds, widened codebooks) are needed for robustness.
- **Entropy estimation:** Noisy for high-dimensional tensors; recommend per-channel estimation with bootstrap confidence intervals.
- **Task-relevance approximation:** The Jacobian proxy J_G is local; global task-relevance may differ. The equivalence $\sim_\tau$ depends on tolerance choice.
- **Practical vs fundamental costs:** We distinguish the Landauer lower bound (information-theoretic) from practical energy (memory, switching). The former motivates the framework; the latter dominates real systems.

Precision is not a computational parameter.
It is a physical quantity determined by information content.
Additional bits have no Landauer lower bound but are computationally wasteful.

References

[1] Landauer, R. (1961). Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5(3), 183–191.

[2] Bennett, C. H. (1973). Logical reversibility of computation. *IBM Journal of Research and Development*, 17(6), 525–532.

[3] Xiao, G., Lin, J., Seznec, M., Wu, H., Demouth, J., & Han, S. (2023). SmoothQuant: Accurate and efficient post-training quantization for large language models. *ICML*.

[4] Lin, J., Tang, J., Tang, H., Yang, S., Dang, X., & Han, S. (2024). AWQ: Activation-aware weight quantization for LLM compression and acceleration. *MLSys*.

[5] Frantar, E., Ashkboos, S., Hoefler, T., & Alistarh, D. (2023). GPTQ: Accurate post-training quantization for generative pre-trained transformers. *ICLR*.

[6] Li, M., et al. (2024). SVDQuant: Absorbing outliers by low-rank components for 4-bit diffusion models. *arXiv:2411.05007*.

[7] Proakis, J. G., & Salehi, M. (2007). *Digital Communications*. McGraw-Hill.

[8] Cover, T. M., & Thomas, J. A. (2006). *Elements of Information Theory*. Wiley.

[9] Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3), 379–423.

[10] Dettmers, T., Lewis, M., Belkada, Y., & Zettlemoyer, L. (2022). GPT3.int8(): 8-bit matrix multiplication for transformers at scale. *NeurIPS*.

[11] Nagel, M., Fournarakis, M., Amjad, R. A., Bondarenko, Y., van Baalen, M., & Blankevoort, T. (2021). A white paper on neural network quantization. *arXiv:2106.08295*.

[12] Jacob, B., et al. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. *CVPR*.

[13] Gholami, A., Kim, S., Dong, Z., Yao, Z., Mahoney, M. W., & Keutzer, K. (2022). A survey of quantization methods for efficient neural network inference. *Low-Power Computer Vision*, 291–326.

[14] Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., & Bengio, Y. (2018). Quantized neural networks: Training neural networks with low precision weights and activations. *JMLR*, 18(187), 1–30.

[15] Wang, K., Liu, Z., Lin, Y., Lin, J., & Han, S. (2019). HAQ: Hardware-aware automated quantization with mixed precision. *CVPR*.

[16] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *CVPR*.

[17] Vaswani, A., et al. (2017). Attention is all you need. *NeurIPS*.