GIFT: Geometry-Informed Low-Precision Gradient Communication for LLM Pretraining

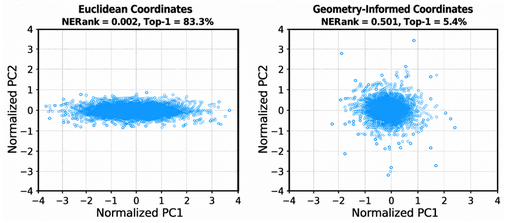
RUTGERS–NEW BRUNSWICK School of Engineering

Jieying Wang, Shuyuan Fan, Mingkai Zheng, Fangru Linghu, Zhao Zhang

MOTIVATION

Communication is a primary scaling bottleneck in distributed LLM pretraining.

- Low-precision gradient communication using FP8 can reduce the communicated payload by 75% relative to FP32 **[1]**.
- However, direct Euclidean quantization ignores the highly anisotropic geometry of gradients, causing direction-dependent distortion and reduced update fidelity.
- GIFT transforms gradients into near-isotropic, geometry-aware coordinates before quantization, improving low-precision fidelity without changing the optimizer, training recipe, communication collective, or FP8 format.



How to Read the Metrics

NERank ↑
Normalized effective rank
Higher = more isotropic

Top-1 ↓
Energy in the leading direction
Lower = more isotropic

DERIVATION

Gradient perturbations are not equally important in all directions. We derive a local geometry from KL divergence and use a Kronecker-factored approximation to expose this anisotropic geometry at the layer level **[2]**.

Cross-entropy is equivalent to minimizing KL divergence.

$$\mathscr{L}(\theta) = \mathbb{E}_{c,x\sim p^*}\left[-\log p_\theta(x\,|\,c)\right] = \text{const} + \mathbb{E}_c D_{\mathrm{KL}}(p^*\|p_\theta)$$

The entropy of the target distribution is independent of θ. Therefore, minimizing the negative log-likelihood is equivalent to minimizing $D_{KL}(p^*||p_\theta)$. This suggests that nearby models should be compared by changes in their predictive distributions, rather than by Euclidean parameter distance alone.

Locally, KL divergence induces the Fisher metric.

$$D_{\mathrm{KL}}(p_{\theta_0}\|p_{\theta_0+d\theta}) \approx \frac{1}{2} d\theta^T F(\theta_0) d\theta$$

A second-order expansion around θ_0 shows that the local change in the predictive distribution is governed by the Fisher matrix $F(\theta_0)$. Directions with larger Fisher curvature are more sensitive: the same-sized parameter displacement produces a larger functional change in the model.

Equal local KL divergence forms an ellipsoid.

$$d\theta^T F(\theta_0) d\theta = \epsilon^2$$

The eigenvectors of $F(\theta_0)$ determine the principal directions of the ellipsoid, while its eigenvalues measure sensitivity along those directions. Consequently, equal Euclidean perturbations can have very different effects on the model's predictive distribution.

Euclidean quantization allocates equal resolution to directions with unequal functional importance. This produces direction-dependent distortion for highly anisotropic gradients.

K-FAC provides a tractable layer-wise approximation.

$$F_W \approx A \otimes G, \qquad A = \mathbb{E}[aa^T], \qquad G = \mathbb{E}[\delta\delta^T]$$

$$\|\Delta W\|^2_{F(W)} = \mathrm{vec}(\Delta W)^T (A \otimes G)\mathrm{vec}(\Delta W) = \mathrm{tr}(A\Delta W^T G \Delta W)$$

Constructing the full Fisher matrix is prohibitively expensive for modern neural networks. For a weight matrix W, K-FAC approximates its Fisher block as $F_W \approx A \otimes G$, separating the geometry into two much smaller factors **[3]**. The activation factor A captures correlations among layer inputs, while the output-gradient factor G captures correlations among output-gradient components. Together, they define the local geometry of perturbations to W.

Quantize in geometry-aware coordinates, then map back before optimization.

GIFT uses the K-FAC factors to transform gradients into a near-isotropic coordinate system, where low-precision quantization is more faithful. The quantized gradients are communicated and mapped back before the optimizer update.

WHITENING

K-FAC represents each layer's anisotropic Fisher geometry using two compact Kronecker factors. Whitening transforms this geometry into approximately isotropic coordinates, reducing direction-dependent distortion during low-precision communication.

Factor the layer-wise K-FAC metric

$$A = L_A L_A^T, \qquad G = L_G L_G^T$$

For each weight matrix W, K-FAC approximates its Fisher block as $F_W \approx A \otimes G$. Cholesky factorization provides efficient layer-wise transforms without constructing or storing the full Fisher matrix.

Whiten parameter displacements into Fisher coordinates

$$U = L_G^{\mathsf{T}} \Delta W L_A, \qquad \mathrm{tr}\left(A\Delta W^{\mathsf{T}} G \Delta W\right) = \|U\|_F^2$$

This transformation converts the anisotropic Fisher metric into the ordinary Frobenius norm. Equal-KL ellipsoids therefore become approximately spherical, equalizing local sensitivity across directions and reducing direction-dependent quantization error.



Transform communicated gradients with the corresponding inverse factors

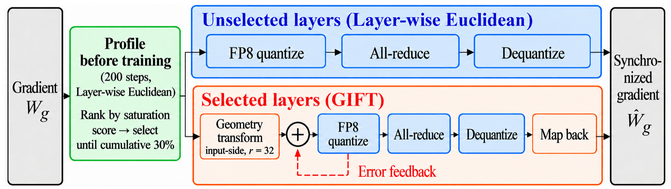
$$\widetilde{W}_g = L_G^{-1} W_g L_A^{-\mathsf{T}}, \qquad \widehat{W}_g = L_G\, Q(\widetilde{W}_g)\, L_A^{\mathsf{T}}$$

Gradients use the inverse factors because they are dual to parameter displacements. GIFT quantizes and communicates $\widetilde{W}_g$ in the whitened coordinates, then maps the result back before the optimizer update. The optimizer and training recipe remain unchanged.

COMMUNICATION

GIFT inserts a geometry-aware coordinate transformation around the existing FP8 all-reduce. Selected layers communicate gradients in near-isotropic coordinates, while the remaining layers use standard Euclidean FP8 communication.

Route each layer through the appropriate communication path



Unselected layers follow standard Layer-wise Euclidean FP8 communication. Selected layers are routed through the GIFT path, which applies a rank-32 input-side geometry transform, performs FP8 quantization and all-reduce in the transformed coordinates with error feedback, and maps the synchronized gradient back before the optimizer update.

Geometry Maintenance

For selected layers, GIFT refreshes the rank-32 input-side geometry every $K = 50$ training steps. Input covariance statistics are synchronized across the data-parallel group, and each rank locally constructs the same low-rank-plus-diagonal factor $L_{A,r}$.

Error Feedback & Scale Synchronization

Error feedback is maintained in the same transformed coordinates used for FP8 quantization and is reset whenever the geometry is refreshed. GIFT uses E5M2 FP8 with a MAX-synchronized scale across data-parallel ranks.

Same FP8 format · same all-reduce collective · same model · same optimizer and training recipe

IMPLEMENTATION

1 Input-side geometry

Among the one-sided variants, the input-side factor provides the stronger approximation. It reduces RelL2 from 5.272×10^{-2} under Euclidean FP8 to 2.223×10^{-2}, compared with 1.545×10^{-2} for the full two-sided transform, while avoiding the output-side factor and reducing geometry overhead.

$$L_G^{-1} W_g L_A^{-T} \Longrightarrow W_g L_A^{-T}$$

2 Rank-32 input-side factor

We approximate the input-side geometry with a rank-32 low-rank-plus-diagonal factor. Rank 32 achieves a RelL2 of 2.494×10^{-2}, compared with 2.223×10^{-2} for the dense input-side geometry, retaining approximately 91% of its error-reduction benefit with substantially lower overhead.

$$\widetilde{A}_r = U_r \Lambda_r U_r^T + D_r, \qquad r = 32$$

3 Automatic layer selection

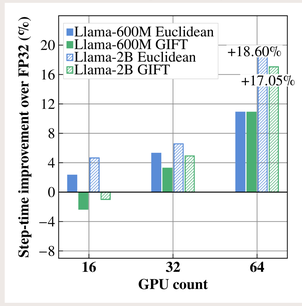
Before formal pretraining, GIFT performs a separate 200-step Layer-wise Euclidean FP8 profiling run over candidate MLP layers. Layers are ranked by cumulative FP8 boundary hits, and GIFT selects the smallest prefix whose cumulative quantization error accounts for at least 30% of the total. The selected set is then fixed for a fresh pretraining run, while the remaining layers continue using standard Layer-wise Euclidean FP8 communication.

200-step profiling → boundary-hit ranking → select 30% error prefix → restart from a fresh model

RESULTS

We evaluate GIFT on Llama-600M and Llama-2B for end-to-end efficiency and downstream-task preservation.

Distributed Training Efficiency at Scale



Downstream Performance

Method	Avg.	≥ Muon FP32
Llama-600M		
Muon FP32	0.5060	–
AdamW FP32	0.5094	6/14
Muon BF16	0.5017	7/14
Layer-wise Euclidean FP8	0.5186	5/14
Block-wise Euclidean FP8	0.5023	4/14
Layer-wise Euclidean FP8 + EF	0.5050	3/14
Block-wise Euclidean FP8 + EF	0.5111	4/14
SQ-FP8	0.5086	6/14
HS-FP8	0.5043	8/14
GIFT	**0.5213**	**8/14**
Llama-2B		
Muon FP32	0.5102	–
Layer-wise Euclidean FP8	0.5099	4/14
GIFT	**0.5231**	**7/14**

Across both model scales, GIFT consistently delivers a better accuracy–efficiency trade-off.

REFERENCES

[1] P. Micikevicius et al., "FP8 Formats for Deep Learning," *arXiv preprint arXiv:2209.05433*, 2022.

[2] S.-I. Amari, *Information Geometry and Its Applications*, Applied Mathematical Sciences, vol. 194. Tokyo, Japan: Springer, 2016.

[3] J. Martens and R. Grosse, "Optimizing Neural Networks with Kronecker-Factored Approximate Curvature," in *Proc. 32nd Int. Conf. Machine Learning (ICML)*, 2015, pp. 2408–2417.