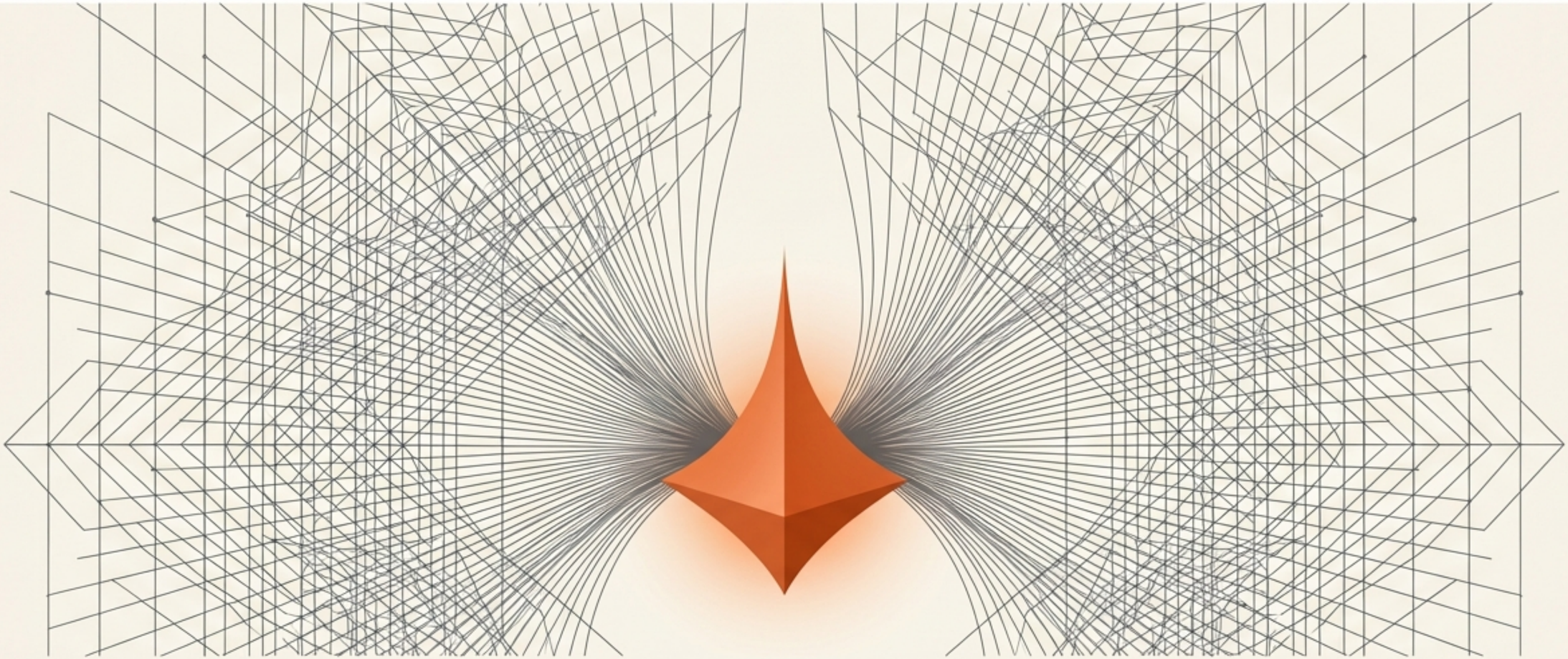
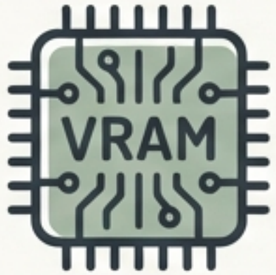


Overcoming the Size Barrier: The Unified Landscape of LLM Efficiency

Democratizing Large Language Models through Efficient Fine-Tuning and Inference.

The Trilemma of LLM Scaling



Memory (GPU VRAM)

Scales at O(d) for weights, optimizer states, and gradients.

Compute (Throughput)

Scales at O(6 * d * D) FLOPs for D tokens during training.

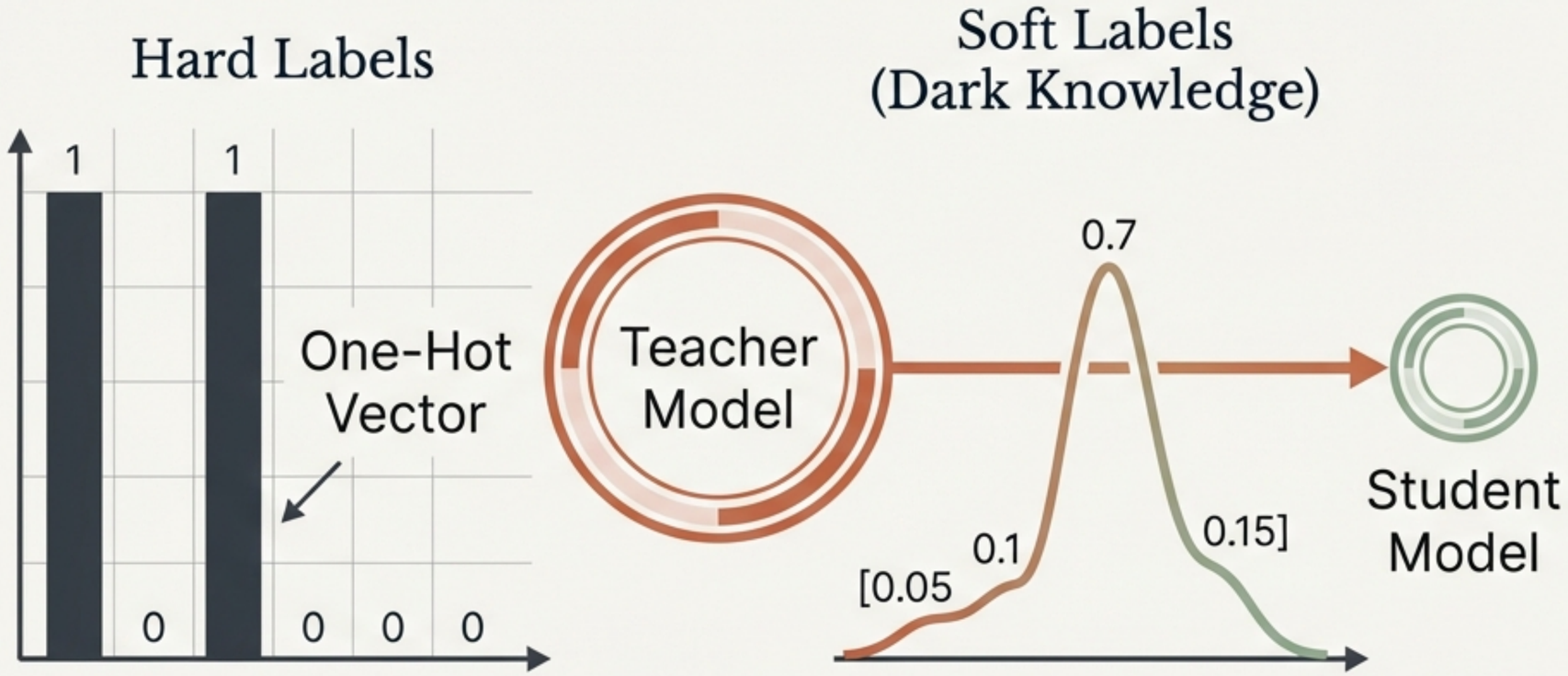
Latency (Real-time serving)

Memory-bound O(d / bandwidth) per token during inference.

$$\mathcal{L}(\hat{M}) \approx \mathcal{L}(M) \quad \text{subject to} \quad C(\hat{M}) \ll C(M)$$

← Match the evaluation loss of the large model while fundamentally shrinking the composite cost function.

Knowledge Distillation: Transferring Dark Knowledge



Distillation Loss

Balances Hard Label Loss (Cross-Entropy) with Soft Label Loss (KL Divergence with temperature $tau > 1$ to smooth probabilities).

SeqKD & Token-KD

Sequence-level generation mimicking vs. autoregressive token-by-token matching.

Speculative Decoding

An implicit distillation target. A small draft model proposes K tokens; a massive verifier model accepts/rejects based on probability ratios.

Model Pruning: Trimming the Fat

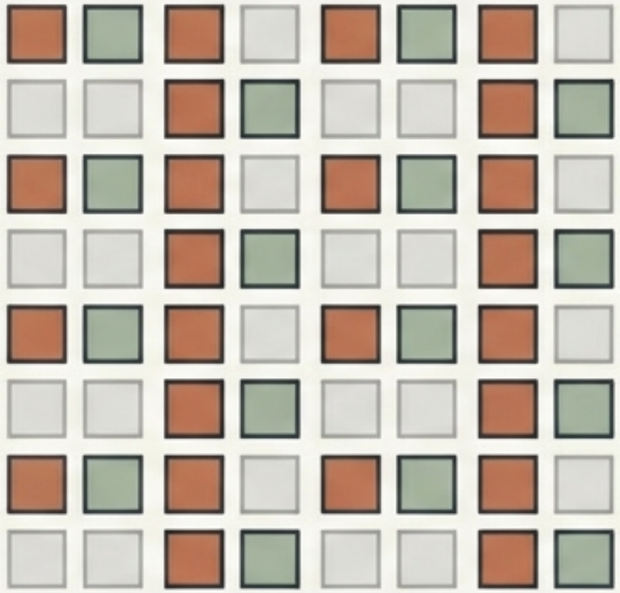
Exploiting over-parameterization to reduce network density.

Unstructured Pruning

Wanda: Prunes based on weight magnitude multiplied by input activation norm.

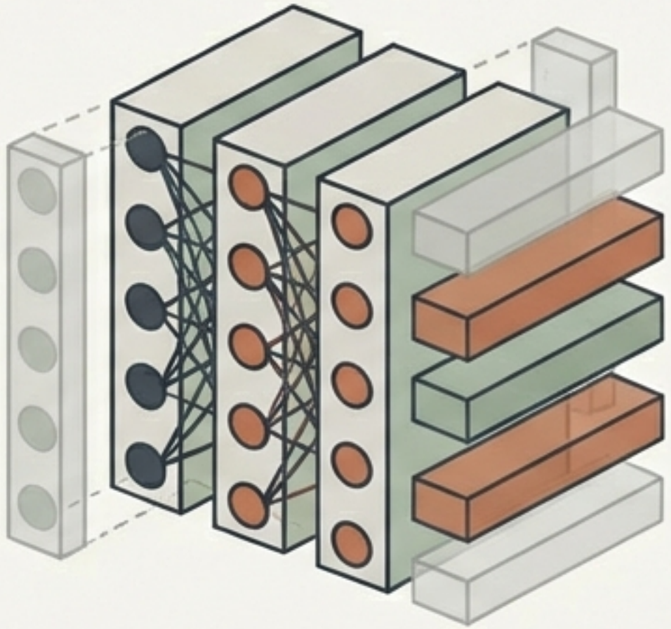
SparseGPT: Layer-wise sparse regression via Optimal Brain Surgeon without retraining.

Semi-Structured Pruning



NVIDIA 2:4 Sparsity: Exactly 2 non-zeros per 4 weights. Guarantees 2x hardware throughput on Ampere hardware.

Structured Pruning

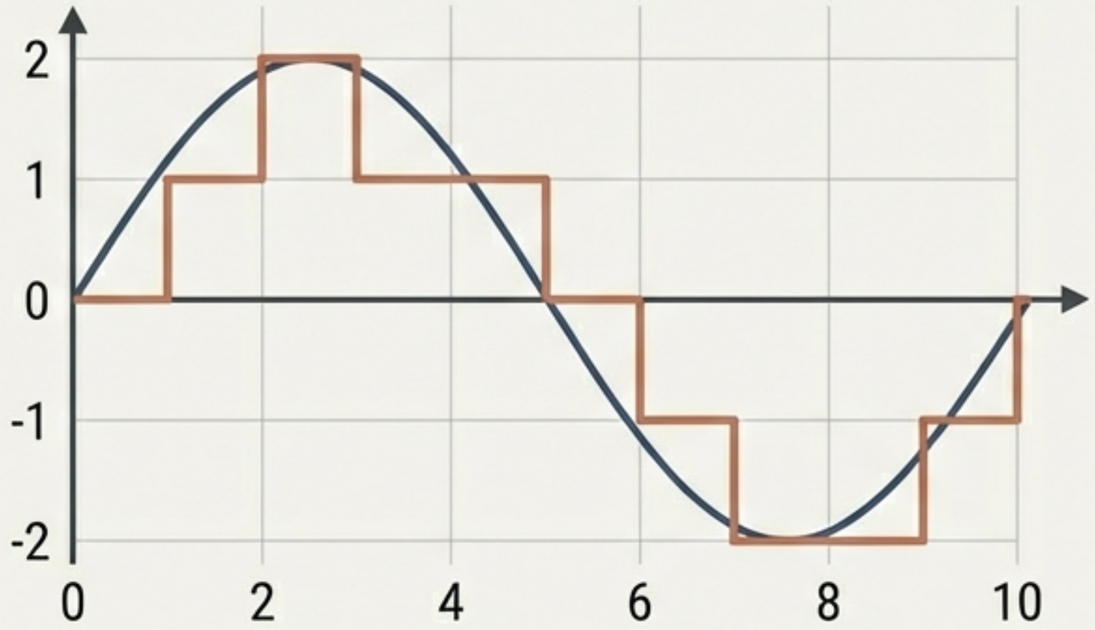


SliceGPT: PCA-based width reduction.

Layer Pruning: Removing layers via angular distance identity checks.

Quantization Fundamentals: Precision vs. Memory

Visual Math



$$q = \text{clamp}\left(\text{round}\left(\frac{x}{s}\right) + z \right)$$

Scale Factor

Zero Point

Granularity Matrix

The Granularity Trade-off

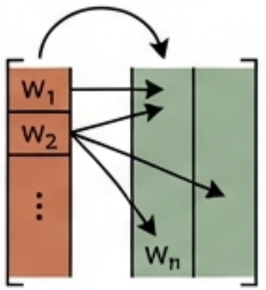
Method	Scaling Granularity	Accuracy Impact
Per-Tensor	1 scale per matrix	Lowest Accuracy
Per-Channel	1 scale per output row	Medium Accuracy
Per-Group	1 scale per g elements (32, 64, 128)	Highest Practical Accuracy (Industry Standard)

Advanced Quantization: Post-Training to QAT

GPTQ

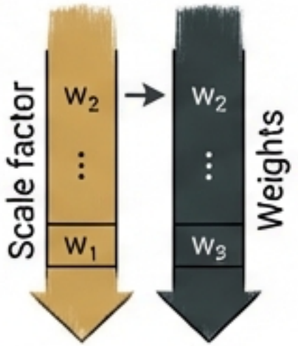
Uses **Hessian**-based weight **compensation**. Error redistributes to unquantized columns.

Strong at 4-bit.



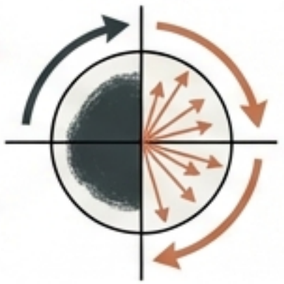
AWQ

Scales salient (highly active) channels prior to quantization to protect critical weights.



QuIP#

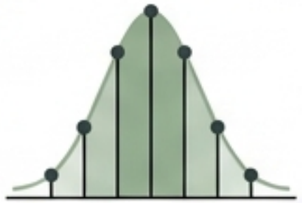
Uses random **orthogonal rotations** to make weight distributions **incoherent**, enabling enabling SOTA 2-bit quantization.



NF4 (NormalFloat)

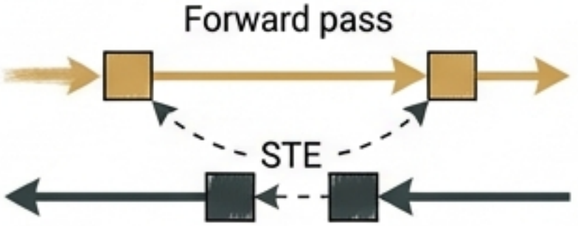
Places quantization grid points at **equal-probability intervals** for normally distributed weights.

Crucial for QLoRA.



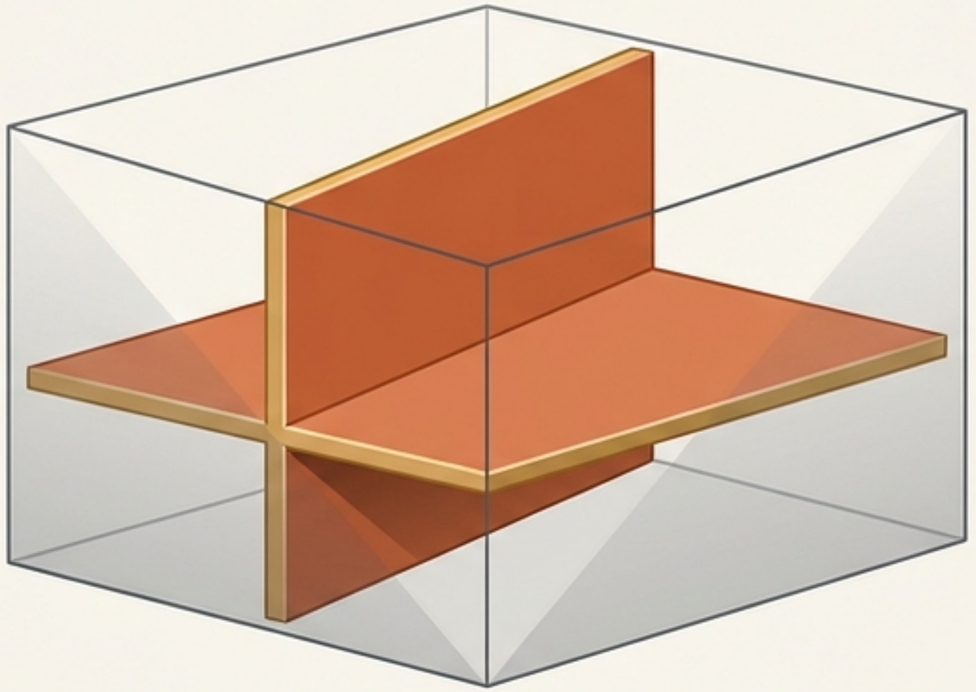
QAT (Quantization-Aware Training)

Uses the **Straight-Through Estimator** (STE) to pass gradients through discrete quantized steps during the actual training loop.



The Intrinsic Dimensionality Hypothesis

High-Dimensional Space
($d \sim 10^9$ parameters)



Low-Dimensional Intrinsic Subspace
($d_{\text{int}} \sim 10^3$)

$$\theta_{\text{adapted}} = \theta^* \oplus \varphi$$

The Core Hypothesis

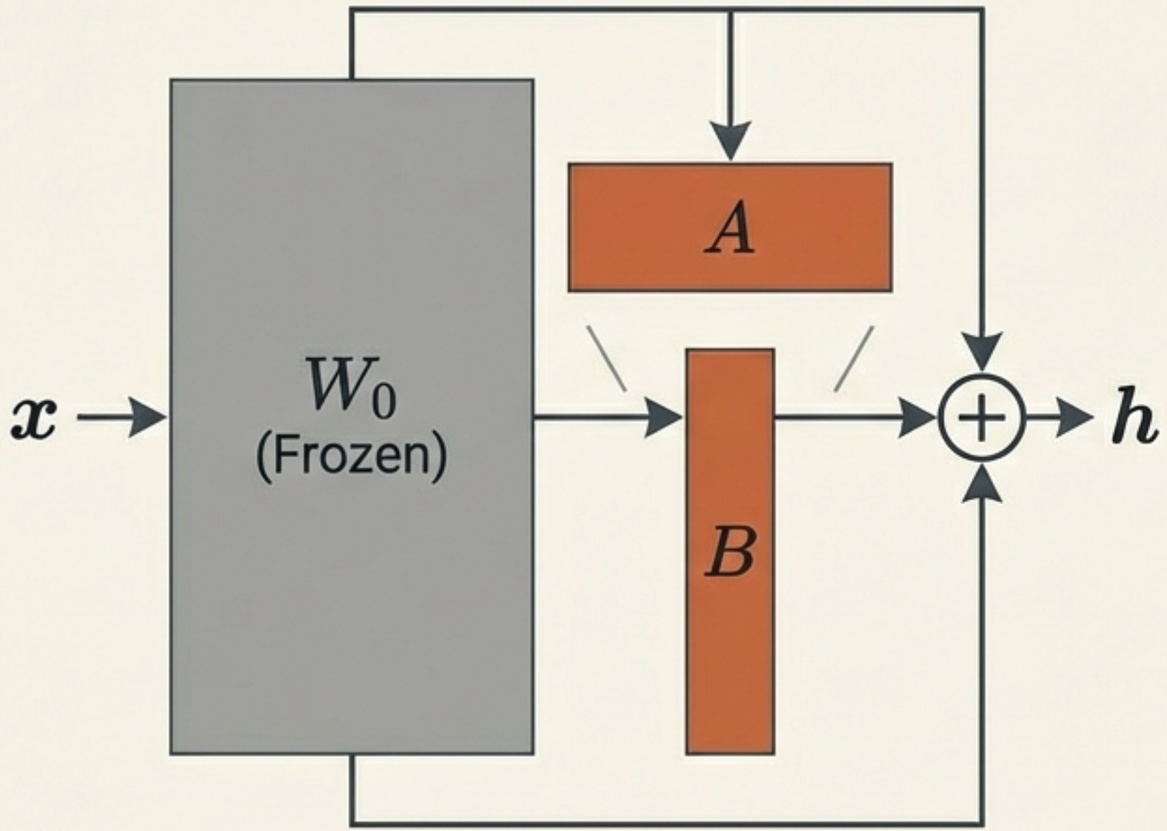
Fine-tuning does not require updating all parameters. The adaptation process naturally operates in a low-dimensional subspace.

The PEFT Pivot

We freeze the massive pretrained base model (θ^*) and only train a tiny, structured set of adaptation parameters (φ).

The LoRA Revolution

Low-Rank Adaptation: The Industry Standard for PEFT



$$h = W_0 x + \frac{\alpha}{r} BAx$$

Stability Scaler: Decouples learning rate sensitivity from rank r.

0.78% Parameter Footprint

Trainable parameters plummet while retaining full expressive power (for r=16, d=4096).

Zero-Latency Inference

Adapters merge mathematically into the base weights prior to serving: $W_{\text{merged}} = W_0 + \frac{\alpha}{r} BA$

Expanding the LoRA Family

QLoRA

Inter

4-bit NF4 base + double quantization + paged optimizers. Reduces **65B** training memory from **520GB** to **34GB**.

DoRA

Inter

Decomposes weight updates into distinct magnitude and direction components.

LoRA+

Inter

Applies distinct, optimized learning rates for A and B matrices.

rsLoRA

Inter

Rank-stabilized scaling using $1/\sqrt{r}$ to maintain gradient magnitude at high ranks.

AdaLoRA

Inter

Dynamically allocates rank across different layers via importance-aware SVD.

VeRA

Inter

Extreme efficiency using frozen, shared random matrices across layers, learning only diagonal scaling vectors.

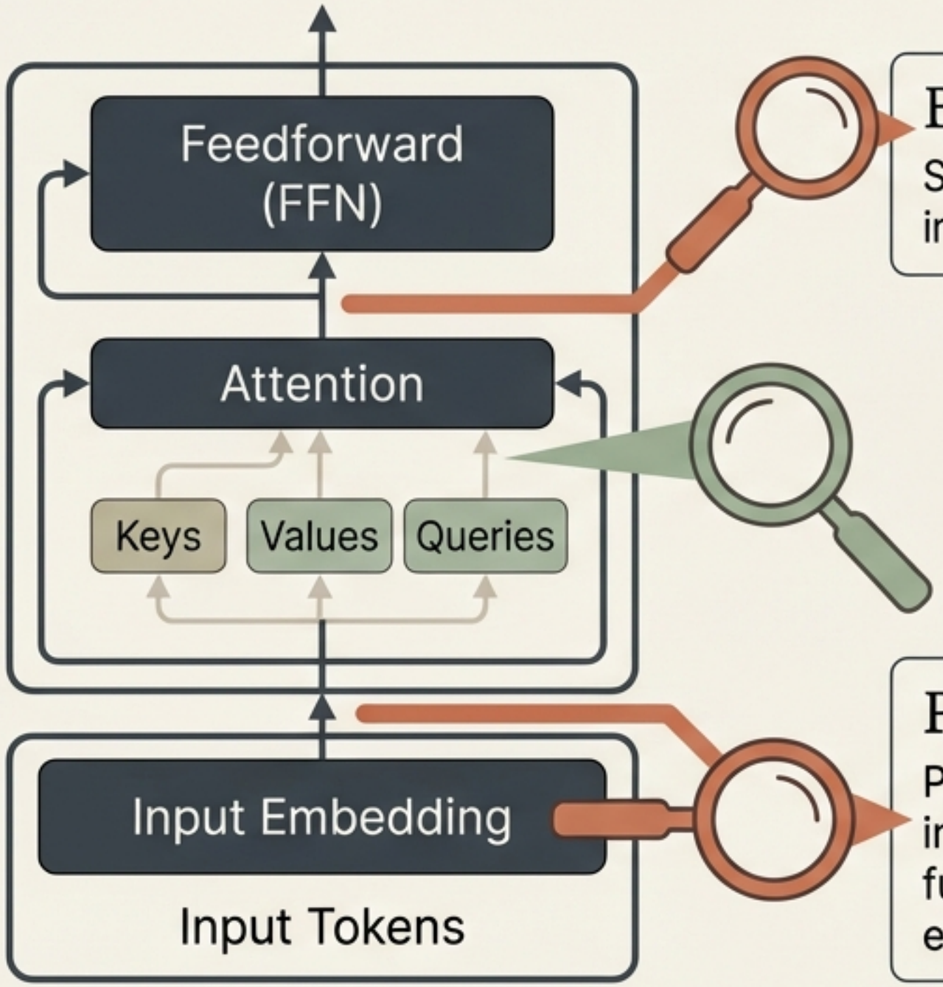
GaLore

Inter

Projects gradients (not weights) into a low-rank subspace, saving up to **8x** optimizer memory.

Beyond LoRA: Adapters, Prefix, & Prompt Tuning

Parameter-Efficient Fine-Tuning (PEFT) Methods Beyond Low-Rank Adaptation



Bottleneck & Parallel Adapters

Small feedforward modules inserted serially in parallel to the frozen sub-layers.

Prefix Tuning

Prepending L_p trainable vectors to Keys and Values at every layer, acting as a task-specific virtual context.

Prompt Tuning

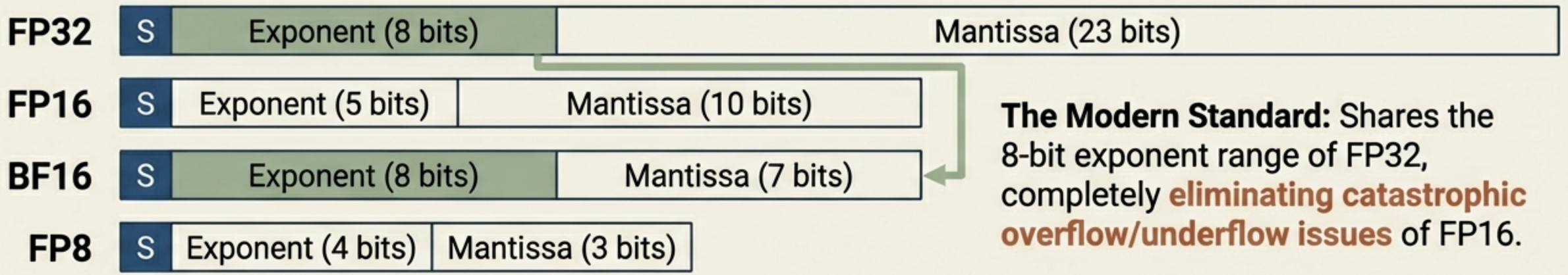
Prepending soft prompts only at the input embedding layer. Gap with full fine-tuning shrinks as models exceed 11B parameters.

IA^3

Infusing inner activations by learning extremely lightweight, per-element rescaling vectors for keys, values, and FFN intermediates.

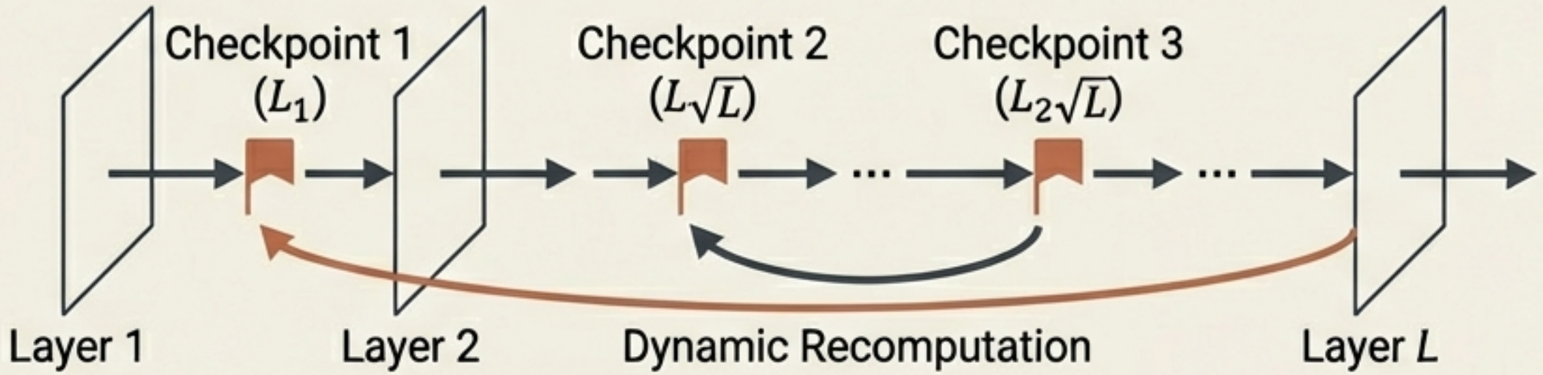
Efficient Training: Math & Memory

Floating-Point Formats



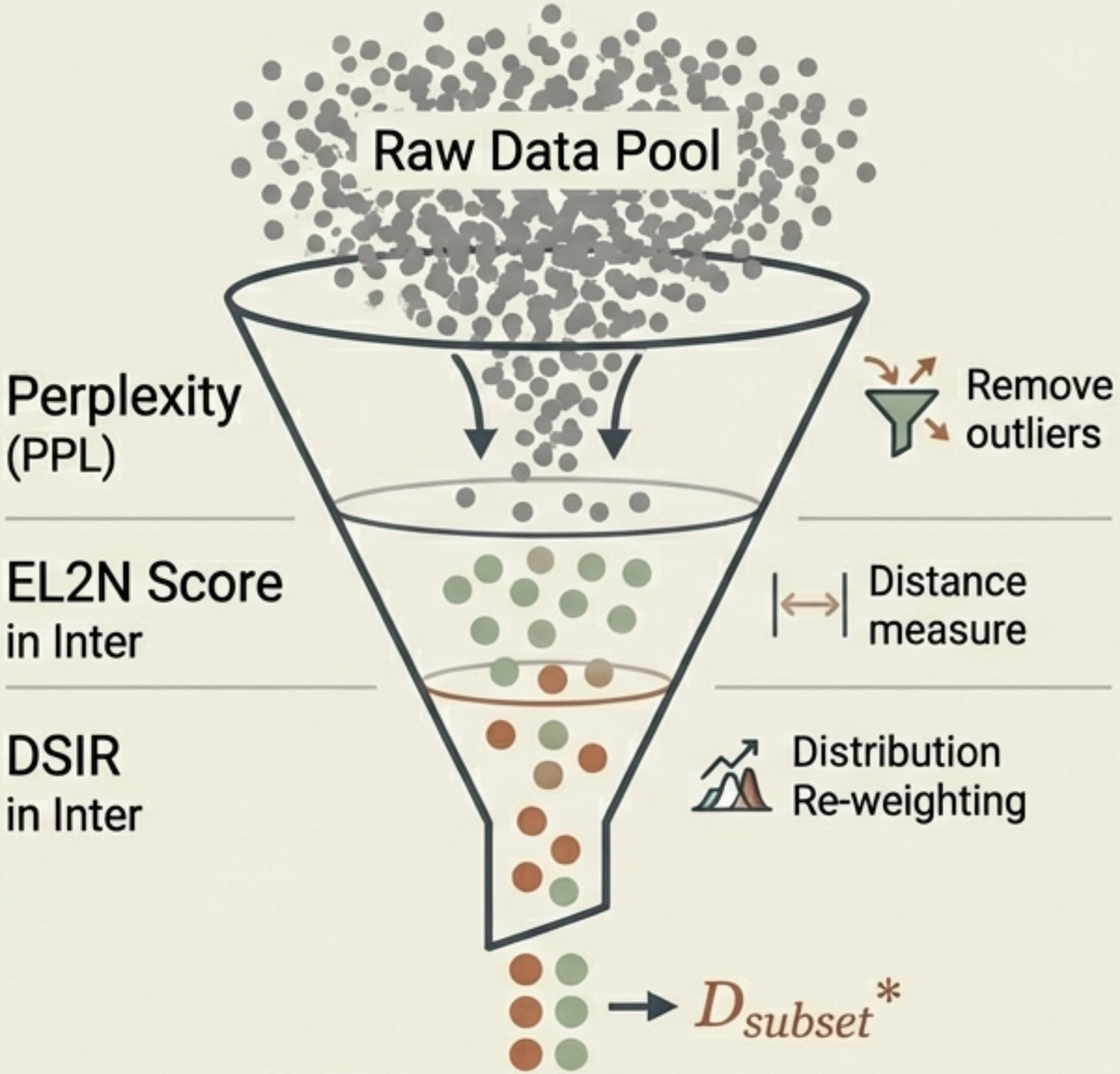
The Modern Standard: Shares the 8-bit exponent range of FP32, completely **eliminating catastrophic overflow/underflow issues** of FP16.

Gradient Checkpointing



Activation Recomputation: Trades a **33% compute overhead** for an $O(\sqrt{L})$ **memory footprint reduction**.

Data Selection & Curriculum



Perplexity (PPL)

Rejects extremes. Too low = memorized/repetitive. Too high = garbage/noise.

EL2N Score

Identifies ‘hard examples’ by measuring the L2 distance between model probabilities and the target.

DSIR

Importance resampling mapping the general data pool distribution to a high-quality target domain.

Curriculum Learning (Skill-It)

Models skills as a Directed Acyclic Graph (DAG), ensuring foundational skills are mastered before composite ones.

Prompt Compression & Context Optimization

Solving the $O(T^{\wedge 2} * d)$ quadratic attention cost during inference.

LLMLingua

Evicts tokens based on conditional perplexity. **High PPL** (informative) are kept; **Low PPL** (redundant) are evicted.

Gist Tokens

Trains the model to compress arbitrary context into a fixed number (k) of **condensed "gist" tokens**.

H_2O (Heavy-Hitters)

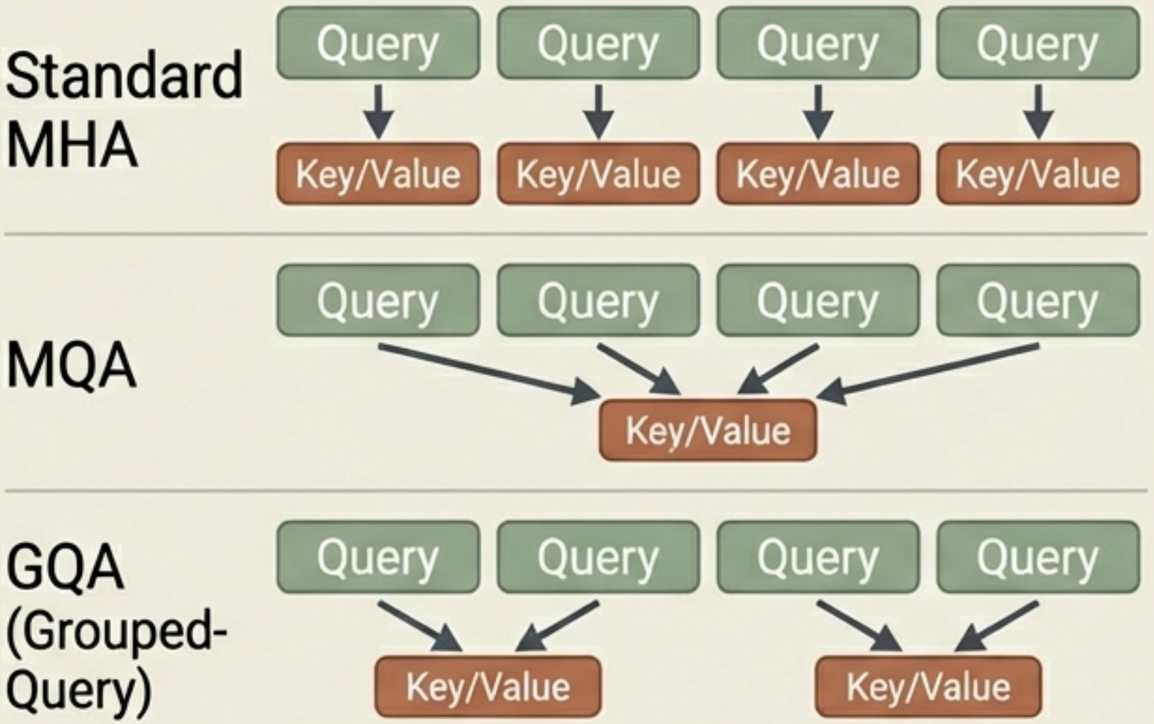
Dynamically evicts tokens from the KV-cache, retaining only the **~5-10%** of tokens with the highest cumulative attention scores.

StreamingLLM

Preserves the first 4 **"attention sink" tokens** (which anchor normalizations) plus a sliding window of recent tokens.

System-Level Inference & Distributed Training

KV-Cache Optimization



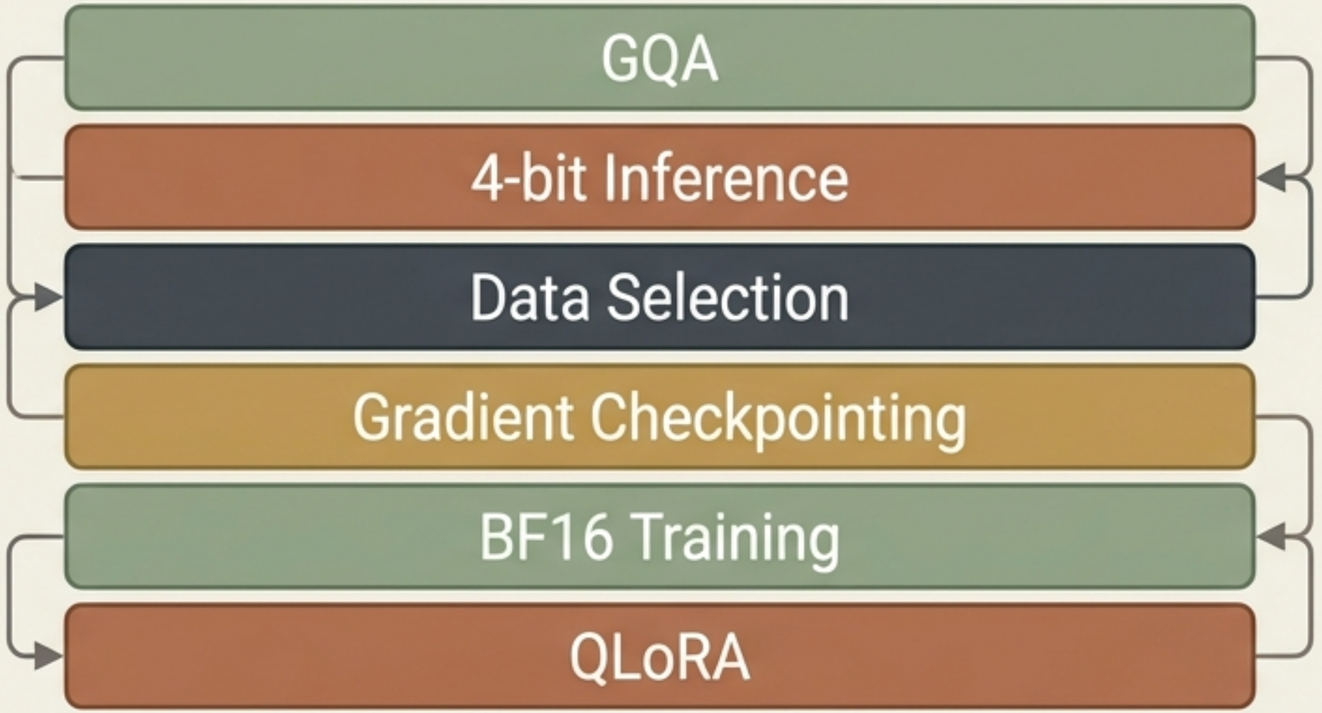
LLaMA 2 70B uses G=8 with H=64, yielding an 8x reduction in KV-cache size.

ZeRO Distributed Training

ZeRO Stages	
Stage 1	**Optimizer** states sharded.
Stage 2	**Optimizer** + **Gradients** sharded.
Stage 3 (FSDP)	**Optimizer** + **Gradients** + **Parameters** sharded.

The Unified Efficiency Stack

$$\text{Efficiency} = f(\text{Compression}, \text{PEFT}, \text{Training}, \text{Inference})$$



The Takeaway: This unified stack enables the fine-tuning of 70B-parameter models on dual 24GB consumer GPUs. Frontier AI is officially democratized.