

KV Cache Quantization (KIVI & FP4) Compression Guide

Directional Asymmetry in Key Channels vs. Value Tokens, Memory Footprint Scaling, and 1M-Token LLM Serving

EXECUTIVE ARCHITECTURAL SUMMARY: While weight quantization slashes model storage footprints, ultra-long context serving (128k to 1M tokens) shifts the GPU memory bottleneck entirely to the Key-Value (KV) cache. KIVI and FP4 micro-scaling exploit the directional distribution asymmetry between Key tensors (channel-wise outliers) and Value tensors (token-wise variance), compressing KV cache footprint by 75% to 80% to enable 1-million-token context serving on a single 80GB GPU.

KV Cache Precision	VRAM per Token (70B GQA)	Max Context on Single 80GB H100	Throughput at 128k Context	Perplexity Degradation (WikiText-2)
Standard FP16 (Baseline)	160 bytes / token	32,768 tokens (OOM threshold)	8.2 tok / sec	0.00 (Reference Baseline)
FP8 (E4M3 / E5M2)	80 bytes / token	65,536 tokens	15.4 tok / sec	+ 0.02 (Negligible)
INT4 Per-Tensor Uniform	40 bytes / token	131,072 tokens	22.8 tok / sec	+ 0.45 (Visible degradation)
FP4 Micro-Scaling (E2M1)	25 bytes / token	262,144 tokens	28.5 tok / sec	+ 0.08 (Highly robust)
KIVI 2-Bit Asymmetric	22 bytes / token	1,048,576 tokens (1M Context)	34.1 tok / sec	+ 0.09 (Near-lossless)

1. The Million-Token VRAM Bottleneck: Key-Value Cache Explosion

In transformer autoregressive decoding, past Key and Value attention tensors must be preserved in VRAM to prevent recomputing attention history. For a 70-billion-parameter model with Grouped Query Attention (GQA, e.g., 8 KV heads, hidden dimension 8192, 80 layers):

- Memory Formula: $\text{Memory} = 2 \cdot n_{\text{layers}} \cdot n_{\text{heads_kv}} \cdot d_{\text{head}} \cdot \text{seq_len} \cdot \text{bytes_per_elem}$. At FP16 (2 bytes), each token consumes 160 kilobytes of memory.
- The Context Wall: A 128k context consumes 20.48 GB of VRAM. A 1-million-token context requires an astonishing 160 GB of VRAM—demanding two full 80GB H100 GPUs solely to store the KV cache of a single user session.

2. The Directional Asymmetry Discovery: Key Outliers vs. Value Tokens

Naive uniform INT4/INT2 quantization collapses LLM reasoning because Key and Value matrices exhibit fundamentally different mathematical geometries:

- Key Tensor Channel Outliers: Specific channels in Key vectors exhibit enormous activation magnitudes across all sequence steps. Quantizing per-token squashes these outlier channels, destroying attention softmax sharpness.
- Value Tensor Token Variance: Value matrices show no persistent channel outliers; their variance is concentrated along individual tokens across sequence time.

3. KIVI 2-Bit Architecture & FP4 Micro-Scaling Kernels

The KIVI framework solves this dilemma through asymmetrical dimensional quantization:

- Per-Channel Key Quantization: Key vectors are quantized along the hidden feature dimension (column-wise), grouping outlier channels together into isolated scaling bins.
- Per-Token Value Quantization: Value vectors are quantized along the sequence time dimension (row-wise), preserving token-level dynamic range.
- Streaming Residual Buffer: The most recent 32 tokens are retained in unquantized FP16 inside a small sliding buffer, shielding near-field attention from quantization noise before packing tokens into 2-bit blocks.
- NVIDIA Blackwell FP4 Native Acceleration: Deploying microscaled FP4 (E2M1 with 16-element block scaling factors) maps directly into Blackwell Tensor Cores, executing 4-bit attention matrix math at line rate.

4. Production Serving Telemetry in vLLM & SGLang

Benchmarked across Llama-3-70B and Mistral-Large, KIVI 2-bit quantization shrinks KV cache footprint by 78%, unlocking 1-million-token context lengths on a single H100 SXM5 GPU while accelerating decoding throughput by 3.8x.

STRATEGIC ENGINEERING TAKEAWAYS & DEPLOYMENT RULES:

- KIVI addresses the KV cache memory wall by quantizing Keys per-channel and Values per-token.
- 2-bit asymmetric quantization reduces KV cache memory footprint by 78% with less than 0.1 perplexity loss.
- Enables serving 1-million-token context windows on a single 80GB GPU without distributed memory pooling.