Kitty: Accurate and Efficient 2-bit KV Cache Quantization with Dynamic Channel-wise Precision Boost

Haojun Xia, Xiaoxia Wu, Jisen Li, Tsai-chuan Wu, Junxiong Wang, Jue Wang, Chenxi Li, Aman Singhal, Alay Dilipbhai Shah, Alpay Ariyak, Donglin Zhuang, Zhongzhu Zhou, Ben Athiwaratkun, Zhen Zheng, Shuaiwen Leon Song

Ninth Annual Conference on Machine Learning and Systems (MLSys 2026)
May 20, 2026, Bellevue, WA

Background & Motivation

Memory Bottleneck in Long-Context Era

- **KV Cache**
  - Required by Attention [1] mechanism
  - Size scaling **linearly** with context length **L**

$$\text{Size} \approx N_{\text{layer}} \times B \times \left(H_{\text{kv}} \times D\right) \times L \times 2$$

- **Rapid Increasing of Context Length**
  - Prohibitive size of KV cache
  - Easily exceeding the size of model parameters for ***long-context inference***

Model	Release	Context Length
GPT-2	2019	1,024
GPT-3	2020	2,048
LLaMA1	2023	2,048
LLaMA2	2023	4,096
GPT-4 Turbo	**2023**	**128,000**
Qwen 3	**2025**	**131,072** (extended)
Claude Opus 4.6	**2026**	**1,000,000** (beta)
Gemini 3.1 Pro	**2026**	**1,000,000**

[1] *Vaswani, Ashish, et al. "Attention Is All You Need."*

Accuracy Drop for 2-bit KV Cache

- **KV Cache Quantization**
  - Using fewer bits to represent cached Key & Value
- **Significant Accuracy Drop with 2-bit KV**
  - 4-bit KV cache can maintain model accuracy well
  - Aggressive **2-bit** quantization leads to substantial accuracy degradations

*Accuracy degradation of low-bit KV cache on **Qwen3-8B.***

Model	Task	FP16	KIVI-4bit	KIVI-2bit
Qwen3-8B	GSM8K	94.79	94.41	89.13 ↓
	MATH	88.26	88.46	47.29 ↓
	GPQA	40.71	38.98	32.24 ↓
	HumanEval	84.82	84.09	76.89 ↓
	AIME24	71.67	77.67	57.00 ↓
	AIME25	66.00	65.33	52.33 ↓
LLaMA3-8B	GSM8K	76.75	76.09	63.58 ↓
	MATH	47.15	47.85	31.45 ↓
	GPQA	26.94	25.82	23.88 ↓
	HumanEval	63.96	62.07	55.30 ↓

Design Space Exploration

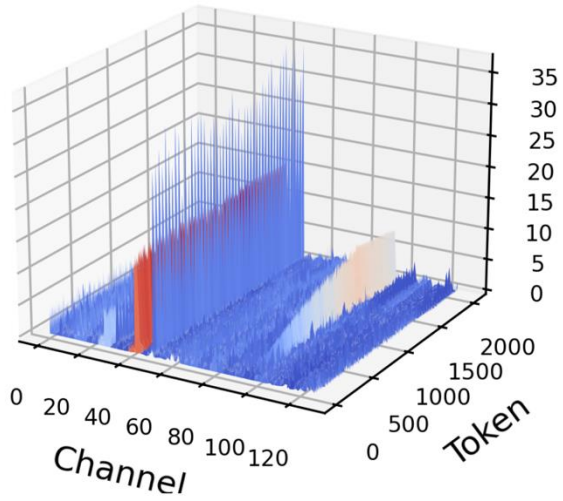
- **Preserving the Initial Tokens in Full Precision**
- **Increasing the Precision of Key Cache**
  - *How about **partially** increasing the precision of Key cache?*

Model	Task	FP16	KIVI-K2V2	KIVI-K2V2*	KIVI-K2V4*	KIVI-K4V2*
Qwen3-8B	GSM8K	94.79	89.13	89.71	90.14	**93.96**
	MATH	88.26	47.29	74.92	82.50	**87.92**
	GPQA	40.71	32.24	36.02	35.82	**40.51**
	HumanEval	84.82	76.89	78.54	81.77	**83.41**
	AIME24	71.67	57.00	67.67	71.00	**76.00**
	AIME25	66.00	52.33	57.67	**64.33**	**64.33**
LLaMA3-8B	GSM8K	76.75	63.58	71.04	72.38	**76.62**
	MATH	47.15	31.45	44.12	44.09	**48.41**
	GPQA	26.94	23.88	23.67	24.80	**25.20**
	HumanEval	63.96	55.30	56.71	59.02	**62.80**

Algorithm-System Co-Design

Channel-wise Patterns in Key Cache

- **Discrepancy of Magnitude across Channels (Key Cache)**
  - The magnitudes of different channels in Key cache vary considerably
  - Using **channel-wise quantization**
- **Quantization Error Reduction**
  - Δ_d : quantization step size (resolution)
  - R_d : max – min (channel-wise)
  - higher **magnitude** -> higher $\mathbf{R_d}$ -> higher **error**
  - Error reduction: **channel-wise precision boost**



$$\Delta P = Q\hat{K}^T - QK^T = QE^T. \quad (3)$$

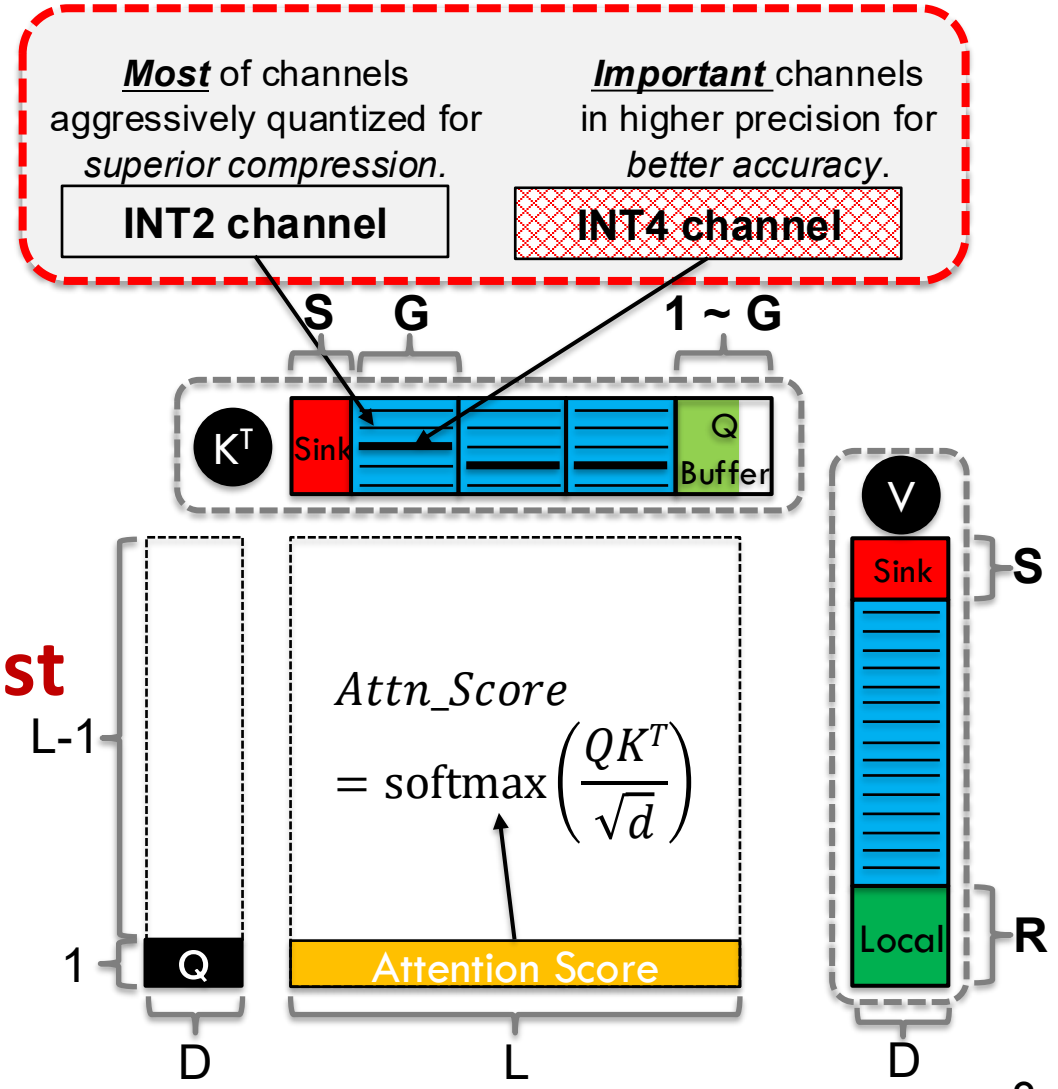
$$|\Delta P_{i,j}| = \left|\sum_{d=1}^{D} Q_{i,d}E_{j,d}\right| \leq \sum_{d=1}^{D} |Q_{i,d}| \cdot |E_{j,d}|. \quad (4)$$

$$|E_{j,d}| \leq \frac{\Delta_d}{2}, \quad \text{where} \quad \Delta_d = \frac{R_d}{2^b - 1}. \quad (5)$$

$$\frac{\Delta_{d,\text{2-bit}}}{\Delta_{d,\text{4-bit}}} = \frac{2^4 - 1}{2^2 - 1} = \frac{15}{3} = 5\times. \quad (6)$$

Overall Quantization Scheme

- **Adopt KIVI-style Scheme**
  - Per-channel Quantization for Keys
  - Per-Token Quantization for Values
  - Preserving local window in full precision
- **Preserving Attention Sink in FP16**
  - Preserving initial tokens (Sink) in full precision
- **Dynamic Channel-wise Precision Boost**
  - Critical channels preserved in **INT4** while the others quantized to **INT2**
  - **Reducing accuracy distortion** of quantization while **improving memory efficiency**

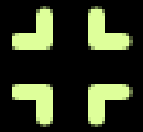


Runtime Components

- **Amortized Runtime Overhead** (Channel Ranking, Quantization)
- **Fused Dequantization & MatMuls** (Triton GPU Kernels)

Online Channel Ranking & Selection

On-the-fly ranking of channels based on sensitivity heuristics (executed every 128 decoding steps).



Online Quantization & Packing

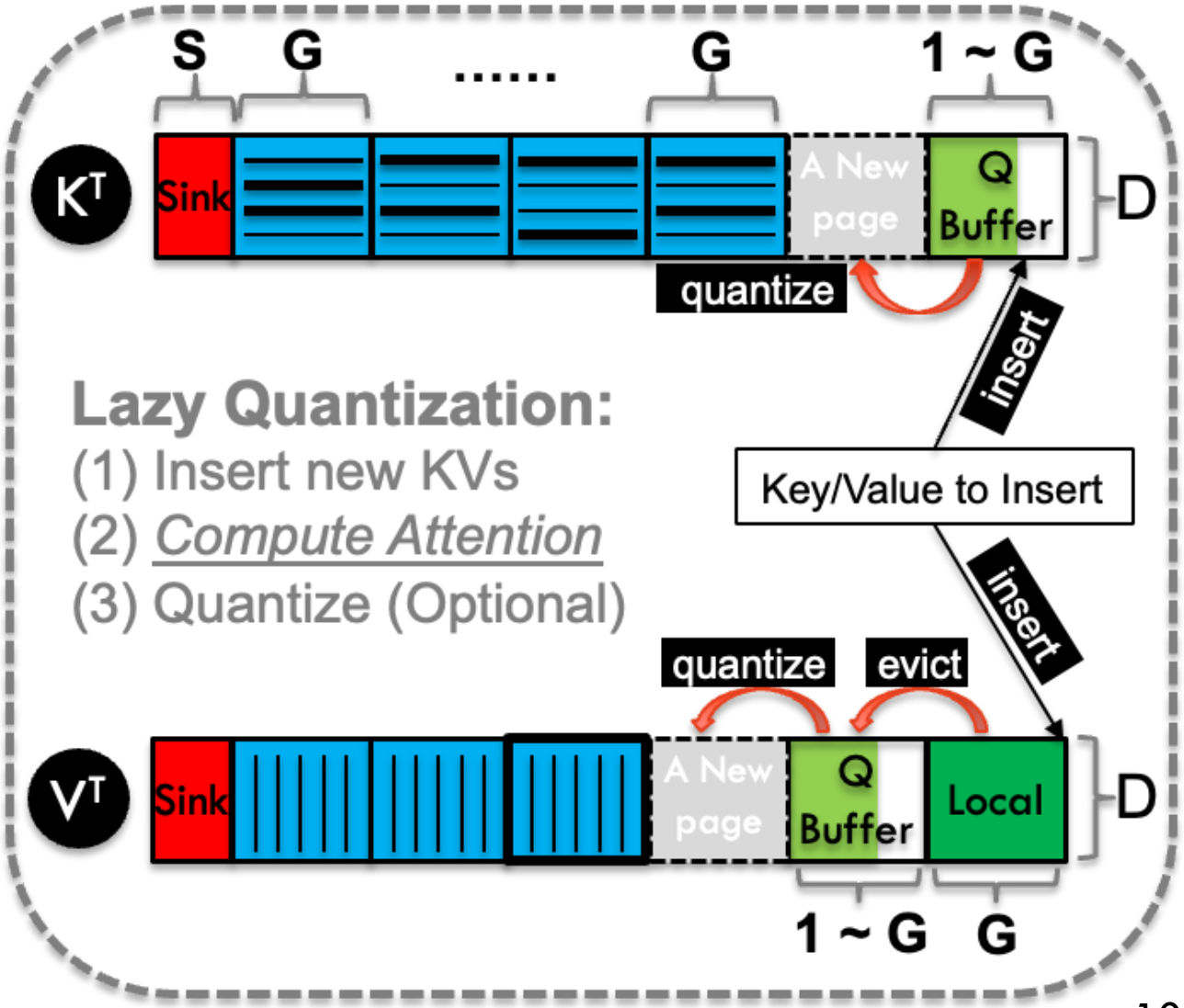
Mixed-precision channel-wise Key cache quantization (executed every 128 decoding steps).

Fused MatMul & Dequantization

Fusing the KV dequantization process into the MatMuls, e.g., QK^T, with triton.jit.

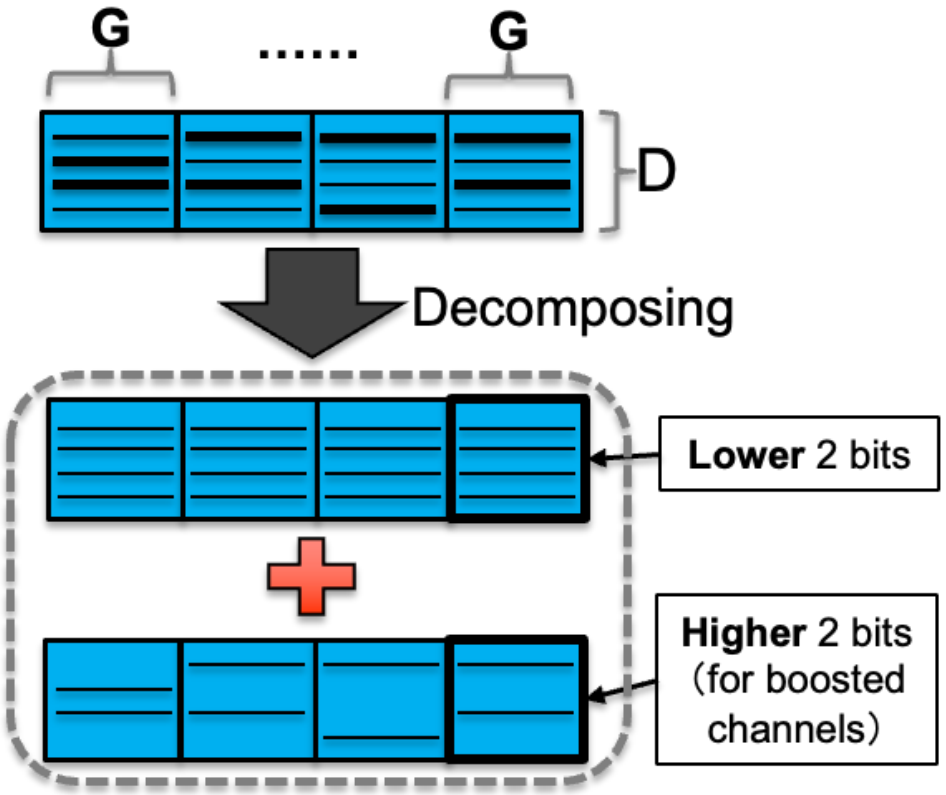
Runtime Memory Management

- **Group-wise Quantization**
  - Quantization granularity: **KV page**
    - Called once for every G steps
  - QBuffer: temporarily storing KVs before forming a complete page
- **Lazy Quantization**
  - (1) Inserting new KVs;
  - (2) Loading and dequanting KV pages & **computing attention**
  - (3) (Optional) quantizing & packing Q-Buffer into a new page



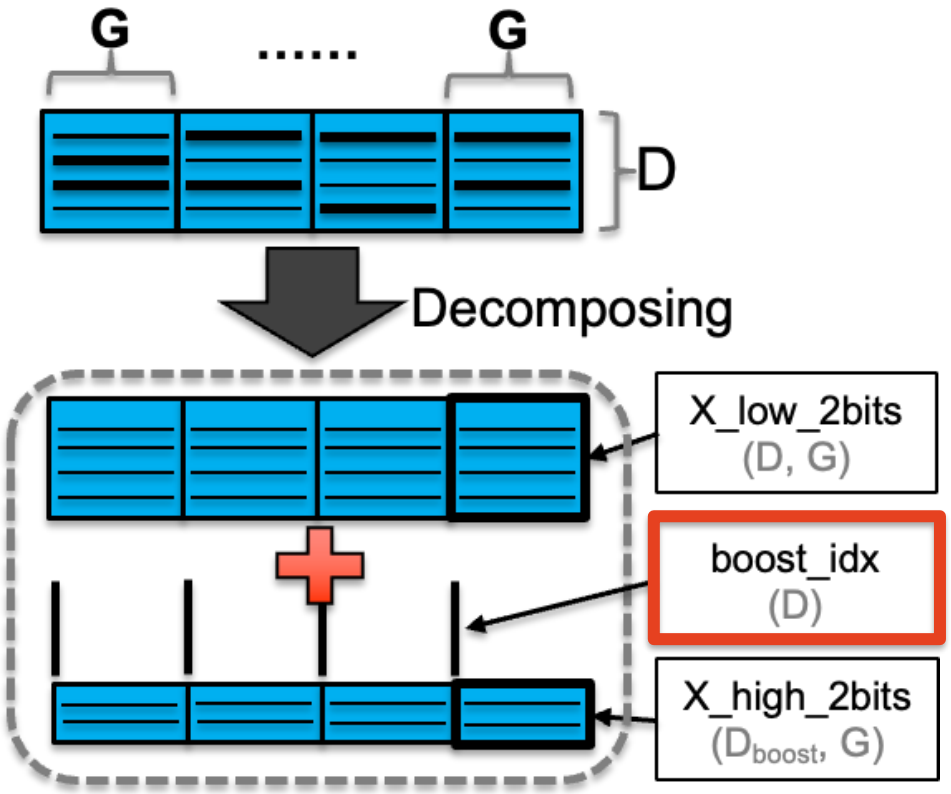
GPU Kernel for Kitty-Attention

- **Dense-Sparse Decomposition**
  - Dense 2-bit pages
  - Sparse 2-bit pages



GPU Kernel for Kitty-Attention

- **Dense-Sparse Decomposition**
  - Dense 2-bit pages
  - Dense 2-bit pages (smaller size)



- **Triton GPU Kernel Design**

Algorithm 1 Dequantize_KeyCache_Page()

shifts = [0, 2, 4, 6, 0, ...]
Input: Quantized cache $\mathbf{C}$, metadata $\mathbf{M}$, boosted channel number D_{boost}
Output: Dequantized key page $\mathbf{K}_{\text{fp16}} \in \mathbb{R}^{D \times T}$
scale, zero_point $\leftarrow$ LoadMeta($\mathbf{M}$)
boost_idx $\leftarrow$ Load($\mathbf{C}$)
boost_mask $\leftarrow$ (boost_idx $\leq D_{\text{boost}}$)
#$\mathbf{X}_{\text{low}}$ is an uint8 tensor with shape (D, T).
$\mathbf{X}_{\text{low}} \leftarrow$ LoadLowBits($\mathbf{C}$)
$\mathbf{X}_{\text{low}} \leftarrow (\mathbf{X}_{\text{low}} \gg \texttt{shifts}) \,\&\, 0x3$
#$\mathbf{X}_{\text{high}}$ is an uint8 tensor with shape (D, T).
$\mathbf{X}_{\text{high}} \leftarrow$ LoadHighBits($\mathbf{C}$, boost_idx, boost_mask)
$\mathbf{X}_{\text{high}} \leftarrow (\mathbf{X}_{\text{high}} \gg \texttt{shifts}) \,\&\, 0x3$
Combining and dequantization.
$\mathbf{X} \leftarrow \mathbf{X}_{\text{low}} \mid (\mathbf{X}_{\text{high}} \ll 2)$
$\mathbf{K}_{\text{fp16}} \leftarrow \mathbf{X} \odot \text{scale} + \text{zero\_point}$
return $\mathbf{K}_{\text{fp16}}$

Results & Conclusions

Accuracy Evaluation Setups

- **Flexible Accuracy Simulation Framework**
  - Extending the *transformers* (HuggingFace) library via customizing the KV cache implementation
  - Easy to be adapted to various quantization schemes
- **Evaluation Benchmarks**
  - Evaluated Models: Qwen3 (8B, 14B, 32B) and LLaMA3 (8B, 70B)
  - Evaluation Datasets: GSM8K, MATH, HumanEval, GPQA, AIME24/25
  - Measuring the accuracy via *lm-evaluation-harness*

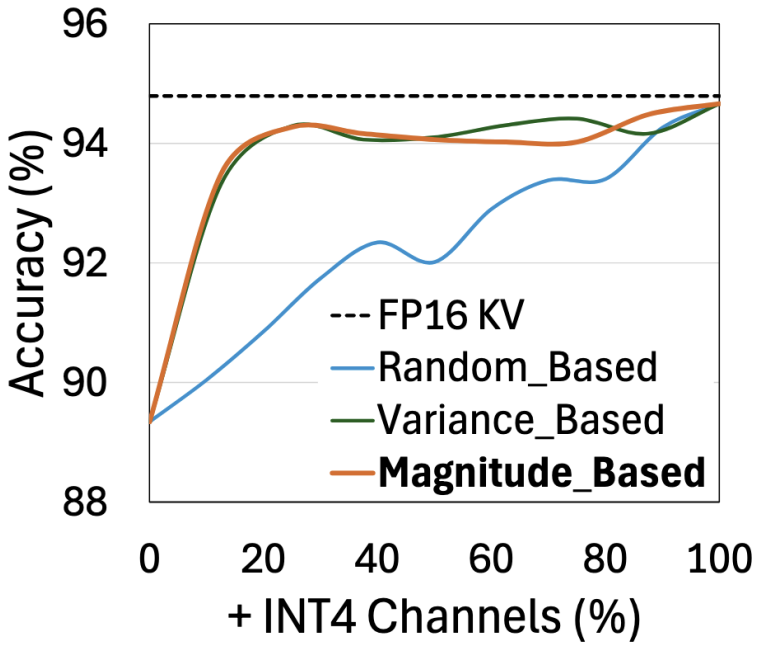
Accuracy Results

- **KIVI-K4V4**
  - Comparable quality to FP16 KV
- **KIVI-K2V2**
  - 2-bit KIVI algorithm
- **KIVI-K2V2***
  - 2-bit KIVI algorithm
  - + FP16 Sink tokens
- **Kitty & Kitty-Pro**
  - 12.5% & 25% channels boosted to 4 bits

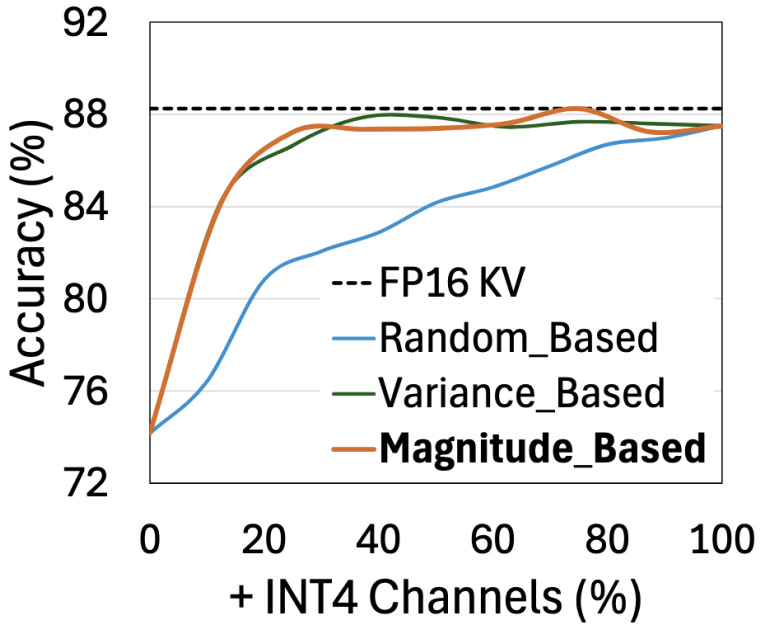
Model / Method		GSM8K	MATH ALGEBRA	HUMAN EVAL	GPQA DIAMOND	Average	Drop
Qwen3-8B	**K16V16**	**94.79**$\pm$0.71	**88.26**$\pm$0.45	**84.82**$\pm$3.72	**40.71**$\pm$2.96	**77.15**	-
	KIVI-K2V2	89.13$\pm$0.51	47.29$\pm$0.20	76.89$\pm$3.11	32.24$\pm$3.16	61.39	-15.76
	KIVI-K2V2*	89.71$\pm$0.63	74.92$\pm$1.83	78.54$\pm$2.56	36.02$\pm$3.27	69.80	-7.35
	Kitty	93.61$\pm$0.58	85.12$\pm$1.54	81.77$\pm$1.89	39.39$\pm$4.18	74.97	-2.18
	Kitty-Pro	94.34$\pm$0.48	88.12$\pm$1.26	81.34$\pm$3.41	40.92$\pm$5.00	76.18	-0.97
Qwen3-14B	**K16V16**	**94.69**$\pm$0.45	**90.68**$\pm$0.14	**86.18**$\pm$2.03	**47.62**$\pm$3.74	**79.79**	-
	KIVI-K2V2	75.82$\pm$1.97	83.66$\pm$1.01	83.74$\pm$0.41	41.50$\pm$0.34	71.18	-8.61
	KIVI-K2V2*	89.56$\pm$0.94	83.74$\pm$0.59	85.98$\pm$1.83	45.24$\pm$2.89	76.13	-3.66
	Kitty	94.67$\pm$0.94	90.31$\pm$0.93	88.21$\pm$2.24	47.11$\pm$4.25	80.08	0.29
	Kitty-Pro	94.90$\pm$0.35	90.54$\pm$0.39	86.38$\pm$1.63	45.92$\pm$2.55	79.44	-0.35
Qwen3-32B	**K16V16**	**91.74**$\pm$0.99	**84.84**$\pm$0.93	**85.98**$\pm$1.22	**48.81**$\pm$1.87	**77.84**	-
	KIVI-K2V2	88.17$\pm$0.38	59.70$\pm$0.81	84.76$\pm$0.61	44.22$\pm$1.19	69.21	-8.63
	KIVI-K2V2*	89.31$\pm$0.91	74.92$\pm$0.45	85.37$\pm$2.44	48.98$\pm$2.55	74.65	-3.19
	Kitty	91.56$\pm$0.58	83.80$\pm$0.62	86.28$\pm$5.79	47.45$\pm$1.02	77.27	-0.57
	Kitty-Pro	90.60$\pm$0.91	83.80$\pm$1.29	88.21$\pm$0.41	51.53$\pm$0.51	78.54	0.70
LLaMA3.1-8B-Instruct	**K16V16**	**76.75**$\pm$1.16	**47.15**$\pm$0.48	**63.96**$\pm$3.11	**26.94**$\pm$5.00	**53.70**	-
	KIVI-K2V2	63.58$\pm$0.63	31.45$\pm$0.62	55.30$\pm$8.72	23.88$\pm$8.57	43.55	-10.15
	KIVI-K2V2*	71.04$\pm$0.83	44.12$\pm$0.62	56.71$\pm$4.88	23.67$\pm$3.88	48.89	-4.81
	Kitty	75.99$\pm$0.96	45.97$\pm$1.57	60.00$\pm$5.85	25.41$\pm$3.47	51.84	-1.86
	Kitty-Pro	75.51$\pm$1.06	47.37$\pm$0.90	61.65$\pm$5.43	25.82$\pm$3.78	52.59	-1.11
LLaMA3.3-70B-Instruct	**K16V16**	**94.92**$\pm$0.30	**71.58**$\pm$1.12	**83.13**$\pm$2.03	**45.92**$\pm$2.55	**73.89**	-
	KIVI-K2V2	93.91$\pm$0.56	66.05$\pm$0.76	79.07$\pm$2.24	45.07$\pm$5.44	71.03	-2.86
	KIVI-K2V2*	94.77$\pm$0.30	69.64$\pm$0.81	82.72$\pm$2.85	47.11$\pm$2.38	73.56	-0.33
	Kitty	95.05$\pm$0.63	70.35$\pm$0.59	83.13$\pm$1.02	45.58$\pm$3.23	73.53	-0.36
	Kitty-Pro	95.00$\pm$0.23	70.77$\pm$0.59	82.72$\pm$0.81	46.94$\pm$3.06	73.86	-0.03

Ablation – Accuracy Recovery

- **Monotonic Accuracy Improvements.**
  - Higher channel boost rates -> Higher accuracy
- **Necessity of Selection Heuristic**
  - **Heuristic-guided** precision boost yields greater benefits than the random counterpart



(a) GSM8K

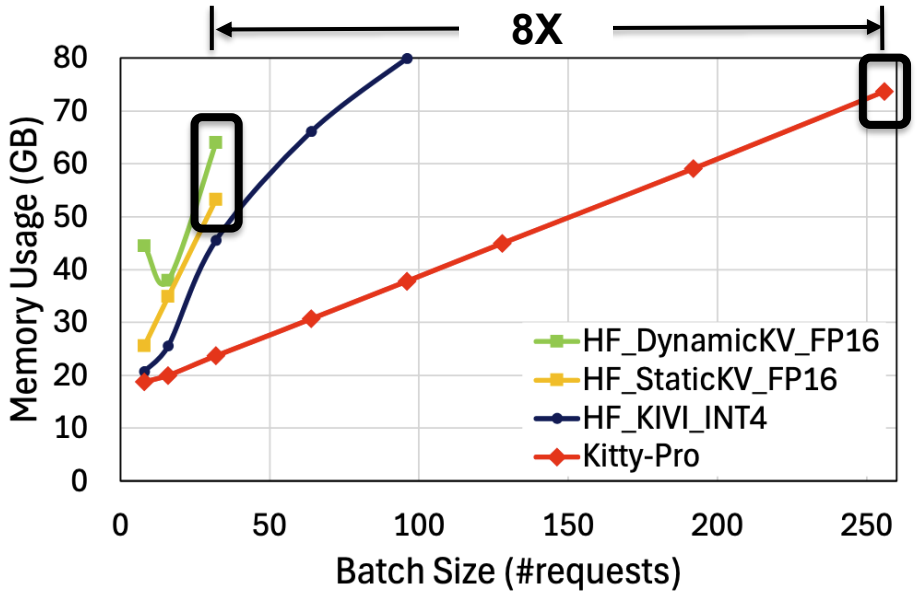


(b) MATH-Algebra

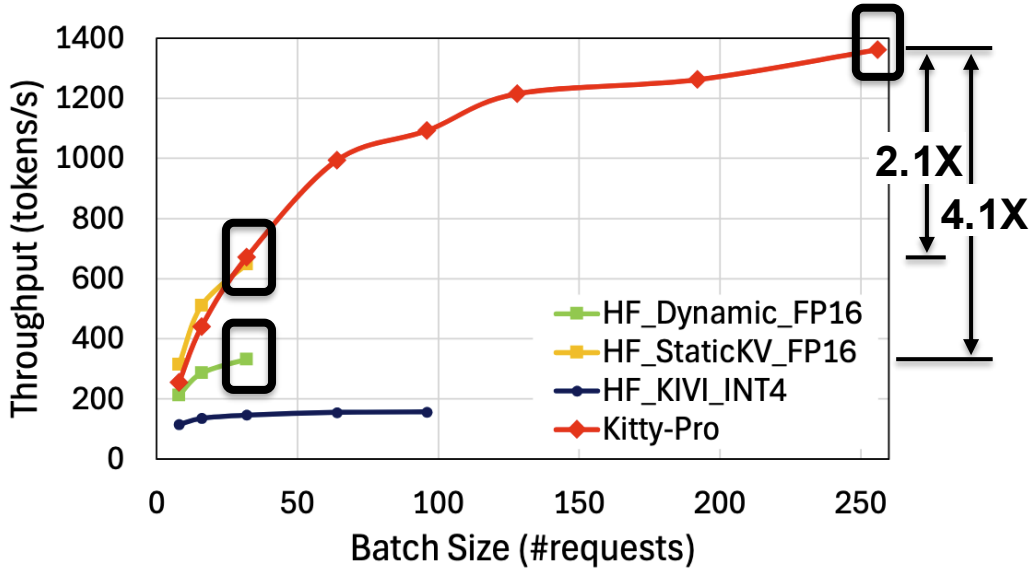
Results - E2E Inference

- **Qwen3-8B on a NVIDIA A100@80GB**
  - HF_DynamicKV_**FP16** (Default HF)
  - HF_StaticKV_**FP16** (Faster)
  - HF_KIVI_**INT4** (Inspired by KIVI)
  - Kitty-Pro_**INT2** (***This paper***)

- **Inference Efficiency**
  - ***8X*** batch size compared to FP16 (256 vs 32)
  - ***2.1X ~ 4.1X*** decoding throughput



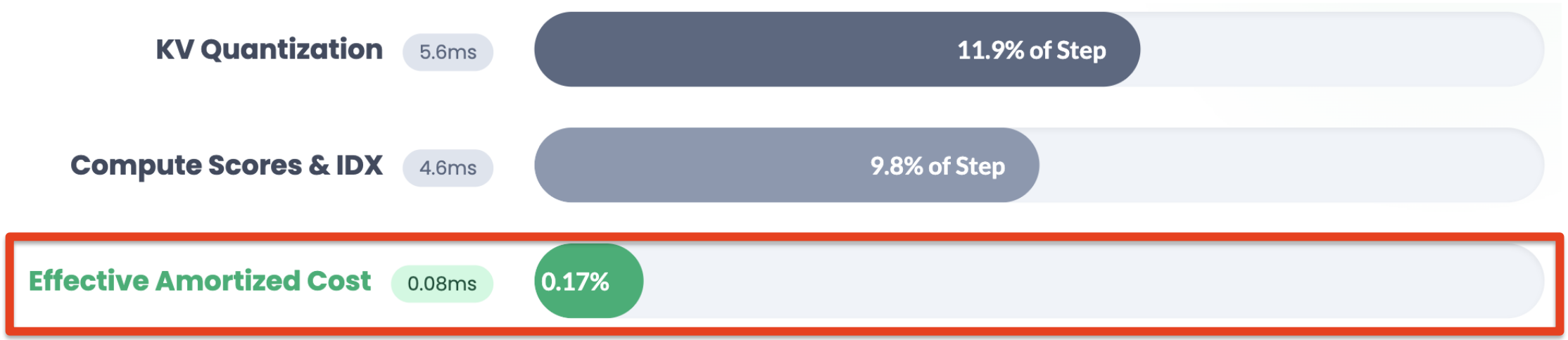
(a) GPU Memory Usage.



(b) Inference Throughput.

Amortized Runtime Overhead

- **Profiling Results on Qwen3-8B (BS=32)**
  - Decoding step time: 47 ms
  - KV quantization: 5.6 ms (every 128 steps)
  - Ranking & Selecting channels: 4.6 ms (every 128 steps)
- **Negligible Amortized Overhead**
  - **0.17%** of overall step time



Conclusions

- **Contributions**
  - Revealed **critical accuracy drops for 2-bit KV** cache quantization.
  - Introduced **channel-wise precision boost**, which significantly outperforms prior 2-bit KV quantization algorithms in accuracy with high memory efficiency.
  - Proposed the corresponding **system designs**, results in 2.1X → 4.1X higher inference throughput compared to the FP16 baseline with same memory budget.

- **Source Code** https://github.com/Summer-Summer/Kitty