Architecture-Conditioned KV Compressibility

Anonymous authors
Paper under double-blind review

Abstract

KV-cache compression is usually evaluated as a deployment method, but a prior question is often left implicit: which native cache representations contain structure worth compressing? We study KV compressibility as a conditional property of representation, architecture, context length, domain, and tensor folding. The main audited screen contains nine completed instruction-tuned models from Qwen2.5, Llama, and Mistral, spanning 0.5–22.2B parameters, four domains, four contexts from 512 to 32K, and three seeds: 432 valid cells and 1,296 representation rows. A separate exact-length matrix contains 420 cells from three models, four domains, seven contexts from 512 to 32K, and five seeds. Three results emerge. First, representation ordering is universal in both protocols: pre-rope keys beat post-rope keys, which beat values, in 432/432 and 420/420 paired cells. Second, qtt compatibility is architecture- conditioned and non-monotone: Qwen2.5-7B reaches a 7.32x median pre-rope-key qtt ratio in the main screen, while neighboring Qwen2.5-3B and 14B are near 1x; the protocol-separated Qwen2.5-1.5B exact-context matrix reaches 26.82x. Third, long context amplifies representation mismatch: from 512 to 32K, rank-16 error rises by 0.0246 for pre-rope K, 0.1448 for post-rope K, and 0.0819 for V. The paper provides a structural screening prior, not a claim of resident-memory or latency improvement.

1 Introduction

Long-context inference is often limited by the KV cache: every generated step must retain and read keys and values from previous tokens. Existing work reduces this cost through eviction, quantization, or cross-layer sharing (Zhang et al., 2023; Xiao et al., 2024; Li et al., 2024; Liu et al., 2024b; Hooper et al., 2024; Liu et al., 2024a; Chang et al., 2025). Those methods answer a deployment question. We ask a preceding measurement question: when does the native cache have a structural regime that makes a compressor worth trying?

The distinction matters because a small post-hoc factorization can coexist with an unchanged resident cache, poor generation quality, or worse latency. It can also succeed on one cache representation and fail after RoPE rotation, on values, or at a longer context. We therefore model compressibility as

$$f(\text{representation}, \text{architecture}, \text{context}, \text{domain}, \text{method}).$$

The goal is a reproducible failure map and selector prior, not a universal claim that TT or QTT storage is ready for deployment.

The study was designed around two plausible intuitions that the measurements contradict. First, a position-wise orthogonal RoPE rotation might be expected to preserve compressibility, yet pre-rope keys are easier than post-rope keys in every paired cell of both large protocols. Second, model size or context might be expected to induce a smooth trend, yet qtt compatibility is sharply non-monotone within Qwen2.5, and longer contexts make fixed-rank error worse, especially for post-rope keys.

Contributions. We make five submission-level claims. (1) We provide a closed 432-cell multi-model structural screen and a closed 420-cell exact-context matrix. (2) We establish

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

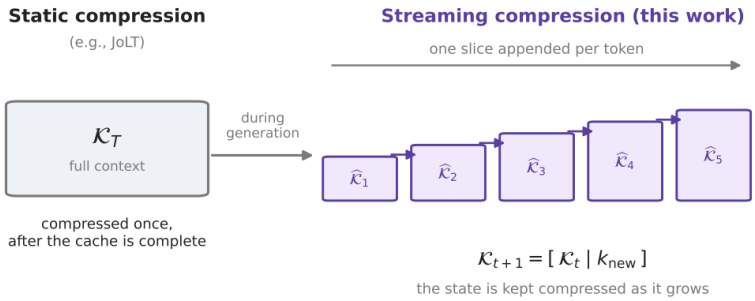

Figure 1: Static versus streaming compression. A static compressor shrinks one finished cache K_T. Streaming compression keeps the cache compressed as it grows, one token at a time.

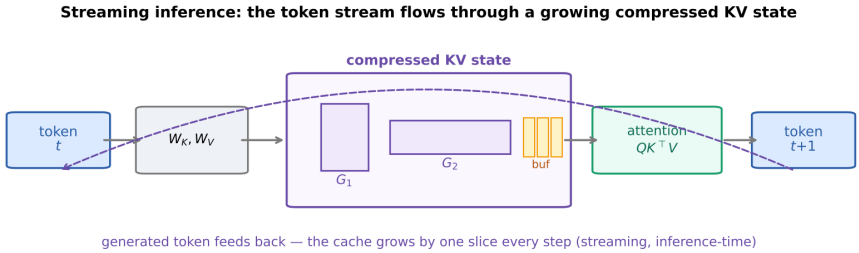

Figure 2: Streaming inference. Each token is projected by W_K, W_V into a new KV slice in the compressed state; attention reads that state, and the produced token feeds back while the state grows by one slice per step.

a robust hierarchy pre-rope K < post-rope K < V for rank-16 error and rank95 diagnostics. (3) We show architecture-conditioned, non-monotone qtt peaks near Qwen2.5-7B and, under a separate protocol, Qwen2.5-1.5B. (4) We separate structural evidence from downstream quality, resident-memory, and latency claims, including a negative shared-basis control. (5) We define and audit an online factor-native decoder boundary that updates a persistent factor state and consumes it without hidden full-history K/V materialization.

Claim boundary. All compression ratios in the structural screen are diagnostics unless the text explicitly refers to quality, resident memory, or latency. A qtt ratio is a factor-size ratio on a stored structural probe, not a decoder memory ratio. We use "factor-native" narrowly: the persistent state stores factors, exact tails, or metadata, and attention consumes them by direct contractions or bounded tiles; an adapter that reconstructs full-history dense K/V remains materializing even when its logical factor files are small.[1] Supplementary material records provenance, reviewer FAQ, F3 quality-grid status, and interpretability audit targets.

[1]The online factor-native result is a systems boundary audit, not a claim that our rank-16 model smoke preserves generation quality or delivers an end-to-end serving Pareto.

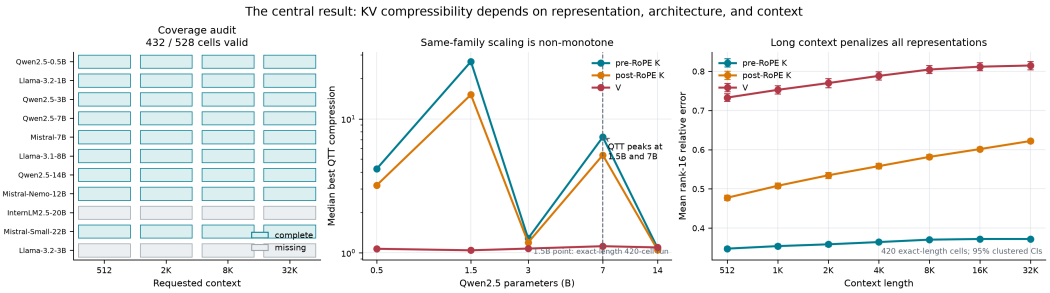

Figure 3: Overview. The main screen contains 432 valid cells; the exact-length matrix contains 420 valid cells. Representation ordering is stable, qtt compatibility is non-monotone across Qwen scales, and rank-16 error rises with context, especially for post-rope K.

2 Related Work and Positioning

KV-cache reduction has several established axes: attention-aware token retention (Zhang et al., 2023; Xiao et al., 2024; Li et al., 2024), low-bit cache quantization (Liu et al., 2024b; Hooper et al., 2024), and cross-layer sharing (Liu et al., 2024a; Chang et al., 2025). Online low-rank compression has also been explored by OjaKV, which adapts a low-rank subspace during prefilling and decoding (Zhu et al., 2025). A contemporaneous tensor study compares Tucker, CP, tensor-train, and t-SVD views of KV caches and reports strong token/feature structure but nearly full-rank head/layer modes (Krishnan & Schulz, 2026). These papers make or analyze concrete compression paths. Our paper instead measures the representation-conditioned screening prior and adds an explicitly audited decoder boundary that distinguishes materializing, bounded-tile, and factor-native attention paths.

We use matrix and tensor diagnostics as probes. Fixed-rank SVD and randomized low-rank methods provide a representation-agnostic baseline (Halko et al., 2011; Brand, 2006); tensor-train and quantized tensor-train factorizations provide a structured storage diagnostic (Oseledets, 2011; Khoromskij, 2011; Aksoy et al., 2024). The novelty is not a new compressor, but an audited scaling study that makes representation, architecture, and context conditioning visible before deployment engineering.

3 Experimental Design

The primary model-scaling protocol planned eleven models and completed nine: Qwen2.5-0.5B, Llama-3.2-1B, Qwen2.5-3B, Qwen2.5-7B, Mistral-7B, Llama-3.1-8B, Qwen2.5-14B, Mistral-Nemo-12B, and Mistral-Small-22B. Two planned models, Llama-3.2-3B and InternLM2.5-20B, failed for infrastructure reasons and are excluded rather than imputed. Each completed model contributes four domains, four requested contexts (512, 2K, 8K, 32K), and three document seeds, for 432 valid cells. Each cell records pre-rope K, post-rope K, and V.

The exact-length long-context matrix is protocol-separated. It contains Qwen2.5-1.5B, Llama-3.1-8B, and Mistral-7B; four domains; seven requested contexts (512, 1K, 2K, 4K, 8K, 16K, 32K); and five seeds. All 420 cells reach the requested token count. We use this matrix for seven-point context claims and for the independent representation-ordering replication; we do not merge it into the nine-model scaling screen.

For each representation we report rank-16 relative Frobenius error e_{16}, the rank95 fraction, and the best listed qtt profile at a 5% relative Frobenius target. The qtt calculation uses a stored 128-token structural probe and reports dense parameters divided by factor parameters. Cross-layer sharing is tested by comparing independent per-layer decompositions with a grouped unadapted shared basis, with aligned CKA and Procrustes residuals used only as geometry diagnostics.

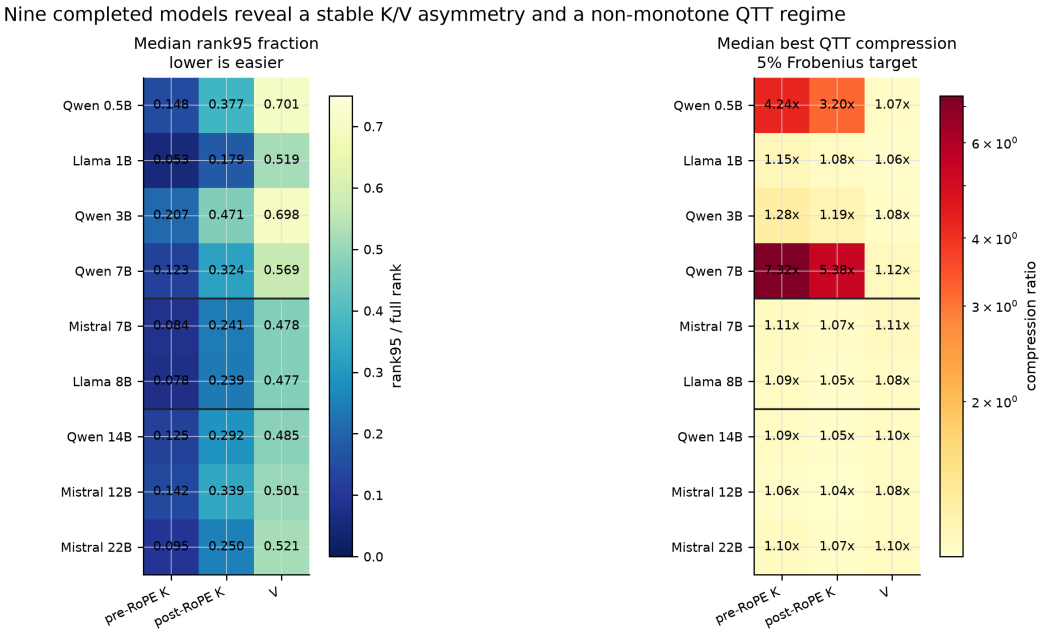

Figure 4: Model-by-representation heatmaps from the 432-cell main screen. The rank95 panels show a stable pre-rope K advantage, while the qtt panel exposes architecture-conditioned peaks.

Uncertainty uses clustered bootstrap units that preserve within-block context structure: model–domain–seed clusters for the main screen (108 clusters) and model–domain–seed clusters for the exact-context matrix (60 clusters). Directional representation and context hypotheses use paired signed-rank tests over the corresponding clusters. The nine-model screen is not presented as a universal scaling law.

4 Results

4.1 Finding 1: representation ordering is universal

Across the 432-cell main screen, mean rank-16 errors are

$$e_{16}(\text{pre-rope}K) = 0.359, \quad e_{16}(\text{post-rope}K) = 0.554, \quad e_{16}(V) = 0.781.$$

pre-rope K beats post-rope K in all 432 paired cells, with mean difference -0.194 and clustered bootstrap 95% CI $[-0.198, -0.190]$. post-rope K beats V in all 432 cells, with mean difference -0.227 and CI $[-0.232, -0.222]$. The independent exact-context matrix replicates the same ordering in all 420/420 cells for both contrasts; mean errors are 0.349, 0.523, and 0.734.

The result rules out a common shortcut: RoPE is orthogonal at each position, but it is not one shared orthogonal transform over all token positions. Position-dependent rotation changes the global token-by-feature matrix and inflates the structural problem for keys. Values remain the hardest spectral bottleneck under the same rank budget.

4.2 Finding 2: QTT compatibility is architecture-conditioned

QTT compatibility is not monotone in parameter count. In the main Qwen2.5 screen, median pre-rope-K qtt ratios are 4.24x at 0.5B, 1.28x at 3B, 7.32x at 7B, and 1.09x at 14B. The 7B peak is stable across its 48 cells: its interquartile range is 7.22–7.50x. Matched-size contrasts are also decisive: Qwen2.5-7B exceeds Mistral-7B in all 48 paired cells, with a

Table 1: Primary screen model summary. Each completed model has 48 cells. QTT values are median best 5% profiles on the 128-token structural probe, not resident-memory ratios.

Model	Params	Mean e_{16}			Median QTT		
	B	pre-K	post-K	V	pre-K	post-K	V
Qwen2.5-0.5B	0.494	0.241	0.462	0.678	4.24x	3.20x	1.07x
Llama-3.2-1B	1.236	0.279	0.505	0.752	1.15x	1.08x	1.06x
Qwen2.5-3B	3.086	0.368	0.551	0.751	1.28x	1.19x	1.08x
Qwen2.5-7B	7.615	0.376	0.539	0.799	7.32x	5.38x	1.12x
Mistral-7B	7.248	0.372	0.567	0.810	1.11x	1.07x	1.11x
Llama-3.1-8B	8.030	0.374	0.559	0.798	1.09x	1.05x	1.08x
Qwen2.5-14B	14.770	0.404	0.581	0.818	1.09x	1.05x	1.10x
Mistral-Nemo-12B	12.248	0.441	0.624	0.814	1.06x	1.04x	1.08x
Mistral-Small-22B	22.247	0.381	0.596	0.808	1.10x	1.07x	1.10x

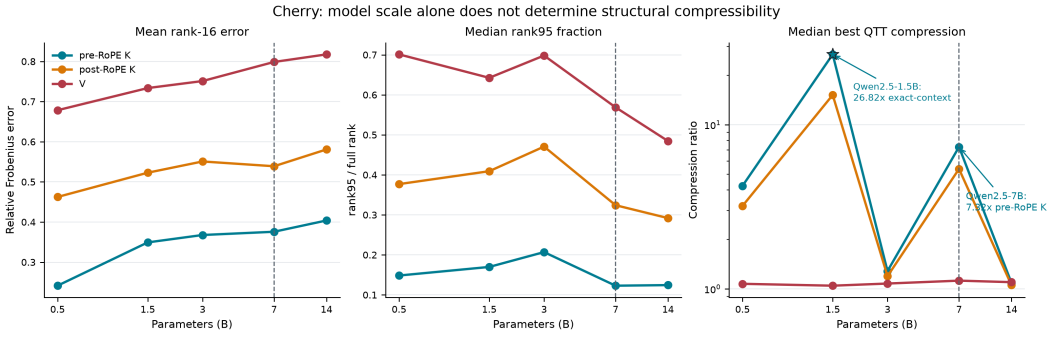

Figure 5: Qwen2.5 scaling. Rank error grows broadly with scale, but qtt compatibility is non-monotone, with protocol-separated peaks near 1.5B and 7B.

mean pre-rope-K difference of 6.30 compression-ratio units and 95% bootstrap CI 6.19–6.42; the contrast to Llama-3.1-8B is 6.31 units, CI 6.21–6.43.

The exact-context matrix adds a protocol-separated anchor. Qwen2.5-1.5B reaches a median pre-rope-K qtt ratio of 26.82x across 140 cells, while Llama-3.1-8B and Mistral-7B reach 1.09x and 1.11x under the same exact-context protocol. Their mean pre-rope-K rank-16 errors are close (0.349, 0.371, 0.368), so the qtt separation is not explained by a large fixed-rank error gap.

The implication is practical. Parameter count is not a safe selector for tensorized cache storage. A deployment pipeline should probe the target architecture and folding directly, and should treat favorable qtt ratios as candidates for held-out decoder experiments rather than as final systems evidence.

4.3 Finding 3: long context amplifies representation mismatch

The exact-length matrix shows that context scaling is representation-dependent. From 512 to 32K tokens, paired rank-16 error increases by 0.0246 for pre-rope K, 0.1448 for post-rope K, and 0.0819 for V. Bootstrap 95% CIs over 60 paired model–domain–seed blocks are $[0.0211, 0.0285]$, $[0.1415, 0.1483]$, and $[0.0758, 0.0878]$. post-rope K and V increase in all 60 blocks; pre-rope K increases in 59/60 blocks and then plateaus from 16K to 32K. The post-rope K penalty is about 5.9x the pre-rope K penalty.

Long-context streaming therefore changes the structural problem rather than just increasing its size. A method chosen on a short cache can fail at 32K even if rank and target error are unchanged; context length must be part of the selector.

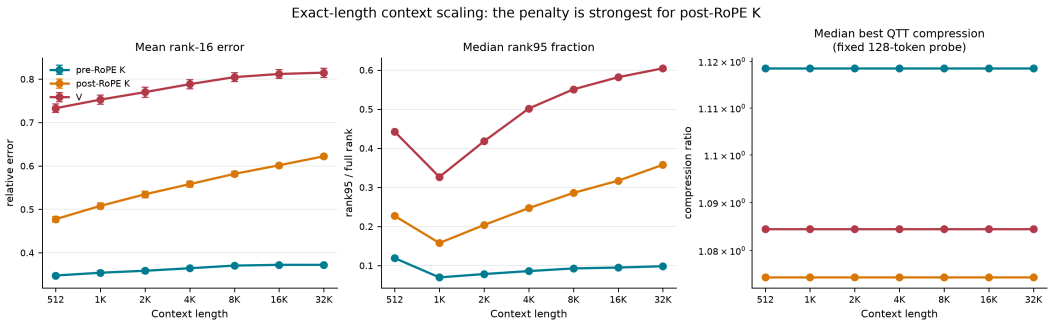

Figure 6: Exact-length context scaling over 420 cells. Rank error rises for all representations, with the largest penalty for post-rope K. QTT ratios are still 128-token structural-probe diagnostics, not full-context memory savings.

4.4 Secondary factors and negative controls

Domain effects are smaller than the main representation and context effects. In the primary screen, the domain range is 0.011 for pre-rope K, 0.018 for post-rope K, and 0.025 for V, while the post-rope-minus-pre-rope representation gap is 0.194 and the 32K-minus-512 post-rope K penalty is 0.1448. Retrieval is the hardest domain, but it does not change the ordering.

High layer similarity also does not justify one raw shared basis. Grouped unadapted bases are worse than independent per-layer factors in all 432 main screen cells for each representation, with mean penalties 0.102 for pre-rope K, 0.085 for post-rope K, and 0.091 for V. The exact-context matrix replicates the negative control in all 420 cells. Aligned CKA is high, but CKA measures subspace geometry; it does not guarantee a shared coordinate system or shared singular value structure.

5 From Structural Signal to a Selector

The data support a conservative screening prior: probe pre-rope and post-rope keys separately; allocate K and V budgets asymmetrically; measure QTT folding on the target architecture rather than extrapolating from size; repeat the probe at the intended context length; and reject raw shared bases unless alignment cost is included in the byte accounting.

This selector is intentionally not claimed as validated transfer. A true selector requires held-out model/domain conditions and a factor-native decoder that consumes the factors without materializing the full dense history.

6 Systems Boundary and Limitations

We include a separate quality and systems pilot only to define the deployment boundary. In that pilot, methods M0–M6 still materialize full K/V, so resident cache remains 14.0 MiB. M6 has the best quality among compressed pilot methods (mean KL 0.215897, top-1 agreement 0.820312, mean PPL ratio 1.408944), but it does not reduce resident memory. Latency also worsens as the structural method gets more elaborate: p99 rises from 55.28 ms for M0 to 84.34, 139.92, and 315.31 ms for M1–M3. Thus

$$\begin{aligned}&\text{structural compressibility} \neq \text{quality preservation},\\ &\text{structural compressibility} \neq \text{resident-memory reduction},\\ &\text{structural compressibility} \neq \text{latency improvement}.\end{aligned}$$

A separate factor-native pilot tests the systems boundary that the structural screen alone cannot establish. The backend maintains an incrementally updated post-RoPE factor state,

Table 2: Takeaway table. The main paper claims structural screening results; deployment claims remain behind their own gates.

Takeaway	Evidence	Boundary
Representation is a gate	pre-rope K < post-rope K < V in 432/432 and 420/420 cells	Not a rank-16 deployment or quality claim
Architecture is not size alone	Qwen2.5-7B reaches 7.32x; separate Qwen2.5-1.5B reaches 26.82x	Not a monotone law or causal covariate claim
Context must be probed	512–32K gives the largest penalty for post-rope K	QTT is not full-context memory saving
Structure is not deployment	Full K/V remains materialized in the pilot	No resident-memory or throughput win is claimed

appends new vectors during one-token decode, and computes attention by direct factor contractions rather than returning full-history dense K/V tensors. In the P1 operator matrix (MHA, GQA, and MQA; contexts 512–8192; ranks 16/32; exact tails 0/64/256), all 54 cells complete: the maximum R2 relative error is 3.51×10^{-4}, one-token versus chunked-last-token error is below 5×10^{-7}, and a fail-fast materialize() guard is never triggered. A Qwen2.5-1.5B exact/full-rank preflight matches dense logits exactly with full_kv_materialized=false. The compressed rank-16 Qwen smoke reduces persistent factor bytes, but its next-token KL is 3.89 and 5.60 at contexts 512 and 2048, respectively; therefore these results establish an online decoder boundary for future serving experiments, not an end-to-end quality or serving claim.

The main limitations follow from this boundary. Nine completed models support strong within-screen contrasts but not a universal scaling law. The 1.5B Qwen point belongs to a separate exact-context protocol. The QTT probe is a stored 128-token structural diagnostic. Architecture covariates are observational, and current systems evidence does not support resident-memory or throughput wins.

7 Conclusion

KV compressibility is conditional, not universal. Across a 432-cell model screen and a 420-cell exact-context matrix, pre-rope K is easier than post-rope K, which is easier than V. QTT compatibility has architecture-conditioned peaks rather than a smooth model-size trend. Long context amplifies the penalty for using the wrong representation, especially after RoPE. These results provide a reproducible screening prior for deciding which factor-native decoder and quality experiments are worth running next.

AI use statement

We used generative AI tools for manuscript editing, LaTeX assistance, and code navigation during preparation of this version. We did not use generative AI tools to generate the experimental measurements, to run statistical analyses without review, or to make autonomous scientific claims. All AI-assisted text and formatting changes were reviewed by the authors, and the numerical results are reported from recorded experiment artifacts. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

Ethics statement

This work studies internal KV-cache structure in existing language models and does not introduce human-subject data collection, user profiling, or model deployment. The main risks are indirect: improved cache compression could lower the cost of long-context systems that may be used in both beneficial and harmful settings. We therefore keep the claims limited to structural diagnostics and report negative systems boundaries rather than presenting unvalidated deployment gains.

Reproducibility statement

The main text specifies the model, domain, context, seed, representation, and statistical protocols used for the closed 432-cell and 420-cell matrices. The supplementary material maps local artifacts, analysis tables, figures, and the quality-grid checkpoint status. The repository version includes the aggregate CSV/JSON files and the LaTeX build inputs needed to rebuild the reported paper figures and PDFs.

References

Doruk Aksoy, David J. Gorsich, Shravan Veerapaneni, and Alex A. Gorodetsky. An incremental tensor train decomposition algorithm. SIAM Journal on Scientific Computing, 46 (2):A1047–A1075, 2024.

Matthew Brand. Fast low-rank modifications of the thin singular value decomposition. Linear Algebra and its Applications, 415(1):20–30, 2006. doi: 10.1016/j.laa.2005.07.021.

Chi-Chih Chang, Chien-Yu Lin, Yash Akhauri, Wei-Cheng Lin, Kai-Chiang Wu, Luis Ceze, and Mohamed S. Abdelfattah. xKV: Cross-layer SVD for KV-cache compression, 2025.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM Review, 53(2):217–288, 2011. doi: 10.1137/090771806.

Coleman Hooper, Sehoon Kim, Hiva Mohammadzadeh, Michael W. Mahoney, Yakun Sophia Shao, Kurt Keutzer, and Amir Gholami. KVQuant: Towards 10 million context length LLM inference with KV cache quantization. In Advances in Neural Information Processing Systems 37, 2024.

Boris N. Khoromskij. $O(d \log N)$-quantics approximation of N-d tensors in high-dimensional numerical modeling. Constructive Approximation, 34(2):257–280, 2011. doi: 10.1007/s00365-011-9131-1.

Rahul Krishnan and Volker Schulz. Tensor decomposition of transformer key-value caches: Spectral structure and format comparison, 2026. arXiv preprint, submitted to SIMAX.

Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. SnapKV: LLM knows what you are looking for before generation. In Advances in Neural Information Processing Systems 37, 2024.

Akide Liu, Jing Liu, Zizheng Pan, Yefei He, Gholamreza Haffari, and Bohan Zhuang. MiniCache: KV cache compression in depth dimension for large language models. In Advances in Neural Information Processing Systems 37, 2024a.

Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhuo Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. KIVI: A tuning-free asymmetric 2bit quantization for KV cache. In Proceedings of the 41st International Conference on Machine Learning, 2024b.

Ivan V. Oseledets. Tensor-train decomposition. SIAM Journal on Scientific Computing, 33 (5):2295–2317, 2011. doi: 10.1137/090752286.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In Proceedings of the Twelfth International Conference on Learning Representations, 2024.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, Zhangyang Wang, and Beidi Chen. H_2O: Heavy-hitter oracle for efficient generative inference of large language models. In Advances in Neural Information Processing Systems 36, 2023.

Yuxuan Zhu, David H. Yang, Mohammad Mohammadi Amiri, Keerthiram Murugesan, Tejaswini Pedapati, and Pin-Yu Chen. OjaKV: Context-aware online low-rank KV cache compression, 2025.