RACE: RoPE-Aligned Attention Compression Based on Query-Key Activation Energies

Anonymous authors
Paper under double-blind review

Abstract

Head-dimension compression can reduce the memory required by the key-value (KV) cache, but existing methods often reconstruct full-dimensional representations during inference, adding significant decoding FLOPs overhead. We formalize the design considerations for converting pretrained attention into a natively reduced-dimensional form that reduces both KV-cache storage and attention computation using a calibration set. Specifically, we consider reductions that (1) apply a shared projection to queries and keys so that they remain in a common reduced feature space, (2) derive this projection jointly from query and key activations to better preserve their interaction, and (3) are compatible with the pretrained RoPE transformation. To obtain a reconstruction-free realization of this reduction, we restrict it to coordinate selection and theoretically show that RoPE compatibility then holds if and only if complete rotary pairs are retained. Building on this characterization, we introduce RACE (**R**oPE-Aligned **A**ttention **C**ompression based on activation **E**nergy), a post-training method that selects rotary pairs using their joint query-key activation energy. We theoretically show that this criterion minimizes a separable, calibration-dependent upper bound on post-RoPE query-key product approximation error. The resulting reductions are folded into the model weights and require neither runtime reconstruction nor specialized kernels. Across three model families and different compression levels, RACE outperforms structured pruning and remains competitive with reconstruction-based methods on language-modeling, zero-shot, and long-context tasks. At 50% KV-cache retention and 256K context, RACE reduces total inference FLOPs by 47.8%, yielding $1.79\times$ higher throughput, 43.2% lower time-to-first-token (TTFT), and 28.1% lower allocated GPU memory in comparison to an uncompressed model. We also show that RACE is compatible with token eviction or compaction as well as quantization based KV-cache compression methods.

1 Introduction

Autoregressive language-model inference is increasingly constrained by the key-value (KV) cache, especially at long contexts. Each attention layer stores a key and value vector for every preceding token, so cache memory grows linearly with context length and with the dimensionality of the cached representations. Reducing the key and value head dimensions offers the possibility of decreasing both KV-cache memory and native attention computation. Realizing these savings is more difficult than merely storing a low-dimensional encoding. Several head-dimension compression methods project keys and values into compact representations but reconstruct full-dimensional keys during inference to effectively apply rotary positional embeddings (RoPE) (Saxena et al., 2024; Chang et al., 2025). Such methods reduce cache storage but introduce decompression and additional arithmetic into the attention path during inference. To obtain the full benefit of head-dimension compression, the attention layer itself must operate directly in the reduced dimensions.

This raises a more fundamental question: *when can a pretrained RoPE-based attention layer be converted into a natively lower-dimensional attention layer without reconstructing the discarded coordinates?* Since attention logits depend on the post-RoPE product $QK^\top$, we view head-dimension reduction as an approximate matrix multiplication (AMM) problem along the shared feature dimension. However, preserving this interaction in the post-RoPE space is not sufficient for native inference. It must be realizable before RoPE. This imposes an additional structural constraint that the

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.

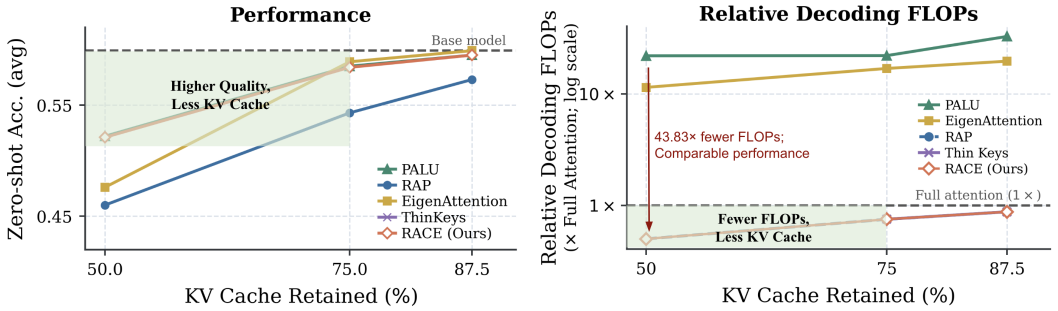


Figure 1: **Quality-efficiency trade-off across KV-cache retention levels on Llama-3.2-3B-Instruct.** Left: average zero-shot accuracy. Right: relative decoding FLOPs on a logarithmic scale, normalized to full attention ($1\times$). The dashed line represent the uncompressed model. RAP and Thin Keys have decoding FLOPs that overlap those of RACE, while Thin Keys' accuracy falls below the displayed y-axis. *RACE combines accuracy competitive with reconstruction-based methods with the low decoding cost of native dimensional reduction.*

Table 1: **Comparison of head-dimension KV-cache compression methods.** RACE combines a shared, jointly selected, RoPE-compatible Q/K reduction with native reduced-dimensional execution. FLOPs are normalized to full attention. See A.4 for derivations of FLOPs.

Method	Excecution		Q/K Reduction Criterion				Efficiency			
	Reconst-ruction free	Custom-Kernel free	Fwd.-only Calibration	Shared Q/K Map	Joint Q/K Signal	Native RoPE	KV Cache Retained	Attn. Params Retained	Prefill FLOPs	Decode FLOPs
Attention (Vaswani et al., 2017)	-	-	-	-	-	-	100%	100%	×1	×1
EigenAttention (Saxena et al., 2024)	✗	✓	✓	✗	✓	✗	50%	69.8%	×0.75	×11.42
PALU (Chang et al., 2025)	✗	✗	✓	✗	✗	✗	50%	58.3%	×0.75	×21.92
Thin Keys (Yao et al., 2026)	✓	✓	✓	✗	✗	✗	75%	50%	×0.75	×0.75
RAP (Xin et al., July 2026)	✓	✓	✗	✓	✗	✓	50%	50%	×0.5	×0.5
RACE (Ours)	✓	✓	✓	✓	✓	✓	**50%**	**50%**	**×0.5**	**×0.5**

reduction must commute with the pretrained rotary transformation. We characterize this condition exactly for coordinate-selection reductions.

These observations define an admissible family of native Q/K reductions: queries and keys can share a reduced feature space, and the reduction must respect the pretrained RoPE structure. The remaining question is which admissible dimensions to retain. We introduce RACE, which ranks complete rotary pairs using joint activation energy for each RoPE pair m, $E_m = \|Q_m\|_F^2 + \|K_m\|_F^2$, and retains the highest-energy pairs. This criterion minimizes a separable, calibration-dependent upper bound on post-RoPE query-key product approximation error. The resulting Q/K reduction, together with a calibrated value subspace, is folded into the attention projections, enabling RoPE, caching, and attention to operate entirely in the reduced dimensions without runtime reconstruction or specialized kernels. Summarizing our contributions:

- **Constrained-AMM formulation.** We formulate native attention compression as preservation of the post-RoPE query-key product subject to pre-RoPE realizability constraint.
- **Structural Characterization.** We prove that a coordinate-selection reduction can commute with the pretrained RoPE if and only if it retains complete rotary pairs, characterizing the admissible family of coordinate reductions.
- **Forward-only pair selection within the family.** RACE ranks RoPE pairs using joint query-key activation energy, minimizing a separable calibration-dependent upper bound.
- **Empirical validation.** Across multiple model families and compression levels, RACE remains competitive with reconstruction-based methods while reducing KV-cache memory, attention cost, parameters, TTFT, and runtime overhead without specialized kernels.

2 Related Work

Compression along other KV-cache axes. KV-cache memory can be reduced along the token, precision, or head-dimension axes as shown in Fig. 8. Token-level methods reduce the number of cached positions (S in Eq.7), through eviction, retention, or compaction (Zhang et al., 2023; Xiao et al., 2024; Li et al., 2024; Cai et al., 2024; Fu et al., 2025; Shi et al., 2025). Quantization methods instead reduce the number of bits used for each cached element, (b in Eq.7), (Liu et al., 2024; Hooper et al., 2024; Wang et al., 2025). Our method preserves the retained token positions and reduces the

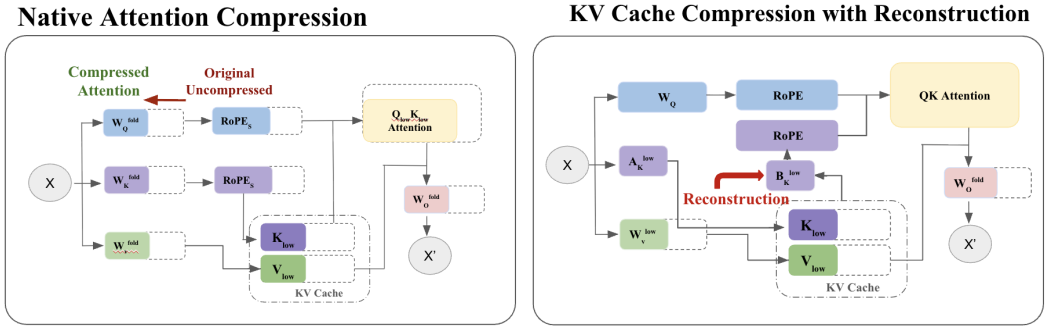


Figure 2: **Native attention compression along the head-dimension.** Reconstruction-based methods (right) may cache low-dimensional representations yet return to the original feature dimension during inference. This is often done using A_K^{low} and B_K^{low} that are decomposition of W_K. We instead seek a native reduction (left) that can be applied before RoPE and folded into the pretrained projections, so that attention is executed natively in the reduced dimension. Dashed boxes indicate the corresponding full-dimensional operations.

number of elements in every key and value vector. It is therefore complementary to both token reduction and quantization; we evaluate representative combinations in Sec. 4.3.

Head-dimension KV-cache compression. Head-dimension methods reduce the width of the cached representations (d_k and d_v in Eq.7) for keys and values respectively. EigenAttention (Saxena et al., 2024) derives low-rank subspaces from calibrated activations, whereas PALU (Chang et al., 2025) factorizes attention projection weights. The dense low-rank compression subspaces in these and similar methods are not generally invariant under the pretrained RoPE transformation and so these methods recover full-dimensional keys before applying RoPE. This can be compute-intense and so, methods like PALU (Chang et al., 2025) mitigate this by using a custom kernel that does not materialize the full-dimensional reconstructed key tensor. However the inference FLOPs remains dominated by the dense reconstruction basis.

Thin Keys (Yao et al., 2026) avoids key reconstruction by reducing the shared Q/K dimension, but retains all dimensions of Values. A concurrent preprint, RAP (Xin et al., July 2026), performs RoPE-aligned structured parameter pruning using empirical-Fisher scores computed from gradients of W_K and arrives at a similar RoPE-aligned native architecture from a different starting point to RACE. While RAP approaches the problem as structured parameter pruning, RACE begins from the problem of constrained-AMM as shown in Sec.3.1. The methods can therefore produce similar reduced-dimensional architectures, but differ in their compression objective and calibration signal. Additionally, our calibration is forward-only and does not depend on a task loss or the batch-sensitivity of Fisher estimates (Tenison et al., 2026). Figure 1 compares our solution to previous techniques in terms of accuracy and FLOPs across a range of compression levels.

Methods including MatryoshkaKV (Lin et al., 2025), STAR-KV (Bhatnagar et al., 2026), ThinK (Xu et al., 2025), and OjaKV (Zhu et al., 2026) additionally study heterogeneous ranks or input-dependent compression. These methods primarily address *where* or *when* to allocate compression across layers, heads, or inputs. Our work instead addresses *how* to construct a fixed, RoPE-compatible reduced representation that operates natively without runtime selection, adaptation, or full-dimensional reconstruction.

3 METHOD

Problem Formulation. Consider an attention head with pre-RoPE activations $Q, K \in \mathbb{R}^{S\times d_k}$ and $V \in \mathbb{R}^{S\times d_v}$. Given target dimensions $r_k < d_k$ and $r_v < d_v$, we seek reduced-dimensional representations

$$Q' = QP_Q, \qquad K' = KP_K, \qquad V' = VP_V, \tag{1}$$

with $P_Q, P_K \in \mathbb{R}^{d_k\times r_k}$ and $P_V \in \mathbb{R}^{d_v\times r_v}$, such that the resulting layer stores only K' and V' in the KV cache and performs attention directly in dimensions r_k and r_v, without reconstructing the discarded coordinates at inference.

For native reduced-dimensional execution, the Q/K reductions must ultimately be applied to the *pre*-RoPE activations so that they can be folded into the query and key projection weights. However, the quantity that determines attention logits is the *post*-RoPE query-key product. We therefore first characterize the low-dimensional Q/K representation that preserves this interaction, and then determine when it can be realized equivalently before RoPE. Finally we introduce RACE, a criterion that satifies these conditions and effectively convert a pretrained attention to a natively reduced-dimensional attention.

3.1 Native Attention Compression as Constrained AMM

Let $\widetilde{Q} = \mathrm{RoPE}(Q), \widetilde{K} = \mathrm{RoPE}(K)$. Since attention logits depend on the post-RoPE product $\widetilde{Q}\widetilde{K}^\top$, we formulate Q/K head-dimension reduction as an approximate matrix multiplication (AMM) problem. We are seeking a column-orthonormal reduction $P \in \mathbb{R}^{d_k \times r_k}$ such that,

$$(\widetilde{Q}P)(\widetilde{K}P)^\top = \widetilde{Q}PP^\top\widetilde{K}^\top \approx \widetilde{Q}\widetilde{K}^\top. \tag{2}$$

This is consistent with classical matrix-sketching and AMM results, where a common lower-dimensional map is applied along the shared dimension of the two operands (Woodruff, 2014; Cohen et al., 2016). Detailed explanation is given in A.5.

This view motivates two design principles. First, *queries and keys should be represented in a shared low-dimensional subspace*, since their interaction is evaluated along the same feature dimension. We therefore seek a common reduction P_{QK} for both operands. Second, when this subspace is chosen from calibration data, *its selection criterion should reflect both matrices whose product is being approximated.* Because the target interaction depends jointly on $\widetilde{Q}$ and $\widetilde{K}$, we construct P_{QK} from statistics of both query and key activations rather than either operand in isolation as in the concurrent work, RAP(Xin et al., July 2026) or ThinKeys(Yao et al., 2026). This is validated empirically in Fig.12.

These observations specify how the Q/K interaction should be reduced, but not yet whether the reduction can be executed natively before RoPE to get the desired systems benefits. Applying P_{QK} only after RoPE would still require full-dimensional query and key projections and full-dimensional rotary encoding, reducing the KV cache width without fully reducing the attention computation as in reconstruction-based methods. *Native execution therefore requires the shared reduction to be realizable before RoPE*. Section 3.1 establishes how the query-key interaction should be reduced, but not whether the shared reduction can be realized before RoPE. Section 3.2 characterizes the additional structural condition under which this is possible.

3.2 RoPE-Compatible Query–Key Reduction

Section 3.1 motivates representing queries and keys through a shared low-dimensional map. To apply this reduction *before* RoPE, it must preserve the corresponding blocks of the pretrained rotary transformation as detailed in A.3.

Let $R_t = \mathrm{diag}(R_{1,t}, \ldots, R_{M,t}) \in \mathbb{R}^{d_k \times d_k}$ be the pretrained RoPE operator at position t, where each $R_{m,t} \in \mathbb{R}^{2\times 2}$ acts on one rotary pair. For a retained pair set $\mathcal{S} \subseteq \{1, \ldots, M\}$, define $R_t^{(\mathcal{S})} = \mathrm{diag}(R_{m,t} : m \in \mathcal{S}) \in \mathbb{R}^{r_k \times r_k}$. Thus, $R_t^{(\mathcal{S})}$ is the restriction of the original pretrained operator to the retained pairs. We say that a reduction $P_{QK} \in \mathbb{R}^{d_k \times r_k}$ is RoPE-compatible if

$$R_t P_{QK} = P_{QK} R_t^{(\mathcal{S})} \qquad \text{for all } t. \tag{3}$$

This condition ensures that applying the reduction before RoPE is equivalent to applying full-dimensional RoPE and then restricting to the retained coordinates. Consequently, query/key projection, rotary encoding, and attention can all be performed directly in dimension r_k. A simple admissible family of P_{QK} is obtained by retaining complete RoPE pairs. The following theorem shows that this family exactly characterizes RoPE-compatible coordinate-selection reductions.

Theorem 1 (RoPE-compatible coordinate selection)**.** *Suppose for every rotary pair m, there exists a position t such that $\sin\phi_{m,t} \neq 0$. A coordinate-selection matrix $P_{QK} \in \mathbb{R}^{d_k \times r_k}$ satisfies Eq. 3 for all t if and only if it selects exactly the coordinates belonging to a union of complete rotary pairs indexed by $\mathcal{S} \subseteq \{1, \ldots, M\}$, with the selected coordinates ordered consistently with $R_t^{(\mathcal{S})}$.*

Algorithm 1 RACE: Post-Training RoPE-Aligned Attention Compression

Require: Pretrained model $\mathcal{M}$, calibration set $\mathcal{D}_{\text{cal}}$, target dimensions r_k, r_v
1: **for** each attention layer **do**
2: Collect post-RoPE Q, K, V activations on $\mathcal{D}_{\text{cal}}$
3: For RoPE pairs, compute $E_m = \|Q_m\|_F^2 + \|K_m\|_F^2$
4: Construct P_{QK} that selects the top-$r_k/2$ pairs based on E_m
5: Fold in P_{QK}. $W_Q^{\text{fold}} = W_Q P_{QK}; W_K^{\text{fold}} = W_K P_{QK}$
6: Compute the top-r_v right singular vectors of V to get P_V
7: Fold in P_V. $W_V^{\text{fold}} = W_V P_V; W_O^{\text{fold}} = P_V^\top W_O$
8: **end for**
9: Optional: recover performance of $\mathcal{M}'$ from $\mathcal{M}$
10: **return** Native attention-compressed model $\mathcal{M}'$

The proof is provided in A.6. The remaining question is which admissible pairs should be retained. Section 3.3 introduces a calibration criterion for ranking the admissible pairs.

3.3 Calibration-Guided Selection Within the RoPE-Compatible Family

Section 3.2 characterizes a family of structurally valid native reductions by restricting selection to complete RoPE pairs. What remains is no longer a compatibility question, but a ranking question over this admissible pairs to determine what pairs should be retained. Because the approximation target in Section 3.1 depends on both queries and keys, we rank candidate pairs using a joint calibration statistic over the two operands.

Let $\mathcal{P}_{\text{rope}} = \{(i_m, j_m)\}_{m=1}^{M}$ denote the RoPE pairs of an attention head, and let $Q_m = Q_{:,\{i_m,j_m\}}, K_m = K_{:,\{i_m,j_m\}}$. We assign pair m the joint query-key energy

$$E_m = \|Q_m\|_F^2 + \|K_m\|_F^2. \tag{4}$$

For target dimension r_k, we retain the $r_k/2$ pairs with the largest energies, $\mathcal{S} = \text{TopK}_{m\in[M]}\left(E_m, \frac{r_k}{2}\right)$ and construct $P_{QK} = [e_i]_{i\in\mathcal{I}_{QK}}$ where $\mathcal{I}_{QK} = \bigcup_{m\in\mathcal{S}}\{i_m, j_m\}$.

The following result provides a calibration-dependent justification for this selection rule.

Proposition 1 (Pair-energy approximation bound)**.** *Let $\widetilde{Q} = \text{RoPE}(Q)$ and $\widetilde{K} = \text{RoPE}(K)$, and let $\widetilde{Q}_m, \widetilde{K}_m$ denote the coordinates associated with RoPE pair m. For any retained pair set $\mathcal{S}$,*

$$\left\|\widetilde{Q}\widetilde{K}^\top - \sum_{m\in\mathcal{S}} \widetilde{Q}_m \widetilde{K}_m^\top\right\|_F \le \sum_{m\notin\mathcal{S}} \left\|\widetilde{Q}_m\right\|_F \left\|\widetilde{K}_m\right\|_F \le \frac{1}{2}\sum_{m\notin\mathcal{S}} E_m. \tag{5}$$

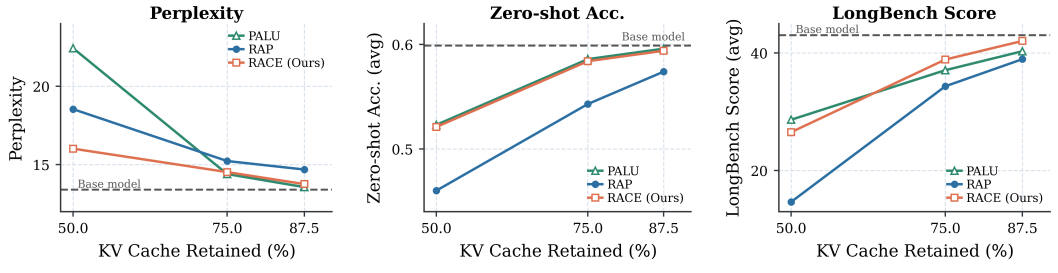
Consequently, among all subsets containing the same number of complete RoPE pairs, selecting the pairs with the largest E_m minimizes the separable upper bound in Eq. 5. Proof of this proposition is given in A.6. Proposition 1 does not imply that the selected pair set minimizes the exact post-RoPE query-key product error. Rather, it shows that the proposed top-energy rule exactly minimizes a worst-case, separable calibration-dependent upper bound in the right-hand side of Eq. 5. We use this result as a tractable forward-only criterion for choosing within the RoPE-compatible family. We evaluate the robustness of this calibration-dependent selection empirically in Section 4.5.

Value compression. Values are not subject to RoPE, so we follow the standard low-rank activation projection used in prior head-dimension compression methods (Saxena et al., 2024). Given calibration activations $V \in \mathbb{R}^{S_{\text{cal}}\times d_v}$, we retain the top-$r_v$ right singular vectors $P_V \in \mathbb{R}^{d_v\times r_v}$ and compute $V' = VP_V$.

3.4 Folding into Native Reduced-Dimensional Attention

Having obtained the RoPE-compatible query-key selector P_{QK} and the value subspace P_V, we fold both transformations directly into the pretrained attention projections. For queries and keys,

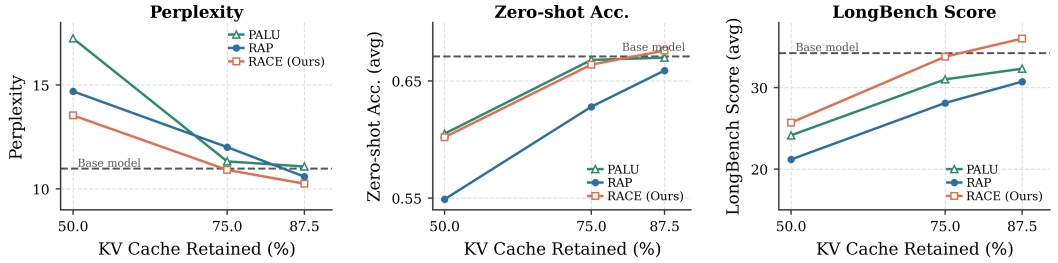
$$Q' = XW_Q^{\text{fold}}, K' = XW_K^{\text{fold}} \qquad \text{where,} \qquad W_Q^{\text{fold}} = W_Q P_{QK}, W_K^{\text{fold}} = W_K P_{QK}. \tag{6}$$



(a) Llama-3.2-3B-Instruct Perplexity

(b) Llama-3.2-3B-Instruct Average zero-shot accuracy

(c) Llama-3.2-3B-Instruct LongBench Score



(d) Qwen2.5-7B Perplexity

(e) Qwen2.5-7B Average zero-shot accuracy

(f) Qwen2.5-7B LongBench Score

Figure 3: **Quality across KV-cache retention ratios on Llama-3.2-3B-Instruct (Top) and Qwen2.5-7B (Bottom).** WikiText perplexity (lower is better), and mean accuracy across 15 zero-shot tasks and mean LongBench score on 16 english tasks (higher is better).

Q' and K' are r_k-dimensional queries and keys to which the restricted RoPE operator from Section 3.2 is applied. Hence, neither the discarded query/key coordinates nor a full-dimensional RoPE representation is materialized during inference.

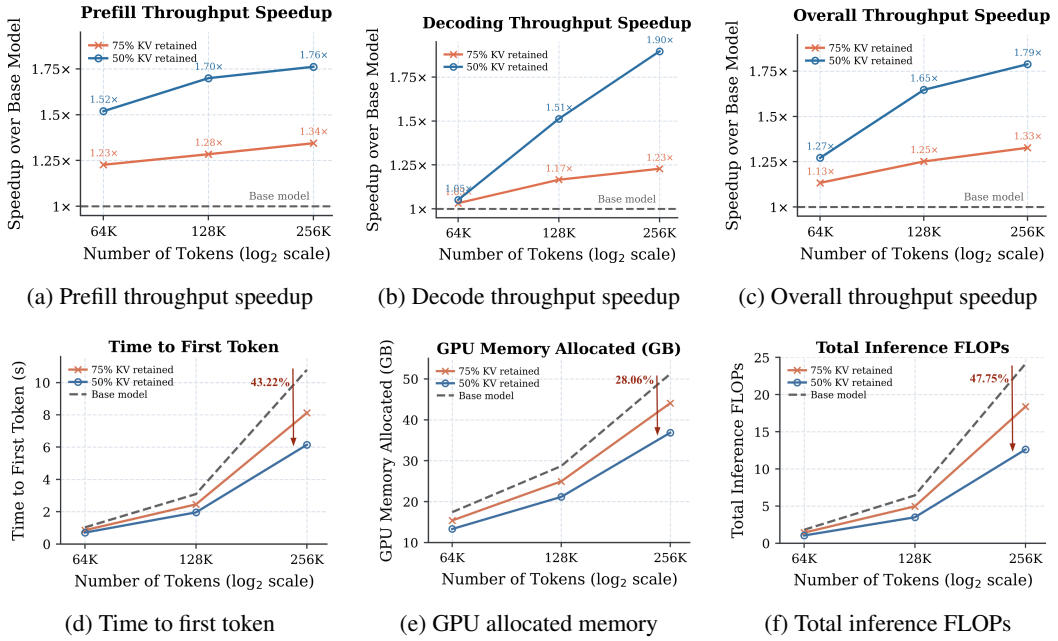
For values, we similarly fold the calibrated subspace into the value and output projections - $W_V^{\text{fold}} = W_V P_V, W_O^{\text{fold}} = P_V^\top W_O$ and $V' = XW_V^{\text{fold}}, Y' = (A'V')W_O^{\text{fold}}$ where A' is computed entirely in the reduced dimension r_k using the retained RoPE pairs.

This folding converts the original pretrained attention layer into a native architecture with query/key dimension r_k and value dimension r_v. The resulting model stores only reduced-dimensional keys and values and performs RoPE, query-key comparison, value aggregation, and output projection without reconstructing the discarded dimensions. Consequently, the reduction decreases KV-cache storage, attention arithmetic, and attention projection parameters simultaneously.

Recovery. For aggressive compression, we recommend a short one-time post-fold recovery stage using KL distillation from the uncompressed model (Fig. 10). We update the reduced attention parameters $W_Q^{\text{fold}}, W_K^{\text{fold}}, W_V^{\text{fold}}, W_O^{\text{fold}}$ while keeping the rest of the model frozen. Recovery is performed directly on the natively compressed checkpoint using 1M FineWeb-Edu tokens. Its effect is modest at upto 75% KV retention but substantial at 50%, where it restores much of the performance. Therefore, while moderate compression requires little or no adaptation; for 50% retention, we recommend a short one-time recovery stage for RACE and RAP.

4 Experiments and Results

Implementation Details. We implement RACE in PyTorch and Hugging Face Transformers. Calibration uses 32,768 FineWeb-Edu tokens, with statistics and SVDs computed in FP32 during calibration and inference is performed in BF16. We respect each architecture's physical rotary-pair layout and retain the original frequency indices and RoPE scaling rules; selected pairs are not reindexed as consecutive frequencies. Compressed checkpoints resize the attention projections and cache dimensions while maintaining the $\frac{1}{\sqrt{d_k}}$ scaling, and run using standard layers and PyTorch attention operations. Thus, RACE requires no specialized CUDA/Triton kernel. While all the re-



(a) Prefill throughput speedup (b) Decode throughput speedup (c) Overall throughput speedup

(d) Time to first token (e) GPU allocated memory (f) Total inference FLOPs

Figure 4: **Long-context inference efficiency on Llama-3.2-3B-Instruct.** The top row reports prefill, decoding, and overall throughput speedups over the uncompressed model (higher is better). The bottom row reports time to first token, allocated GPU memory, and total inference FLOPs (lower is better). Results are shown at 75% and 50% KV-cache retention for prompt lengths of 64K, 128K, and 256K tokens, with the output length fixed at 128 tokens. These results are average over 10 runs.

sults in Sec.4 were reported with SDPA attention backend, Tab.2 shows systems results with eager and flash attention backends. Although different kernels optimize for wall-clock time, the number of FLOPs remian the same. Although optional, all the results of RACE and the concurrent work, RAP involve recovery with KL-distillation over approximately 1M FineWeb-Edu tokens as detailed in A.7.

4.1 Quality at Matched KV-Cache Budgets

We evaluate Qwen2.5-7B, Llama-3.2-3B-Instruct, and Granite-4.1-3B against RoPE-compatible implementations of PALU-GLRD (Chang et al., 2025), EigenAttention(Saxena et al., 2024), Thin Keys(Yao et al., 2026) and concurrent work, RAP (Xin et al., July 2026). The full results are shown in Table 3 in the Appendix. Fig.3 shows the same on Qwen2.5-7B and Llama-3.2-3B-Instruct for RACE, the concurrent work and structured pruning baseline: RAP, and the reconstruction-based baseline: PALU. We finetune reduced attention weights of both RAP and RACE for 1M tokens of FineWeb-Edu using KL distillation from the respective uncompressed model for fair comparison. We report WikiText perplexity, mean accuracy across 15 zero-shot tasks (Tab.3), and LongBench score average over 16 English tasks (Tab.4). At 75% KV-cache retention, RACE preserves 98.5%, 96.5%, and 97.5% of the uncompressed zero-shot accuracy on Qwen, Granite, and Llama, respectively. Across compression levels, on Llama and Qwen models (Figure 3), RACE closely matches PALU on zero-shot accuracy and LongBench without runtime reconstruction. RACE consistently outperforms RAP. RACE also achieves the lowest perplexity at 50% and 75% retention.

4.2 Measured Systems Benefits

Reducing KV-cache storage does not automatically reduce inference cost. As discussed in Section 1, this requires native reduced attention. Such native reduced attention methods like *RACE eliminates reconstruction at the architectural level, exposing standard reduced-dimensional attention operations that can be supported by general-purpose backends*. Figure 4 shows that these measured gains become larger with prompt length as attention accounts for a greater fraction of inference cost. At 256K tokens, 50% retention achieves 1.76×, 1.90×, and 1.79× prefill, decoding, and overall throughput speedups, respectively. It also reduces TTFT by 43.22%, allocated GPU memory by

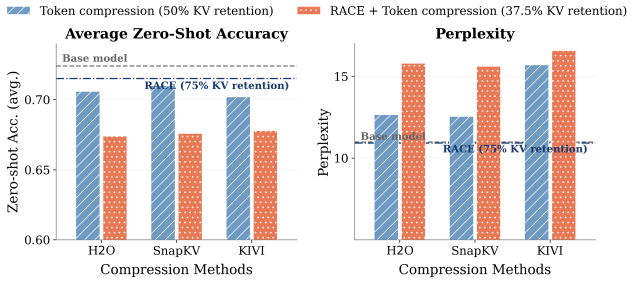


Figure 5: **Composing RACE with token-level compression and quantization.** Combining RACE with H_2O, SnapKV, and KIVI further reduces total KV-cache retention. We report average zero-shot accuracy (left; higher is better) over 5 tasks as described in 5 and WikiText perplexity (right; lower is better).

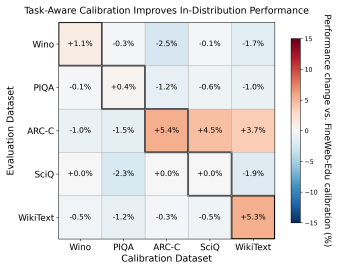


Figure 6: **Effect of calibration data on compressed-model quality.** Values show change on evaluation with the respective calibration data relative to FineWeb-Edu calibration.

28.06%, and total inference FLOPs by 47.75% relative to the uncompressed model. The end-to-end memory reduction is smaller than the 50% KV-cache reduction because total allocated memory also includes model parameters, activations, and other components unaffected by the proposed method. Figure.11 shows the memory reduction with compression on total model parameters, attention parameters and KV cache. While parameter memory is unaffected by sequence length, KV Cache memory grows with it.

4.3 Compatibility with Token-Dimension Compression and Quantization

RACE reduces the KV-cache feature dimension or head dimension and can therefore be combined with methods that compress its token dimension or numerical precision. We apply 75% RACE retention before H2O and SnapKV token compression and KIVI quantization, each with a 50% cache footprint, yielding 37.5% combined retention. Figure 5 shows that applying RACE to an already compressed model causes greater degradation than applying it to the uncompressed model, indicating that the effects of the two compression stages are not independent. Nevertheless, the combinations provide an additional 25% reduction in cache size and feature-dependent attention FLOPs as well as attention parameters. The degradation is smaller with KIVI, particularly in perplexity, suggesting that RACE is especially complementary to quantization. Overall, RACE can augment existing KV-cache methods with further memory, bandwidth, and, for token compression, decoding-FLOP reductions.

4.4 Effect of Calibration Data

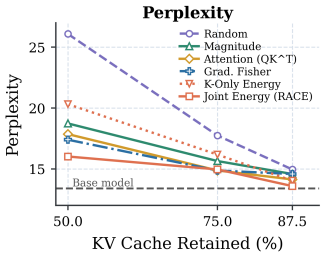
Our selection criterion explained in Section 3.3 is derived from a calibration-dependent upper bound on the query-key approximation error, so the retained subspace is expected to reflect the calibration distribution. Figure 6 evaluates this dependence by varying the calibration dataset and reporting performance relative to FineWeb-Edu calibration. Note that for WikiText, the perplexity change is sign-inverted so that positive values consistently denote improvement. Using the same dataset as the downstream task generally gives the strongest results - upto +5.4% on the evaluated datasets with respect to calibration on FineWen-Edu, and calibration on a related task family may also improve performance as observed in evaluation on arc-c with calibration on SciQ. However, the differences are modest ($< 6\%$) for most tasks. Therefore, *FineWeb-Edu remains a strong generic calibration source, especially when in-distribution task-specific samples are unavailable or the downstream task family is unknown.*

4.5 Ablation: Selection Criteria

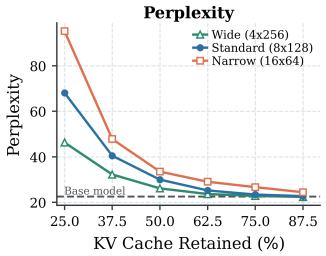
The shared Q/K reduction and Rope-compatibility constraint define a family of native coordinate reductions; the remaining options are on how to rank candidate rotary pairs. We first compare RACE's joint Q/K activation energy with otherwise identical Q-only and K-only selectors. As shown in Fig. 7(b), K-only selection degrades perplexity more strongly, while joint Q/K selection performs best overall. A similar result was observed for Q-only (Table 6). This is consistent with preservation

	Energy (Ours)	Magnitude	Attn. Product	Grad. Fisher
Backprop-free	✓	✓	✓	✗
Task-loss-free	✓	✓	✓	✗
Explicit $O(S^2)$ scoring	✗	✗	✓	✗
Time (sec)	2.717	2.925	11.616	8.290
Throughput (tok./sec)	12061.6	11202.4	2821	3952.7
Memory (GB)	6.142	6.142	10.592	9.218

(a) Calibration cost of alternative data-dependent RoPE-pair scoring criteria.



(b) Perplexity under alternative pair-selection criteria.



(c) Compression behavior of wide, standard, & narrow heads.

Figure 7: **Analysis of RoPE-pair selection and compression structure. (a)** Computational requirements of RoPE-compatible scoring criteria. **(b)** Perplexity under alternative pair-selection criteria. All methods retain complete RoPE pairs. **(c)** Effect of attention-head width at matched model size across compression ratios.

of the query-key interaction: the average normalized error $\|QK^\top - Q_{\mathcal{S}}K_{\mathcal{S}}^\top\|_F / \|QK^\top\|_F$ is 0.1159, 0.1120, and 0.1051 for Q-only, K-only, and joint Q/K selection, respectively.

We further compare RACE with random, magnitude, attention-product, and gradient-Fisher scoring. Random selection is data-oblivious; energy(RACE), magnitude and attention-product use joint Q/K forward statistics; and gradient-Fisher uses joint Q/K gradients from backward-pass on the calibration data. RACE achieves the lowest perplexity across the evaluated retention ratios among these selectors. Table 7(a) further compares calibration cost of methods that uses Rope-compatible shared joint Q/K statistics for selection. Attention-product scoring explicitly incurs $O(S^2)$ computation and gradient-Fisher requires backpropagation and a task loss. Accordingly, athese methods incur substantially higher calibration time and peak memory. Joint QK energy selection therefore offers a favorable trade-off between perfromance and calibration cost.

4.6 Towards Models with Fewer and Wider Heads

If post-training attention compression is anticipated at deployment time, larger head dimensions may be a useful architectural choice during pretraining. We study three matched $\sim$ 245 M-parameter GPT-NeoX-style models pretrained from scratch on FineWeb-Edu for 2.5B tokens: *wide* (4×256), *standard* (8×128), and *narrow* (16×64). All three reach comparable uncompressed WikiText validation perplexity. Figure 7(c) shows that wider-head models retain perplexity more effectively under aggressive RACE compression. At each retention ratio, the models have approximately matched aggregate retained attention dimension, KV-cache memory, attention FLOPs, and compressed attention parameter count; the primary difference is how this capacity is partitioned across heads. Wider heads expose more RoPE pairs from which RACE can select, providing greater flexibility in structured dimensional reduction. Thus, *when post-training attention compression is a foreseeable deployment requirement, allocating attention width across fewer, wider heads during pretraining may yield a more favorable quality-efficiency trade-off.*

5 Conclusion

In this work, we study when and how pretrained RoPE attention can be converted into a natively lower-dimensional form that reduces KV-cache storage, attention computation, and parameters without runtime reconstruction. Viewing Q/K compression as constrained approximate matrix multiplication motivates a shared reduction derived jointly from queries and keys. This reduction has to be RoPE-compatible to enable reconstruction-free compression. We further show that, for coordinate-selection reductions family, RoPE compatibility holds if and only if complete rotary pairs are retained. RACE selects within this admissible family using joint Q/K activation energy, a forward-only criterion that minimizes a separable calibration-dependent upper bound on post-RoPE product error. The resulting reductions are folded directly into the projection weights, enabling attention to execute entirely in the compressed dimensions without specialized kernels. Across multiple model families and compression levels, RACE outperforms evaluated reconstruction-free baselines and remains competitive with reconstruction-based methods while reducing KV-cache memory, attention cost, and latency.

AI USE STATEMENT

We used generative AI tools to assist with editing portions of the manuscript, retrieving related works, generating and debugging experimental and plotting code, and formatting figures and tables. We also used AI tools to discuss and refine aspects of theoretical presentation and interpretation of experimental results. All AI-assisted technical suggestions, mathematical arguments, code, and interpretations were independently reviewed and verified by the authors against the underlying derivations, implementations, and experimental outputs. No AI generated datasets were used in this work. We take responsibility for the final content of this work, including all text, claims, code, and artifacts produced with AI assistance.

ETHICS STATEMENT

The proposed method is intended to reduce the memory and computational cost of long-context inference, which may improve the accessibility and efficiency of deploying language models. As with other efficiency techniques, however, reducing inference cost can also lower the barrier to large-scale model deployment; the downstream societal effects therefore depend on how the resulting models are used. Our work does not modify model objectives, safety mechanisms, or training data, and we do not make claims about changes to model safety, fairness, or robustness beyond the evaluated performance metrics

REPRODUCIBILITY STATEMENT

We provide sufficient implementation and experimental details to reproduce the results in this work. A.7 specifies the RACE calibration and folding procedure, RoPE-pair handling, value compression, optional KL-based recovery, model and benchmark configurations, long-context evaluation settings, hardware/software environment, precision, number of tokens, kernels, and analytical FLOP derivations. Detailed per-task results, selector ablations, recovery settings, and systems measurements are also provided in the A.8.

REFERENCES

Priyansh Bhatnagar, Ashkan Moradifirouzabadi, Se-Hyun Yang, SeungJae Lee, Jungwook Choi, and Mingu Kang. Star-kv: Low-rank kv cache compression via soft thresholding for adaptive rank control, 2026. URL https://arxiv.org/abs/2606.08382.

Zefan Cai, Yichi Zhang, Bofei Gao, Yuliang Liu, Tianyu Liu, Keming Lu, Wayne Xiong, Yue Dong, Baobao Chang, Junjie Hu, and Wen Xiao. Pyramidkv: Dynamic kv cache compression based on pyramidal information funneling. *arXiv preprint arXiv:2406.02069*, 2024.

Chi-Chih Chang, Wei-Cheng Lin, Chien-Yu Lin, Chong-Yan Chen, Yu-Fang Hu, Pei-Shuo Wang, Ning-Chi Huang, Luis Ceze, Mohamed S. Abdelfattah, and Kai-Chiang Wu. Palu: Kv-cache compression with low-rank projection. In *International Conference on Learning Representations*, 2025.

Michael B. Cohen, Jelani Nelson, and David P. Woodruff. Optimal approximate matrix product in terms of stable rank. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*, volume 55 of *Leibniz International Proceedings in Informatics*, pp. 11:1–11:14, 2016. doi: 10.4230/LIPIcs.ICALP.2016.11.

Yu Fu, Zefan Cai, Abedelkadir Asi, Wayne Xiong, Yue Dong, and Wen Xiao. Not all heads matter: A head-level kv cache compression method with integrated retrieval and reasoning. In *International Conference on Learning Representations*, 2025.

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. *arXiv preprint arXiv:2401.18079*, 2024.

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. *arXiv preprint arXiv:2404.14469*, 2024.

Bokai Lin, Zihao Zeng, Zipeng Xiao, Siqi Kou, TianQi Hou, Xiaofeng Gao, Hao Zhang, and Zhijie Deng. MatryoshkaKV: Adaptive KV compression via trainable orthogonal projection. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=BQwsRy1h3U.

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. *arXiv preprint arXiv:2402.02750*, 2024.

Utkarsh Saxena, Gobinda Saha, Sakshi Choudhary, and Kaushik Roy. Eigen attention: Attention in low-rank space for kv cache compression. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15332–15344, 2024.

Dachuan Shi, Yonggan Fu, Xiangchi Yuan, Zhongzhi Yu, Haoran You, Sixu Li, Xin Dong, Jan Kautz, Pavlo Molchanov, and Yingyan Celine Lin. Lacache: Ladder-shaped kv caching for efficient long-context modeling of large language models. In *International Conference on Machine Learning*, 2025.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Irene Tenison, Soumyajit Chatterjee, Fahim Kawsar, and Mohammad Malekzadeh. AdaBet: Gradient-free layer selection for efficient training of deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Hao Wang, Ligong Han, Kai Xu, and Akash Srivastava. Squat: Subspace-orthogonal kv cache quantization. In *Conference on Language Modeling*, 2025.

David P. Woodruff. Sketching as a tool for numerical linear algebra. *Foundations and Trends in Theoretical Computer Science*, 10(1–2):1–157, 2014.

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

Jihao Xin, Tian Lyu, David Keyes, Hatem Ltaief, and Marco Canini. Rap: Kv-cache compression via rope-aligned pruning, July 2026. URL https://arxiv.org/abs/2602.02599.

Yuhui Xu, Zhanming Jie, Hanze Dong, Lei Wang, Xudong Lu, Aojun Zhou, Amrita Saha, Caiming Xiong, and Doyen Sahoo. Think: Thinner key cache by query-driven pruning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=n0OtGl6VGb.

Hengshuai Yao, Xing Chen, Ahmed Murtadha, and Guan Wang. Thin keys, full values: Reducing kv cache via low-dimensional attention selection, 2026. URL https://arxiv.org/abs/2603.04427.

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. H2o: Heavy-hitter oracle for efficient generative inference of large language models. In *Advances in Neural Information Processing Systems*, 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. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 10161–10178, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.494. URL https://aclanthology.org/2026.findings-acl.494/.