# ROKI: RoPE-Aware KV Cache Quantization via Isofrequency Grouping

**Anonymous authors**
Paper under double-blind review

## Abstract

Low-bit KV-cache quantization reduces the memory and bandwidth costs of long-context LLM inference, but post-RoPE per-token quantization is sensitive to large activations that dominate the shared quantization range. Cross-channel rotations can reduce these outliers, yet general cross-pair rotations do not commute with Rotary Position Embedding (RoPE) and therefore cannot be folded into Q/K projection weights. We propose ROKI, a RoPE-aware KV-cache quantization framework based on Isofrequency Commutant Expansion (ICE). Rather than restricting transformations to the fixed RoPE operator, ICE selectively ties nearby low-frequency RoPE pairs, creating isofrequency groups that enlarge the exact RoPE-commuting transformation space from pair-wise to cross-pair rotations. ROKI uses this expanded space to redistribute Q/K activations and applies transformation-aware mixed-precision key quantization based on post-transformation QK contributions. All Q/K transformations are optimized offline and folded into the projection weights, requiring no additional online transformation. At an average 3-bit KV-cache budget, ROKI achieves the highest average LongBench-E score among the evaluated quantized baselines on three Llama/Qwen models while remaining close to FP16. Our ROKI implementation also achieves up to 4.22× decode speedup over the evaluated low-bit baselines at long contexts.

## 1 Introduction

Long-context large language model (LLM) inference incurs substantial memory and bandwidth costs because every decoding step repeatedly accesses the accumulated key-value (KV) cache. Low-bit KV-cache quantization can reduce both cache capacity and memory traffic, but practical methods must preserve accuracy without adding substantial computation to the attention path.

For key-cache quantization, one design choice is whether keys are quantized before or after Rotary Position Embedding (RoPE) (Su et al., 2024). Pre-RoPE quantization, as used by KVQuant (Hooper et al., 2024), preserves the original channel-wise structure of key activations before it is mixed by positional rotation, which is particularly favorable for per-channel quantization. However, cached pre-RoPE keys must undergo positional rotation when they are later used in attention. Post-RoPE caching instead stores position-encoded keys and avoids reapplying RoPE to cached keys during decoding. We therefore target **post-RoPE per-token quantization**, where a shared scale across multiple channels simplifies cache handling but makes quantization sensitive to large activations that dominate the shared range.

Existing methods reshape Q/K activations to improve post-RoPE quantization, but face a trade-off between transformation flexibility and runtime efficiency. Figure 1 summarizes this design space in terms of the placement of quantization and Q/K transformation relative to RoPE. QuaRot (Ashkboos et al., 2024) and SpinQuant (Liu et al., 2025) use post-RoPE rotations for cross-channel redistribution, but because these rotations do not commute with RoPE, they cannot be folded into the Q/K projection weights and must be applied online after RoPE. In contrast, QServe (Lin et al., 2025) and FPTQuant (van Breugel et al., 2026) use RoPE-compatible transformations that can be folded into Q/K projection weights. However, because different RoPE pairs use different frequencies, these foldable transformations are restricted to individual pairs.



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.

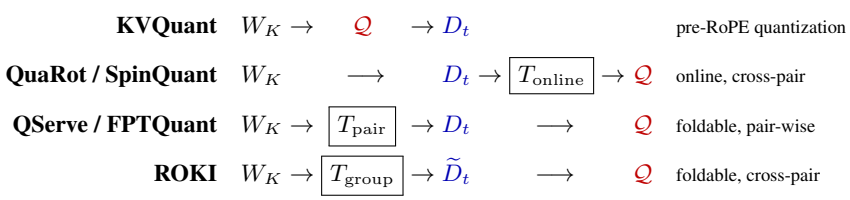



Figure 1: Operation order of representative approaches for key-cache quantization and Q/K distribution shaping around RoPE. $\mathcal{Q}$ denotes key quantization and caching, $D_t$ denotes the original RoPE operator at position $t$, and $\widetilde{D}_t$ denotes the frequency-tied RoPE operator used by ROKI. Red and blue indicate quantization and RoPE operations, respectively. Only the key path is shown for clarity; corresponding transformations are applied to the query path as required to preserve the unquantized QK interaction.

This pair-wise restriction limits outlier redistribution. Pair-wise rotation can balance values within a pair but cannot move magnitude across pairs, while aggressively scaling down a large key pair can amplify the effect of residual key-quantization error through reciprocal query scaling. Our key insight is to *relax the RoPE constraint itself*. We observe that large Q/K activations tend to concentrate in later, low-frequency RoPE pairs, where neighboring frequencies are relatively close. By selectively assigning a common frequency to such pairs, we create an **isofrequency subspace** in which cross-pair rotations commute with RoPE and can therefore be folded into the projection weights. We call this mechanism **Isofrequency Commutant Expansion (ICE)**.

Based on ICE, we propose **ROKI**, a RoPE-aware KV-cache quantization framework. ROKI first redistributes large activations across selected low-frequency pairs through group-wise rotation, allowing subsequent reciprocal scaling to correct only the remaining imbalance rather than serving as the primary mechanism for outlier suppression. ROKI further employs transformation-aware mixed-precision key quantization: after distribution shaping, it measures each pair's QK inner-product contribution and applies a matched permutation to the corresponding Q/K pairs, placing more important key pairs in higher-precision quantization blocks. All Q/K transformations are determined offline and folded into the projection weights, requiring no additional online Q/K transformation.

Across Llama-3.1-8B-Instruct, Qwen3-8B, and Qwen3-14B, ROKI achieves the highest average LongBench-E score among the evaluated quantized baselines under an average 3-bit KV-cache budget, while remaining close to FP16 accuracy. On an RTX 3090, ROKI achieves up to $4.12\times$ and $4.22\times$ decode speedups over KIVI (Liu et al., 2024) and Block-GTQ (Liang et al., 2026), respectively, at long contexts.

Our contributions are:

- **Isofrequency Commutant Expansion.** We selectively tie low-frequency RoPE pairs to enlarge the space of RoPE-commuting, weight-foldable transformations from pair-wise to group-wise rotations.
- **Transformation-aware mixed-precision key quantization.** ROKI estimates pair importance after ICE-based distribution shaping using QK inner-product contribution and jointly permutes corresponding Q/K pairs to map more important key pairs to higher-precision quantization blocks.
- **Efficient post-RoPE inference.** ROKI folds all learned Q/K transformations into projection weights, providing average-3-bit KV caching without additional online Q/K transformations.

## 2 Motivation and Key Insight

### 2.1 Limitations of Pair-Wise RoPE-Compatible Transformations

To improve post-RoPE key quantization without runtime distribution-shaping overhead, Q/K transformations should be foldable into the projection weights. RoPE, however, fundamentally restricts the transformations that admit such folding. Let the head dimension be $d = 2m$. RoPE partitions

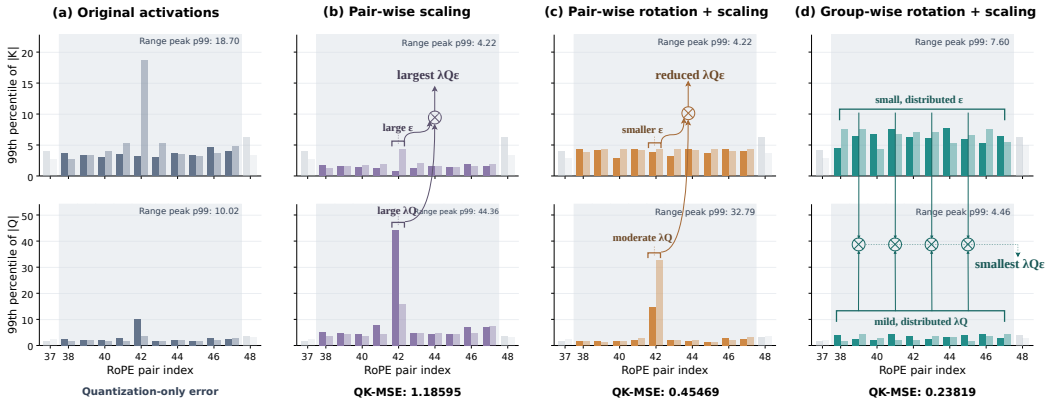



Figure 2: Effect of rotation scope and reciprocal scaling on Q/K activation distributions. Panels show the 99th-percentile magnitudes of K and Q. Pair-wise rotation acts within each RoPE pair, while group-wise rotation also redistributes values across pairs.

each query and key head into $m$ two-dimensional pairs and assigns an angular frequency $\omega_p$ to each pair $p$. Let $R(\alpha)$ denote a two-dimensional rotation by angle $\alpha$. At position $t$, pair $p$ is rotated by angle $t\omega_p$, giving

$$D_t = \mathrm{diag}\left(R(t\omega_0), \ldots, R(t\omega_{m-1})\right). \tag{1}$$

For a Q/K transformation $U$ to be moved across RoPE and folded into the projection weights, it must commute with RoPE at every position: $\forall t, D_t U = U D_t$.

Any two-dimensional rotation within a single RoPE pair commutes with that pair's RoPE rotation. In contrast, a general rotation mixing pairs with different frequencies does not commute with $D_t$. Consequently, under the original RoPE frequencies, foldable rotations are restricted to individual pairs. A general cross-pair rotation cannot be folded through RoPE and must therefore be applied online after RoPE. We provide a formal derivation in Appendix D.

This pair-wise restriction becomes problematic when large activations are concentrated in particular RoPE pairs. A pair-wise rotation can redistribute values between the two components of a pair, but it preserves the pair norm: $\|R(\phi_p)k_p\|_2 = \|k_p\|_2$. It therefore cannot transfer magnitude from a large pair to other pairs. Reciprocal scaling can change the magnitude of a pair, but stronger scaling may be needed when pair-level outliers cannot first be redistributed across pairs. Suppose that a key pair is scaled down while the corresponding query pair is scaled up to preserve their full-precision inner product: $q'_p = \lambda_p q_p, \quad k'_p = k_p/\lambda_p$. Let the reconstructed key after quantization be $\hat{k}'_p = k'_p + \epsilon_p$. Then, ${q'_p}^\top \hat{k}'_p = q_p^\top k_p + \lambda_p q_p^\top \epsilon_p$. Thus, aggressively suppressing a large key pair does not necessarily reduce its contribution to the QK-logit error, because the residual key-quantization error is amplified by the reciprocal query scaling. The issue is therefore not reciprocal scaling itself, but relying on strong scaling when large activations remain concentrated in individual pairs.

Figure 2 illustrates this limitation. Scaling alone reduces the dominant key activation but increases the corresponding query magnitude. Adding pair-wise rotation improves the balance within each pair, yet the dominant pair norm remains localized. In contrast, extending the rotation across multiple pairs can redistribute the large activation more broadly, reducing the reliance on strong reciprocal scaling.

Appendix A.2 (Table 6) quantifies this effect on Llama-3.1-8B-Instruct. Compared with scaling alone, pair-wise rotation followed by scaling reduces QK-MSE by 29.41%. Extending the rotation across multiple pairs further reduces QK-MSE, while also lowering both K and Q activation peaks. These results show that extending the rotation scope across pairs can further reduce quantization error beyond pair-wise transformation. The remaining challenge is therefore to enable such cross-pair redistribution while preserving RoPE compatibility and weight foldability.

| Setting | Group-wise and native freq. | Group-wise and tied freq. (ICE) |
|---|---|---|
| FP16 | $8.5476 \times 10^{-3}$ | $\mathbf{2.589 \times 10^{-4}}$ **(↓ 96.97%)** |
| Quant. | $3.8992 \times 10^{-2}$ | $\mathbf{2.9377 \times 10^{-2}}$ **(↓ 24.66%)** |

Table 1: Effect of frequency tying on group-wise rotation for Qwen3-8B at a 32K context length. Both settings use the same group configuration and group-wise rotation; the native-frequency setting retains the original RoPE frequency of each pair, whereas ICE assigns a shared frequency within each group. The group-wise rotation is applied before RoPE, as required for weight folding.

### 2.2 Low-Frequency Opportunity for Foldable Cross-Pair Rotation

The low-frequency RoPE region is particularly attractive for cross-pair redistribution. Since the angular frequency of RoPE pair $p$ decreases approximately geometrically with the pair index, $\omega_p = \theta_{\text{base}}^{-2p/d}$, neighboring frequencies become increasingly close in the later, low-frequency region, as illustrated in Figure 8. Prior work has observed that trained LLMs allocate substantial Q/K norm to low-frequency RoPE components (Barbero et al., 2025). Consistently, our analysis in Appendix A.3 (Figures 6 and 7) shows that later RoPE pairs tend to exhibit larger Q/K activations and higher QK inner-product contributions. Thus, the region where cross-pair redistribution is most useful also contains pairs whose frequencies can be tied with relatively small frequency perturbations.

However, frequency proximity alone does not enable foldable cross-pair rotation; exact commutation requires assigning a shared frequency within the group, thereby creating an isofrequency subspace that admits RoPE-commuting cross-pair rotations.

For a group $\mathcal{G}$ whose pairs are assigned a shared frequency $\nu_{\mathcal{G}}$, the modified RoPE operator becomes

$$\widetilde{D}_t^{\mathcal{G}} = \operatorname{diag}\left(R(t\nu_{\mathcal{G}}), \ldots, R(t\nu_{\mathcal{G}})\right). \tag{2}$$

Because every pair in $\mathcal{G}$ now undergoes the same positional rotation, a broader class of group-wise rotations can be constructed to commute with $\widetilde{D}_t^{\mathcal{G}}$. This expands the RoPE-commuting, weight-foldable transformation space from independent pair-wise rotations to cross-pair group-wise rotations. We refer to this expansion as *Isofrequency Commutant Expansion* (ICE). Applying such an orthogonal group-wise rotation consistently to the corresponding Q and K groups preserves their unquantized attention score under the frequency-tied RoPE.

Frequency tying is not a lossless reparameterization: for a relative query–key distance $\Delta = t_q - t_k$, replacing $\omega_p$ with $\nu_{\mathcal{G}}$ introduces a phase perturbation that grows with $\Delta|\omega_p - \nu_{\mathcal{G}}|$. ICE therefore trades this positional distortion against the quantization benefit of a larger rotation space.

Table 1 compares the two settings using the same group configuration and group-wise rotation, differing only in whether the original RoPE frequencies are retained or tied within each group. Frequency tying reduces the FP16 attention-output MSE by 96.97% and the quantized MSE by 24.66% relative to the native-frequency setting. This shows that frequency tying enables the cross-pair rotation to commute exactly with the modified RoPE, while substantially reducing the reconstruction error relative to the native-frequency configuration. A context-length analysis is provided in Appendix C.1.

## 3 ROKI

### 3.1 Overview

Figure 3 summarizes the ROKI workflow. ROKI first constructs isofrequency groups by selectively tying the frequencies of low-frequency RoPE pairs. It then optimizes group-wise rotations within the resulting ICE groups and pair-wise rotations for the remaining pairs. After this distribution shaping, ROKI measures the importance of each transformed pair using its QK inner-product contribution and applies a matched pair permutation to Q and K, placing more important key pairs in higher-precision quantization blocks. Finally, reciprocal scaling is optimized for the Q/K pairs associated with the resulting key quantization blocks.

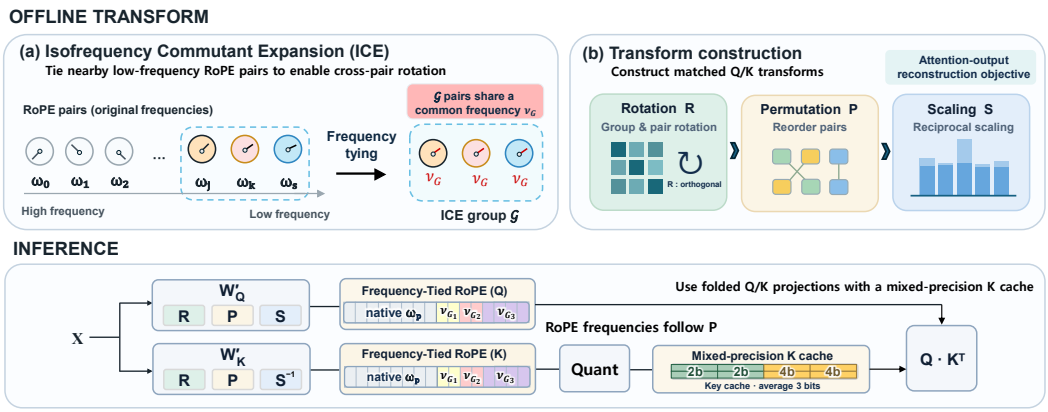



Figure 3: Overview of ROKI. Offline, ROKI constructs isofrequency groups and matched Q/K transformations for distribution shaping and mixed-precision key quantization. The transformations are folded into the projection weights for inference.

The rotation and pair permutation are applied consistently to the corresponding Q and K representations, while reciprocal scaling uses inverse factors for Q and K. These matched Q/K transformations reshape the representation for key quantization while preserving the unquantized QK interaction under the frequency-tied RoPE. Only K is quantized using the mixed-precision scheme, whereas V is quantized independently using uniform per-token quantization.

All group construction and transformations are determined offline. The rotation, pair permutation, and reciprocal scaling are folded into the Q/K projection weights, with the RoPE frequencies reordered consistently with the pair permutation. Inference therefore requires no additional online Q/K transformation.

### 3.2 Isofrequency Commutant Expansion

**Expanding the foldable rotation space.** ICE assigns a common RoPE frequency to selected pairs, creating an isofrequency group in which cross-pair rotations can commute with RoPE. For a group $G$, let $T_G$ denote an orthogonal group-wise rotation that is compatible with the frequency-tied RoPE:

$$\widetilde{D}_t^G T_G = T_G \widetilde{D}_t^G, \qquad \forall t.$$

ROKI applies the same rotation to the corresponding query and key vectors:

$$\widetilde{q}_G = T_G q_G, \qquad \widetilde{k}_G = T_G k_G. \tag{3}$$

Because $T_G$ is orthogonal and commutes with the frequency-tied RoPE, applying it consistently to Q and K preserves the unquantized attention score under the modified RoPE while redistributing activation magnitude across pairs within the group. A formal construction of such RoPE-compatible group-wise rotations is provided in Appendix D.

**Outlier-guided low-frequency group search.** ROKI determines the ICE groups and their shared frequencies independently for each layer and KV head. We restrict the candidate set $\mathcal{P}_{\mathrm{ICE}}$ to the low-frequency half of the RoPE pairs, where large Q/K activations tend to occur and neighboring frequencies are relatively close. For each candidate pair $p$, we use its average key magnitude as an outlier proxy:

$$a_p = \mathbb{E}\left[\|k_p\|_2\right], \tag{4}$$

where the expectation is taken over the calibration data. Pairs are considered as group seeds in descending order of $a_p$, prioritizing pairs with a stronger need for cross-pair redistribution.

For a seed pair $s$, ROKI uses its original frequency as the shared group frequency, $\nu_G = \omega_s$. This preserves the frequency of the dominant seed pair while avoiding an additional search over the shared frequency. Starting from $G = \{s\}$, the remaining candidates are considered in increasing

order of $|\omega_p - \omega_s|$, so that the group is expanded first with pairs requiring relatively small frequency perturbations. Alternative choices of the shared frequency are evaluated in Appendix C.2.

Because the quantization benefit of a candidate group depends on how activation magnitude is redistributed across its pairs, evaluating each candidate requires a group-wise rotation. Jointly optimizing a rotation for every candidate, however, would make the search prohibitively expensive. ROKI therefore uses a random RoPE-compatible group-wise rotation as a lightweight proxy for group utility during search. This random rotation is used only for candidate evaluation; once the group structure and shared frequencies are fixed, the actual rotations are optimized in Section 3.3.

For a candidate configuration $C$, ROKI evaluates the attention-output reconstruction error:

$$\mathcal{L}_{\text{out}}(C) = \text{MSE}\left(O_{\text{Quant}}(C), O_{\text{FP}}^{\text{native}}\right), \tag{5}$$

where $O_{\text{FP}}^{\text{native}}$ is the FP attention output using the original RoPE frequencies. Because $O_{\text{Quant}}(C)$ includes both frequency tying and low-bit key quantization, this objective jointly captures the positional distortion introduced by frequency tying and the quantization error.

For each seed, ROKI progressively expands the group and selects the lowest-loss candidate only if it improves the current configuration. Once a group is accepted, its pairs are removed from subsequent searches. As shown in Appendix C.6, this greedy search empirically avoids the failure mode where the first seed absorbs all low-frequency candidate pairs. Across the evaluated models, it forms 6.13–8.15 groups per head on average, with mean group sizes of 3.32–4.67 pairs. Multiple ICE groups of different sizes coexist within a single KV head, and some groups are non-contiguous in pair index. These results indicate that the greedy search selectively groups pairs according to the reconstruction loss rather than merging the entire candidate region into a single group. The complete search procedure is provided in Algorithm 1.

### 3.3 Distribution-Shaping Rotation Optimization

After the ICE group search, the selected groups and shared frequencies are fixed, and the random rotations used during group search are discarded. ROKI then optimizes the actual distribution-shaping rotations used for low-bit key quantization. Unlike the position-dependent RoPE rotations, these learned rotations are position-independent and are optimized solely to reshape the Q/K activation distribution for quantization.

For a pair $p$ that does not belong to an ICE group, ROKI applies the same pair-wise rotation $R(\phi_p)$ to Q and K:

$$q'_p = R(\phi_p) q_p, \qquad k'_p = R(\phi_p) k_p. \tag{6}$$

Since $R(\phi_p)$ commutes with the positional RoPE rotation $R(t\omega_p)$, it redistributes magnitude within the pair while retaining the original RoPE frequency $\omega_p$.

For an ICE group $G$, ROKI instead applies a group-wise orthogonal rotation $T_G$ that mixes multiple RoPE pairs. As defined in Eq. (3), all pairs in $G$ share the frequency $\nu_G$, allowing $T_G$ to commute with the frequency-tied RoPE operator $\widetilde{D}_t^G$. Thus, $T_G$ redistributes magnitude across pairs while preserving the unquantized QK interaction under the frequency-tied RoPE. The parameterization and commutation properties of the pair-wise and group-wise rotations are detailed in Appendix D.

With the group structure and RoPE frequencies fixed, the rotation parameters are optimized using the attention-output reconstruction objective in Eq. (5), with the corresponding RoPE operation and low-bit key quantization applied during optimization. For grouped-query attention (GQA), the rotation and frequency configuration is defined per KV head and shared by all query heads associated with that KV head.

### 3.4 Transformation-Aware Mixed-Precision Quantization

After ICE-based distribution shaping, ROKI allocates precision according to the transformed representation. Because group-wise rotation can change each pair's contribution, we measure importance using the post-transformation QK inner-product contribution.

Table 8 shows that higher precision on later RoPE pairs improves PPL under the same average 3-bit budget. Figure 7 further shows that pair importance varies across layers and heads. ROKI therefore allocates precision using post-transformation QK contribution rather than fixed pair indices.

**Post-transformation importance.** Group-wise rotation preserves the group-level QK inner product but can change each pair's contribution. We therefore compute importance from the FP Q/K activations after rotation. For pair $p$, we define $I_p = \mathbb{E}_{(t,j)}[(\hat{q}_{t,p}^{\top}\hat{k}_{j,p})^2]$. Importance is computed per layer and KV head. A comparison with Block-GTQ's Q/K-energy metric is provided in Appendix B.3.

**Importance-aware pair reordering.** ROKI assigns high-importance pairs to 4-bit blocks and low-importance pairs to 2-bit blocks. Q/K pairs and their RoPE frequencies are permuted consistently to preserve the pre-quantization attention score under a fixed mixed-precision layout.

**Final block-wise scaling.** After permutation, pair-wise scaling is optimized for the final block layout. Reciprocal scaling is applied to Q and K to preserve their inner product, with parameters optimized using the attention-output reconstruction loss in Eq. (5).

### 3.5 Offline Weight Folding and Inference

The learned rotation, permutation, and scaling are fixed linear transformations and can be folded into the Q/K projection weights, with the RoPE frequencies reordered consistently with the pair permutation. ROKI therefore requires no additional online Q/K transformation during inference. The complete folding formulation is provided in Appendix F.3.

## 4 Experiments

### 4.1 Experimental Setup

**Experimental Setup.** We evaluate ROKI on Llama-3.1-8B-Instruct (Llama Team, AI @ Meta, 2024), Qwen3-8B, and Qwen3-14B (Yang et al., 2025) using NVIDIA GeForce RTX 3090 GPUs. We report LongBench-E (Bai et al., 2024) results over 13 tasks and additionally evaluate GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), HumanEval (Chen et al., 2021), and GPQA (Rein et al., 2024). Baselines are QuaRot, QServe, FPTQuant, KIVI, and Block-GTQ, with FP16 as the reference. All quantized methods use W16A16 and an average 3-bit KV cache. ROKI uses per-token 2/4-bit mixed-precision quantization for K and uniform 3-bit quantization for V. The quantization group size is 32 except for Block-GTQ, which uses layer- and head-dependent block sizes. ROKI and KIVI retain the most recent 128 KV tokens in FP16.

### 4.2 Long-Context Evaluation

**LongBench-E.** Table 2 shows that ROKI achieves the highest average LongBench-E score among the evaluated quantized methods on all three models while remaining close to FP16 under an average 3-bit KV-cache budget.

**Needle-in-a-Haystack.** To evaluate long-context retrieval with frequency tying, we conduct Needle-in-a-Haystack experiments across different context lengths and target insertion positions. As shown in Figure 10, ROKI achieves an average accuracy of 95.0% over 2K–80K contexts, outperforming KIVI (94.2%) and Block-GTQ (94.6%). These results show that ROKI largely preserves retrieval performance even with frequency tying.

**Context-Length Sensitivity.** Frequency tying introduces larger positional distortion as the context length increases. As shown in Table 3, the FP16 reconstruction error of ICE on Qwen3-8B increases with context length; nevertheless, ICE consistently achieves lower quantized attention-output MSE than the pair-wise configuration, including a 10.59% reduction at 128K. Full results, including Llama-3.1-8B-Instruct, are provided in Appendix C.1.

### 4.3 Reasoning and Code Generation

Table 4 shows that ROKI achieves the highest average score among quantized methods, with average-score gaps of only 0.82, 0.94, and 0.06 points from FP16, respectively. Here, MATH and GPQA denote MATH-Algebra and GPQA-Diamond, respectively. ROKI also achieves the best or tied-best quantized performance on 11 of the 12 task–model combinations.

Table 2: LongBench-E results on Llama-3.1-8B-Instruct, Qwen3-8B, and Qwen3-14B. Higher scores are better. Bold and underline denote the best and second-best results among quantized methods, respectively.

| Model | Method | Single-doc QA | Multi-doc QA | Summarization | Few-shot | Synthetic | Code | Avg. |
|---|---|---|---|---|---|---|---|---|
| Llama-3.1 8B-Instruct | FP16 | 49.78 | 54.72 | 30.13 | 68.24 | 58.92 | 53.50 | 53.75 |
| | QuaRot | 47.66 | 46.51 | 29.39 | 67.07 | 52.68 | 52.58 | 50.68 |
| | QServe | 47.86 | 47.68 | 29.91 | 65.69 | 51.64 | 52.63 | 50.50 |
| | FPTQuant | 48.10 | 42.19 | 29.52 | 65.48 | 52.30 | 50.94 | 48.09 |
| | KIVI | 49.80 | 52.75 | **30.19** | **68.57** | 55.89 | 52.74 | 52.96 |
| | Block-GTQ | 48.46 | 51.88 | 29.94 | 68.31 | 58.58 | **53.08** | 52.98 |
| | ROKI | **50.74** | **54.25** | 29.97 | 68.39 | **58.68** | 53.05 | **53.73** |
| Qwen3-8B | FP16 | 47.82 | 54.20 | 28.40 | 67.03 | 55.70 | 59.77 | 53.30 |
| | QuaRot | 44.94 | 47.87 | 28.26 | 67.29 | 54.17 | 56.58 | 51.19 |
| | QServe | 42.89 | 50.68 | 28.13 | 66.78 | 52.28 | 58.23 | 51.13 |
| | FPTQuant | 45.09 | 48.94 | 28.24 | 67.05 | 53.58 | 54.58 | 50.92 |
| | KIVI | 46.48 | 54.08 | 28.41 | 67.50 | 54.92 | 59.09 | 52.96 |
| | Block-GTQ | 45.96 | 54.12 | **28.73** | **67.90** | 55.06 | 57.08 | 52.74 |
| | ROKI | **47.18** | **54.19** | 28.46 | 67.73 | **55.44** | **59.44** | **53.28** |
| Qwen3-14B | FP16 | 47.68 | 61.30 | 28.19 | 68.41 | 61.52 | 57.91 | 55.26 |
| | QuaRot | 44.29 | 51.51 | 28.19 | 65.37 | 53.33 | 47.25 | 49.63 |
| | QServe | 45.39 | 56.57 | 28.02 | 68.27 | 54.40 | 55.15 | 52.61 |
| | FPTQuant | 46.10 | 57.18 | **28.33** | 67.77 | 54.00 | 54.76 | 52.62 |
| | KIVI | **48.13** | 58.71 | 28.06 | 68.31 | 59.11 | **56.71** | 54.34 |
| | Block-GTQ | 47.39 | 59.39 | 28.06 | 68.00 | **60.68** | 56.34 | 54.44 |
| | ROKI | 47.94 | **60.44** | 28.16 | **68.61** | 60.17 | 55.64 | **54.66** |

Table 3: Context-length sensitivity on Qwen3-8B. Attention-output MSE is measured relative to Native FP16. Gain denotes the reduction in quantized attention-output MSE over Pair-wise.

| Context | ICE FP16 MSE | Pair-wise Quant. | ICE Quant. | Gain |
|---|---|---|---|---|
| 2K | $3.505 \times 10^{-6}$ | $4.0328 \times 10^{-2}$ | $3.1674 \times 10^{-2}$ | 21.46% |
| 32K | $2.589 \times 10^{-4}$ | $3.5878 \times 10^{-2}$ | $2.9377 \times 10^{-2}$ | 18.12% |
| 128K | $5.005 \times 10^{-3}$ | $3.3120 \times 10^{-2}$ | $2.9612 \times 10^{-2}$ | 10.59% |

Table 4: Reasoning and code-generation results. Scores are reported in the order of Llama-3.1-8B-Instruct / Qwen3-8B / Qwen3-14B. Bold denotes the best result among quantized methods.

| Method | GSM8K | MATH | HumanEval | GPQA | Avg. |
|---|---|---|---|---|---|
| FP16 | 77.79 / 91.59 / 92.49 | 47.01 / 93.68 / 94.95 | 68.90 / 85.98 / 89.63 | 33.67 / 45.41 / 48.47 | 56.84 / 79.16 / 81.39 |
| KIVI | **77.18** / 90.60 / 91.36 | 46.67 / 92.59 / 94.69 | 60.98 / 84.76 / 86.59 | 26.53 / 38.27 / 45.92 | 52.84 / 76.55 / 79.64 |
| Block-GTQ | 72.86 / 89.16 / 91.81 | 42.97 / 91.58 / 95.11 | 60.37 / **87.81** / 87.81 | 25.00 / 36.74 / 46.43 | 50.30 / 76.32 / 80.29 |
| ROKI | 77.03 / **90.83** / **92.19** | **47.85** / **94.95** / **96.13** | **67.07** / **87.81** / **89.02** | **32.14** / **39.29** / **47.96** | **56.02** / **78.22** / **81.33** |

### 4.4 Ablation Study

Across the evaluated models, ICE groups contain 3.32–4.67 pairs on average, with maximum sizes of 27–30 pairs. Representative group assignments are provided in Appendix C.6.

Table 5 evaluates ICE and importance-aware permutation on Qwen3-8B under the same K2/4 mixed-precision setting. The same rotation is used with and without permutation, while scaling is optimized for each final pair layout. Under this setting, ICE reduces attention-output MSE by 18.12% over the pair-wise configuration, and adding permutation further improves the MSE to 0.021522 and the PPL to 7.31.

Table 5: Qwen3-8B component ablation. P denotes permutation.

| Configuration | Attn.MSE ↓ | PPL ↓ |
|---|---|---|
| FP16 | — | 7.27 |
| Pair-wise | 0.035878 | 7.41 |
| ICE | 0.029377 | 7.36 |
| ICE + P | **0.021522** | **7.31** |

### 4.5 End-to-End Decode Latency

Figure 4 reports end-to-end decode latency for Llama-3.1-8B-Instruct on an RTX 3090 with batch size 1 across 4K–128K contexts. Each run generates 128 tokens, excluding prefill. ROKI remains close to FP16 at short contexts and reduces latency by 2.5% and 6.5% at 32K and 64K, respectively.

At 64K, ROKI is $3.24\times$ and $3.57\times$ faster than KIVI and Block-GTQ. At 128K, FP16 runs out of memory, while ROKI remains executable at 40.52 ms/token, achieving $4.12\times$ and $4.22\times$ speedups over KIVI and Block-GTQ, respectively. ROKI requires no online Q/K transformation and retains a regular per-token KV-cache layout. Details of the latency baselines and kernel implementation are provided in Appendix F.4.

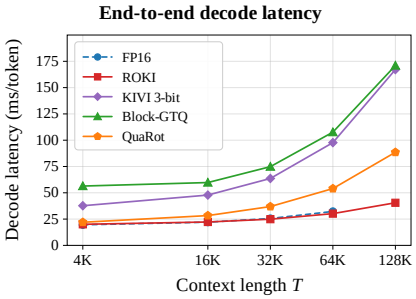



Figure 4: End-to-end decode latency.

## 5 Related Work

**Low-bit KV-cache quantization.** KIVI (Liu et al., 2024) uses per-channel quantization for keys and per-token quantization for values to accommodate their different activation distributions. KVQuant (Hooper et al., 2024) quantizes keys before RoPE to preserve channel-wise structure. ROKI targets post-RoPE per-token quantization, reshaping activations to reduce the impact of outliers on shared quantization ranges.

**RoPE-aware distribution shaping.** QuaRot (Ashkboos et al., 2024) and SpinQuant (Liu et al., 2025) redistribute outliers using online post-RoPE Q/K rotations. TurboQuant (Zandieh et al., 2026) combines runtime rotation with codebook-based quantization. QServe (Lin et al., 2025) and FP-TQuant (van Breugel et al., 2026) use RoPE-compatible pair-wise transformations that fold into projection weights. ROKI extends this foldable transformation space to multi-pair groups through frequency tying, enabling cross-pair redistribution without additional online Q/K transformation.

**Importance-aware mixed-precision quantization.** Block-GTQ (Liang et al., 2026) allocates bits using pair-wise Q/K energy, while Kitty (Xia et al., 2026) dynamically selects precision based on key-channel magnitudes. ROKI instead permutes pairs into a fixed mixed-precision layout using their post-transformation QK inner-product contributions, aligning high-precision placement with the distribution shaped by ICE. Unlike metrics computed from the original activations, this measure reflects the representation after distribution shaping.

## 6 Conclusion

We propose ROKI, a KV-cache quantization method that improves low-bit accuracy while preserving the efficiency of per-token quantization. Its core technique, Isofrequency Commutant Expansion (ICE), assigns shared frequencies to selected low-frequency RoPE pairs, expanding the space of rotations that can be folded into the Q/K projection weights from individual pairs to groups. ROKI combines this expanded rotation space with importance-aware pair permutation and block-wise scaling to redistribute large values and reduce quantization error, while all learned transformations are folded into the weights offline with no additional inference-time transformation. Under an average 3-bit KV-cache setting, ROKI achieves the highest average LongBench-E score among the evaluated quantization methods. On Llama-3.1-8B-Instruct, Qwen3-8B, and Qwen3-14B, ROKI remains close to FP16 in accuracy. Our ROKI implementation achieves up to $4.12\times$ and $4.22\times$ end-to-end decode speedups over KIVI and Block-GTQ, respectively. These results show that carefully modifying the RoPE frequency structure can expose richer foldable transformations, improving low-bit KV quantization without adding online Q/K transforms.

### AI USE STATEMENT

In this work, we used generative AI tools to assist with writing and editing, literature search and related-work discovery, brainstorming research ideas and experimental methodology, drafting parts of the paper, and reviewing mathematical claims and derivations. References suggested by AI tools were independently checked by the authors against the original sources. Research ideas and experimental methods were reviewed by the authors and validated through actual experiments. Mathematical claims and derivations assisted by AI were also independently verified by the authors. All AI-assisted or AI-generated content was reviewed and revised by the authors before inclusion in the final manuscript. The authors take full responsibility for all claims, results, and content presented in this work.

### REPRODUCIBILITY STATEMENT

Implementation, optimization, and evaluation details are provided in Appendix F. The ICE group search procedure is described in Algorithm 1.

## REFERENCES

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. Quarot: Outlier-free 4-bit inference in rotated LLMs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, NeurIPS'24, 2024. URL https://openreview.net/forum?id=dfqsW38v1X.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longbench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3119–3137. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.172. URL https://aclanthology.org/2024.acl-long.172/.

Federico Barbero, Alex Vitvitskyi, Christos Perivolaropoulos, Razvan Pascanu, and Petar Veličković. Round and round we go! what makes rotary positional encodings useful? In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=GtvuNrk58a.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. URL https://arxiv.org/abs/2107.03374.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 1, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Coleman Richard Charles Hooper, Sehoon Kim, Hiva Mohammadzadeh, Michael W. Mahoney, Sophia Shao, Kurt Keutzer, and Amir Gholami. KVQuant: Towards 10 million context length

LLM inference with KV cache quantization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, NeurIPS'24, 2024. URL https://openreview.net/forum?id=0LXotew9Du.

Xing Li, Zeyu Xing, Yiming Li, Linping Qu, Hui-Ling Zhen, Yiwu Yao, Wulong Liu, Sinno Jialin Pan, and Mingxuan Yuan. Kvtuner: Sensitivity-aware layer-wise mixed-precision kv cache quantization for efficient and nearly lossless llm inference. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 36451–36485, 2025. URL https://proceedings.mlr.press/v267/li25dd.html.

Fengfeng Liang, Yuechen Zhang, and Jiaya Jia. Rope-aware bit allocation for kv-cache quantization, 2026. URL https://arxiv.org/abs/2606.24033.

Yujun Lin, Haotian Tang, Shang Yang, Zhekai Zhang, Guangxuan Xiao, Chuang Gan, and Song Han. Qserve:w4a8kv4 quantization and system co-design for efficient llm serving. In *Proceedings of Machine Learning and Systems*, MLSys'25, 2025. URL https://proceedings.mlsys.org/paper_files/paper/2025/file/fbe2b2f74a2ece8070d8fb073717bda6-Paper-Conference.pdf.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. Spinquant: LLM quantization with learned rotations. In *The Thirteenth International Conference on Learning Representations*, ICLR'25, 2025. URL https://openreview.net/forum?id=ogO6DGE6FZ.

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 *Forty-first International Conference on Machine Learning*, ICML'24, 2024. URL https://openreview.net/forum?id=L057s2Rq8O.

Llama Team, AI @ Meta. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1609.07843.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. doi: 10.1016/j.neucom.2023.127063. URL https://doi.org/10.1016/j.neucom.2023.127063.

Boris van Breugel, Yelysei Bondarenko, Paul N. Whatmough, and Markus Nagel. FPTQuant: Function-preserving transforms for LLM quantization. In *Forty-third International Conference on Machine Learning*, ICML'26, 2026. URL https://openreview.net/forum?id=2jpCo5lnSZ.

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, and Shuaiwen Song. Kitty: Accurate and efficient 2-bit kv cache quantization with dynamic channel-wise precision boost. In *Proceedings of Machine Learning and Systems*, 2026. URL https://proceedings.mlsys.org/paper_files/paper/2026/hash/e4d8d1b5120be349d3fff8878650cf45-Abstract-Conference.html.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang,

Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Amir Zandieh, Majid Daliri, Majid Hadian, and Vahab Mirrokni. Turboquant: Online vector quantization with near-optimal distortion rate. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/5c802ef38ab6e366c2ea06eee554c088-Abstract-Conference.html.

# APPENDIX

## A ADDITIONAL MOTIVATION AND ROPE-PAIR ANALYSIS

### A.1 KEY ACTIVATION DISTRIBUTIONS

Figure 5 compares the distribution of K activations before and after applying ROKI. In the original activations, large values are concentrated in a few channels across multiple tokens, potentially dominating the quantization range of each block. After applying ROKI, these large values are redistributed across channels, reducing the magnitude imbalance among channels. This helps mitigate the impact of channel-wise outliers on the representation precision of the remaining elements in per-token quantization, where a scale is shared within each block.

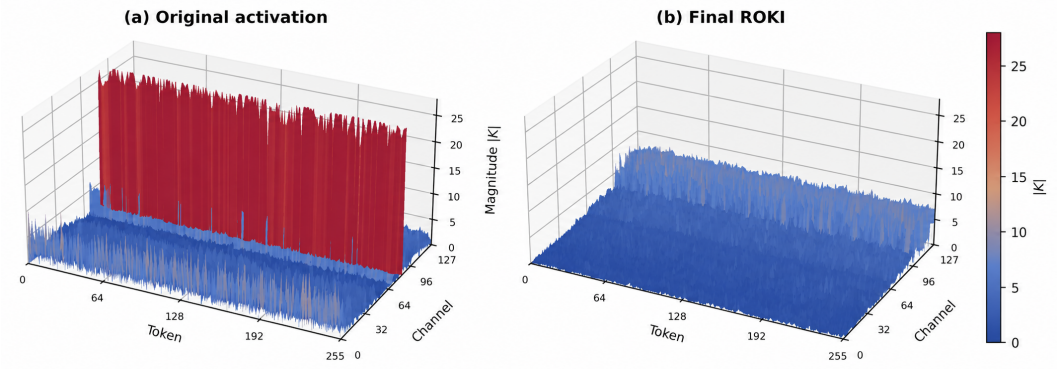



Figure 5: The absolute-value distribution of K activations before and after applying ROKI at layer 29, head 6 of Llama-3.1-8B-Instruct. (a) In the original distribution, large values are concentrated in specific channels, which can limit the representation precision of the remaining elements when multiple elements share a scale. (b) ROKI reduces this concentration of large values, producing a distribution that is more favorable for shared-scale quantization.

### A.2 ROTATION-SCOPE ANALYSIS

Table 6: Effect of rotation scope on K/Q activation peaks and QK-MSE for RoPE pairs 32–63 of Llama-3.1-8B-Instruct, averaged over all 32 layers. All configurations use reciprocal scaling; parentheses show QK-MSE reduction relative to scaling only.

| Configuration | K peak ↓ | Q peak ↓ | QK-MSE ↓ |
|---|---|---|---|
| Scaling-only | 12.36 | 9.83 | 0.12078 |
| Pair-wise rotation + scaling | 10.58 | 8.65 | 0.08526 (↓ 29.41%) |
| Group-wise rotation + scaling | **8.82** | **7.59** | **0.06187** (↓ 48.77%) |

### A.3 PAIR-WISE STATISTICS AND IMPORTANCE-BASED REORDERING

We collect post-RoPE Q/K activations from the original FP model of Llama-3.1-8B-Instruct and analyze the magnitude and QK inner-product contribution of each pair. Statistics are aggregated over sampled causal query–key position pairs and the corresponding GQA heads at each layer. Q/K magnitude is measured by the RMS norm of each pair vector, while QK contribution is defined as $S_p = \sqrt{\mathbb{E}[(q_p^\top k_p)^2]}$.

The top row of Figure 6 shows a tendency for later pairs to exhibit larger Q/K activations and higher QK contributions, although the distributions vary across layers and pairs. In the bottom row, pairs are first divided into the upper and lower halves according to QK contribution and then reordered within each region according to the absolute K value. After reordering, pairs with high QK contribution

are concentrated in the later region, forming an ordering suitable for assignment to fixed high-bit blocks. This analysis applies only the permutation to the original activations and therefore does not represent the statistics after the final ROKI ICE transformation.

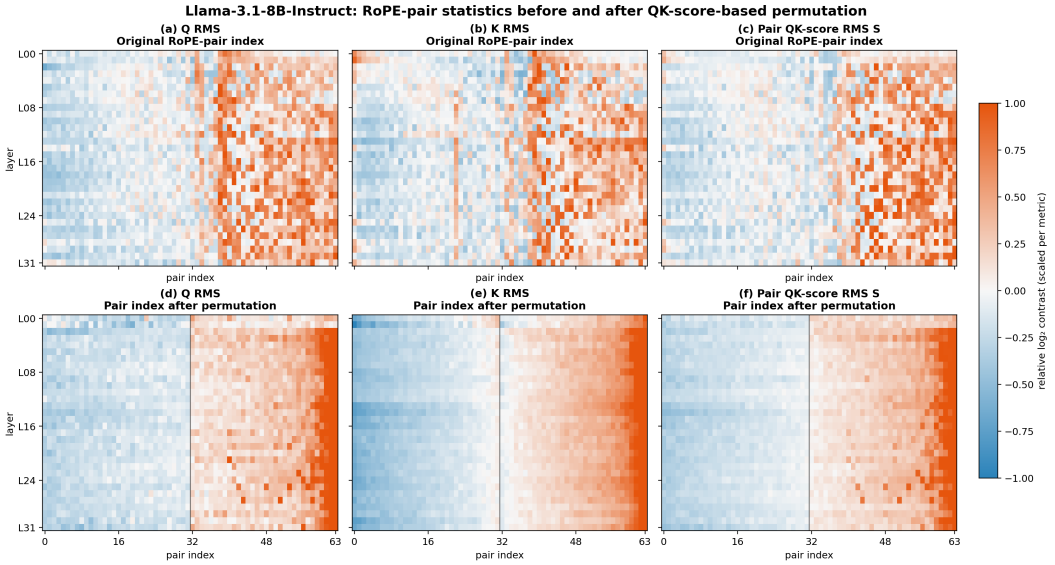



Figure 6: Pair statistics before and after reordering for Llama-3.1-8B-Instruct. The top row shows the original pair order, while the bottom row shows the order after QK-contribution-based reordering. The columns show Q RMS, K RMS, and the pair-wise QK inner-product contribution $S_p$, respectively. Colors indicate the $\log_2$ ratio relative to the layer-wise median in each panel, with the same color range used for the before/after panels in each column. The vertical line in the bottom row marks the boundary between the regions assigned to 2-bit and 4-bit precision. Only permutation is applied to the same original FP activations.

Figure 7 compares Q/K magnitudes and pair-wise QK inner-product contributions before and after permutation using the original FP activations of Llama-3.1-8B-Instruct. Although later pairs tend to exhibit higher QK contributions, the locations and distributions of important pairs vary across heads. In particular, some heads also contain highly important pairs at earlier indices, making the original pair index alone insufficient for precision allocation.

Reordering pairs according to their QK inner-product contributions moves important pairs in each head toward the later high-bit region. This enables higher precision to be assigned according to the head-specific importance distribution while retaining a fixed mixed-precision layout. ROKI applies this principle to the transformed activations: QK inner-product contributions are computed after ICE and pair-wise rotation, and the pairs are then reordered accordingly.

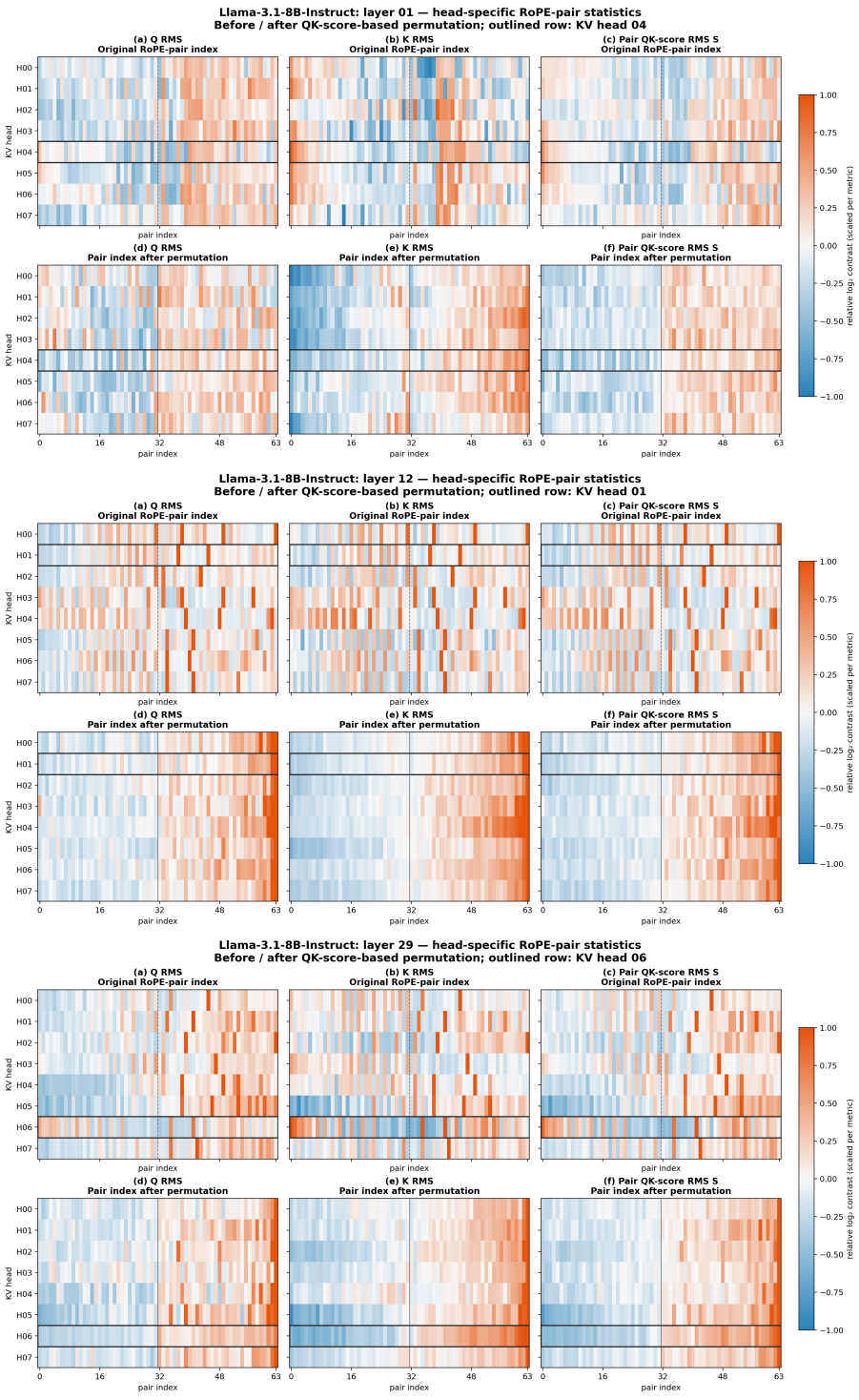



Figure 7: Distribution of RoPE-pair importance across layers and heads of Llama-3.1-8B-Instruct, before and after permutation. From top to bottom, the rows correspond to Layers 1, 12, and 29, with black boxes highlighting KV heads 4, 1, and 6. For each case, the upper and lower rows show the original and permuted pair orders, while the columns show Q magnitude, K magnitude, and QK inner-product contribution. Although later pairs tend to have higher contributions, important-pair locations vary across layers and heads. Importance-aware permutation captures these differences by moving high-QK-contribution pairs to the later high-bit region. All statistics are measured from the original FP activations before ICE.

### A.4 RoPE Frequency Characteristics

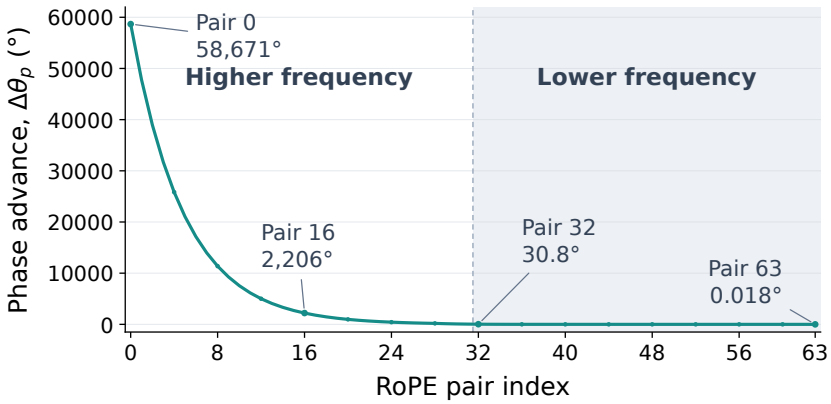



Figure 8: Relative RoPE rotation angle $\Delta\theta_p = \Delta\omega_p$ at token distance $\Delta = 1{,}024$ for Llama-3.1-8B-Instruct. The shaded region denotes the low-frequency pairs (32–63).

## B Additional Optimization and Quantization Analysis

### B.1 Comparison of Optimization Objectives

We compare QK-MSE, attention-output MSE, and the end-to-end JSD-based training objective used by FPTQuant. QK-MSE minimizes the difference in QK inner products before and after quantization, while attention-output MSE minimizes the reconstruction error of the attention output. FPTQuant's end-to-end training instead minimizes the Jensen–Shannon divergence (JSD) between the final output distributions of the FP and quantized models, and does not directly minimize the output error of each attention layer.

Table 7: Comparison of attention-output MSE, QK-MSE, and LongBench scores across different training objectives. The best results are shown in bold.

| Training Objective | Attention-output MSE ↓ | QK-MSE ↓ | LongBench ↑ |
|---|---|---|---|
| QK-MSE | 0.029122 | 0.412828 | 50.88 |
| Attention-output MSE | **0.028703** | **0.319822** | **51.41** |
| FPTQuant E2E-JSD | 0.042773 | 0.471790 | 50.92 |

In Table 7, training with attention-output MSE achieves the lowest values for both reconstruction errors. Its LongBench score is 51.41, outperforming QK-MSE-based training and FPTQuant E2E-JSD by 0.53 and 0.49 points, respectively.

Differences in QK logits can have different effects on attention weights after the softmax operation, making QK-MSE insufficient to fully capture changes in the resulting attention output. Attention-output MSE instead directly minimizes the discrepancy in the attention output, accounting for these effects. This likely contributes to the improved performance observed in our evaluation setting. We therefore use attention-output MSE as the training objective for ROKI.

### B.2 Bit-Width Placement Sensitivity

Table 8: WikiText-2 perplexity under different RoPE-pair bit-width allocations for Llama-3.1-8B-Instruct. Without importance-aware permutation, different precisions are assigned to pairs 0–31 and 32–63 while maintaining an average 3-bit KV-cache budget.

| Earlier original pairs (0–31) | Later original pairs (32–63) | PPL ↓ |
|---|---|---|
| 1-bit | 5-bit | 38.5435 |
| **2-bit** | **4-bit** | **8.1037** |
| 3-bit | 3-bit | 10.6502 |
| 4-bit | 2-bit | 29.6364 |
| 5-bit | 1-bit | 273.1700 |

### B.3 QK-Contribution-Based Precision Allocation

To examine the rationale for assigning higher precision to blocks with larger QK contributions, we analyze the block-wise K distribution and quantization error after the final transformation. As shown in Table 9, blocks with larger QK contributions also exhibit higher K variance and range, and applying 4-bit instead of 2-bit quantization yields larger absolute reductions in both K-MSE and QK-MSE. In particular, after permutation, 97.47% of the QK contribution is concentrated in Blocks 2 and 3, where the error reduction from higher precision is also substantially larger. These results support assigning 4-bit precision to blocks with high QK contribution and 2-bit precision to less important blocks under a fixed bit budget.

Table 9: Block-wise K distribution and QK contribution after rotation, permutation, and scaling on Llama-3.1-8B-Instruct. K variance and range are the averages of the variance and maximum–minimum difference, respectively, computed within each 32-dimensional block for each token and head. QK contribution denotes the block-wise fraction of the pair-wise squared inner-product metric. ΔMSE is the absolute error reduction obtained by using 4-bit instead of 2-bit quantization ($\mathrm{MSE}_{2\mathrm{bit}} - \mathrm{MSE}_{4\mathrm{bit}}$). Bold rows indicate blocks assigned 4-bit precision.

| Block | K variance | K range | QK contribution | ΔK-MSE ↑ | ΔQK-MSE ↑ |
|---|---|---|---|---|---|
| 0 (pair 0–15) | 1.11199 | 4.19808 | 0.7504% | 0.132634 | 0.015892 |
| 1 (pair 16–31) | 2.14659 | 5.84181 | 1.7754% | 0.251964 | 0.038472 |
| **2 (pair 32–47)** | **4.37967** | **7.88122** | **11.9284%** | **0.466894** | **0.246789** |
| **3 (pair 48–63)** | **10.12974** | **11.90659** | **85.5458%** | **1.051591** | **1.565235** |

**Case Study on the Pair Importance Metric.** We analyze how the choice of importance metric affects high-bit pair selection and attention-output reconstruction error on Llama-3.1-8B-Instruct. Under the same 2/4-bit mixed-precision layout, pairs are reordered using either the norm-based metric of Block-GTQ or the QK-contribution metric of ROKI. Block-GTQ computes importance from the sum of the squared norms of Q and K, whereas ROKI uses the mean squared pair-wise QK inner product, thereby capturing not only vector magnitude but also the relative orientation between Q and K.

Across all 256 KV heads in the 32 layers, the QK-contribution-based allocation of ROKI reduces attention-output MSE by an average of 1.124% relative to the norm-based allocation of Block-GTQ under the same sequence-length setting. To examine cases with larger differences, we identify the heads with the largest disagreement in 4-bit pair selection. The largest differences occur at Layer 0 / KV head 6, Layer 29 / KV head 4, and Layer 26 / KV head 2.

When only the target KV head is quantized and all other heads are kept in FP, ROKI reduces attention-output MSE by 29.34% at Layer 0 / KV head 6 ($1.96080 \times 10^{-7} \rightarrow 1.38548 \times 10^{-7}$), 29.74% at Layer 29 / KV head 4 ($2.17609 \times 10^{-5} \rightarrow 1.52900 \times 10^{-5}$), and 10.41% at Layer 26 / KV head 2 relative to Block-GTQ. These results show that QK-contribution-aware high-bit allocation reduces reconstruction error overall and can provide substantially larger gains for heads where the selected important pairs differ from those identified by the norm-based metric.

# C Additional Analysis of ICE

## C.1 Context-Length Sensitivity

To examine how transformation cost and quantization benefit vary with context length, we evaluate Qwen3-8B from 2K to 128K contexts. Both configurations apply rotation and scaling, while permutation is excluded.

We evaluate on WikiText-2 (Merity et al., 2017) using the same layer inputs obtained from the Native FP16 model for each configuration. Reconstruction error is measured by comparing the attention output after the output projection $W_O$ with the Native FP16 output.

Table 10: Attention-output MSE relative to Native FP16 across different context lengths on Qwen3-8B. Both configurations apply rotation and scaling, while ICE additionally includes frequency tying and group-wise rotation. Permutation is excluded. Values in parentheses indicate the MSE reduction relative to quantized Pair-wise.

| Context length | Setting | Pair-wise | ICE |
|---|---|---|---|
| 2K | FP16 | $0$ | $3.505 \times 10^{-6}$ |
| | Quant. | $4.0328 \times 10^{-2}$ | $\mathbf{3.1674 \times 10^{-2}}$ ($\downarrow$ 21.46%) |
| 8K | FP16 | $0$ | $3.648 \times 10^{-5}$ |
| | Quant. | $3.8143 \times 10^{-2}$ | $\mathbf{3.0917 \times 10^{-2}}$ ($\downarrow$ 18.95%) |
| 32K | FP16 | $0$ | $2.589 \times 10^{-4}$ |
| | Quant. | $3.5878 \times 10^{-2}$ | $\mathbf{2.9377 \times 10^{-2}}$ ($\downarrow$ 18.12%) |
| 128K | FP16 | $0$ | $5.005 \times 10^{-3}$ |
| | Quant. | $3.3120 \times 10^{-2}$ | $\mathbf{2.9612 \times 10^{-2}}$ ($\downarrow$ 10.59%) |

Table 11: Attention-output MSE relative to Native FP16 across different context lengths on Llama-3.1-8B-Instruct. Both configurations retain rotation and scaling, while ICE additionally includes frequency tying and group-wise rotation. Permutation is excluded.

| Context length | Setting | Pair-wise | ICE |
|---|---|---|---|
| 2K | FP16 | $9.404 \times 10^{-9}$ | $9.460 \times 10^{-9}$ |
| | Quant. | $2.329 \times 10^{-4}$ | $\mathbf{2.027 \times 10^{-4}}$ ($\downarrow$ 12.95%) |
| 8K | FP16 | $1.277 \times 10^{-8}$ | $5.586 \times 10^{-8}$ |
| | Quant. | $2.555 \times 10^{-4}$ | $\mathbf{2.229 \times 10^{-4}}$ ($\downarrow$ 12.75%) |
| 32K | FP16 | $1.492 \times 10^{-8}$ | $4.624 \times 10^{-7}$ |
| | Quant. | $2.702 \times 10^{-4}$ | $\mathbf{2.379 \times 10^{-4}}$ ($\downarrow$ 11.95%) |
| 128K | FP16 | $1.684 \times 10^{-8}$ | $1.646 \times 10^{-5}$ |
| | Quant. | $2.714 \times 10^{-4}$ | $\mathbf{2.465 \times 10^{-4}}$ ($\downarrow$ 9.17%) |

Table 11 shows the results across different context lengths on Llama-3.1-8B-Instruct. ICE consistently achieves lower attention-output MSE than the pair-wise configuration at all evaluated lengths, even when the error introduced by frequency tying is included.

**Pair concentration and model-dependent gains.** To examine the model-dependent gains of ICE, we compare how much larger the key norm of the largest pair is than the average across all pairs. After pair-wise rotation, the largest-pair key norm is $4.739\times$ the average on Llama-3.1-8B-Instruct and $7.257\times$ on Qwen3-8B, indicating stronger concentration in a small number of pairs for Qwen. After applying ICE, these ratios decrease to $3.318\times$ and $4.296\times$, corresponding to reductions of approximately 30.0% and 40.8%, respectively.

Pair-wise rotation can redistribute values within a pair but cannot change the norm of that pair. In contrast, ICE mixes multiple pairs within a group and can therefore mitigate such concentration. Qwen exhibits stronger initial concentration and a larger reduction after ICE, suggesting that this effect may contribute to the larger quantization-error reduction observed on Qwen.

### C.2 Shared Frequency Selection

**Outlier anchor vs. QK-MSE search vs. arithmetic mean.** When all pairs in a group share the same frequency, a RoPE-compatible group-wise rotation can be constructed, and the shared frequency need not be that of the starting pair. ROKI selects an outlier pair with a large K activation as the starting pair $s$ and uses its frequency $\omega_s$ as the shared frequency of the group. This preserves the original frequency of the outlier pair while avoiding a separate search for the shared frequency.

To examine the effect of shared-frequency selection, we compare the outlier anchor, QK-MSE-based selection, and arithmetic mean. The outlier anchor uses the frequency of the starting pair. QK-MSE-based selection evaluates candidate frequencies on calibration data and selects the shared value that minimizes QK-MSE. The arithmetic mean uses the equally weighted average of the original frequencies of the pairs in the group.

We fix the ICE groups, rotation, permutation, and scaling of Llama-3.1-8B-Instruct and vary only the shared frequency assigned to each group. Attention-output MSE is aggregated over all 32 layers. All errors are measured after applying $W_O$, using the Native FP output with the original frequencies as the reference. We evaluate both the unquantized setting and the setting with mixed-precision quantization applied to K, while keeping V in FP.

Table 12: Comparison of shared-frequency selection methods for a fixed group. FP tying-only MSE is the attention-output error relative to Native FP, measured without quantization. The relative change in post-quantization MSE is reported with respect to QK-MSE-based frequency selection.

| Frequency selection | FP tying-only MSE ↓ | Quant. MSE change (%) |
|---|---|---|
| QK-MSE-based | $1.03 \times 10^{-7}$ | — |
| Outlier anchor | $\mathbf{1.02 \times 10^{-7}}$ | $+0.08$ |
| Arithmetic mean | $6.31 \times 10^{-7}$ | $+0.53$ |

Table 12 shows that the outlier anchor yields frequency-tying error comparable to QK-MSE-based selection, with only a 0.08% difference in MSE after quantization. In contrast, the arithmetic mean produces larger tying error than either method.

The arithmetic mean assigns equal weight to the frequencies of all pairs, even though their contributions to the QK inner product differ. As a result, using the mean frequency can also alter the frequencies of pairs that contribute strongly to QK interactions, potentially causing larger changes in attention. The outlier-based strategy instead preserves the original frequency of a pair with large K values, based on the observation that such a pair may also have a strong influence on the QK inner product. Although the frequencies of the remaining pairs are changed, the overall change in attention can be smaller if their contributions are relatively limited.

These results show that, for a fixed group, simply using the outlier pair's frequency as the shared value can achieve reconstruction performance comparable to a separate QK-MSE-based frequency search. The lower frequency-tying error than the arithmetic mean further supports the use of the outlier anchor without requiring additional frequency search.

### C.3 Low-Frequency Candidate Range

**Sensitivity to the low-frequency candidate range.** To analyze the effect of the low-frequency candidate range used for ICE group search, we compare the lowest-frequency 25% of RoPE pairs (pairs 48–63) with the lowest-frequency 50% (pairs 32–63) on Llama-3.1-8B-Instruct.

When the candidate range is restricted to 25%, the groups that can be formed entirely within pairs 48–63 account for 18.77% of the ICE groups obtained under the 50% setting. Accordingly, the FP16 MSE with frequency tying only decreases from $4.60437 \times 10^{-7}$ to $2.046 \times 10^{-8}$. This reduction

occurs because fewer pair frequencies are modified, resulting in a smaller perturbation to the original RoPE.

However, narrowing the candidate range also limits the extent to which outliers can be redistributed across multiple pairs. The final quantized attention-output MSE is $1.575 \times 10^{-4}$ under the 25% setting and $1.157 \times 10^{-4}$ under the 50% setting, with the 50% setting achieving 26.54% lower error. Thus, although the 25% setting introduces less error from frequency tying itself, its restricted group range does not provide sufficient cross-pair redistribution.

Meanwhile, under the 50% candidate range, all selected groups remain within the low-frequency half. Therefore, the low-frequency half provides a practical candidate range that includes the observed effective ICE groups without unnecessarily modifying higher-frequency pairs.

### C.4 Random-Rotation Sensitivity

**Sensitivity to random rotations.** To examine the effect of the random rotation used during ICE group search, we repeat the group search using three randomly selected rotation seeds. The selected pair sets achieve an average Jaccard similarity of 87.17% across the three seeds, indicating that the pairs selected for ICE remain largely consistent despite changes in the random rotation.

### C.5 Frequency-Tying-Only FP16 Accuracy

Table 13: We evaluate the effect of frequency tying itself on the FP16 models of Llama-3.1-8B-Instruct and Qwen3-8B. Specifically, without quantization, we modify only the RoPE frequencies of the selected ICE groups to a shared value, while disabling rotation, scaling, and permutation, and compare this tying-only FP16 setting against Native FP16. Attention-output MSE is measured after applying $W_O$, using the Native FP16 output as the reference.

| Model | Configuration | Attention-output MSE ↓ | PPL ↓ |
|---|---|---|---|
| Llama-3.1 8B-Instruct | Native FP16 | 0 | 6.23 |
| | FP16 + Frequency tying | $4.60437 \times 10^{-7}$ | 6.24 |
| Qwen3-8B | Native FP16 | 0 | 7.27 |
| | FP16 + Frequency tying | $2.58409 \times 10^{-4}$ | 7.27 |

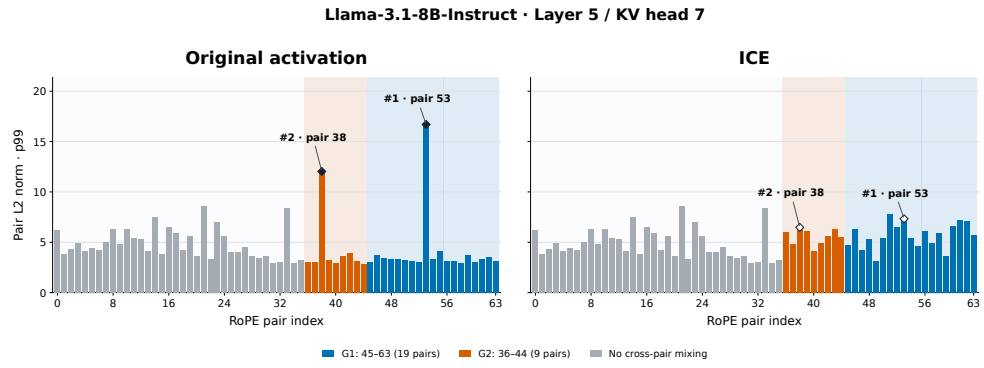
**Effect of Frequency Tying on FP16 Accuracy.** Table 13 shows that modifying the original positional encoding through frequency tying introduces a small but measurable error. Thus, frequency tying is not strictly lossless; however, when restricted to the selected low-frequency pairs, the degradation in downstream FP16 model performance remains limited.

### C.6 ICE Group Formation Examples

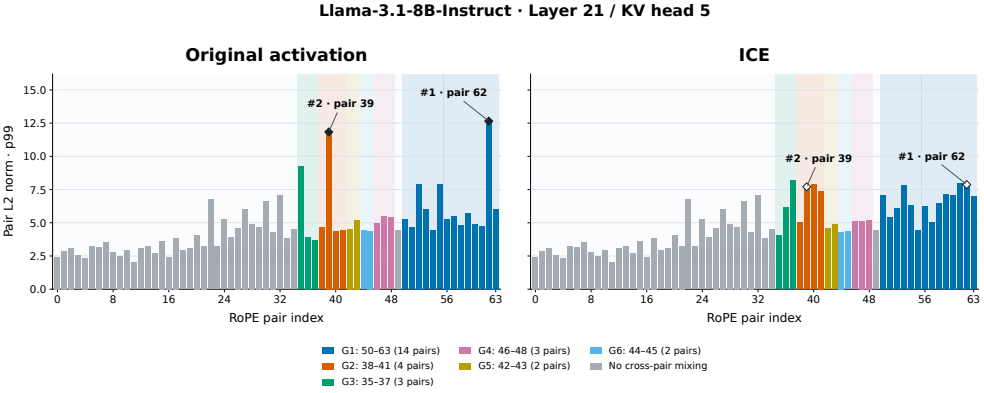
Table 14: ICE grouping statistics across all layers and KV heads. We report the fraction of RoPE pairs assigned to ICE groups, the average number of groups per KV head, and the mean and maximum group sizes.

| Model | ICE pair coverage (%) | Groups / KV head | Group size mean | Max |
|---|---|---|---|---|
| Llama-3.1-8B | 44.73 | 6.13 | 4.67 | 27 |
| Qwen3-8B | 42.19 | 8.15 | 3.32 | 27 |
| Qwen3-14B | 42.57 | 7.99 | 3.41 | 30 |

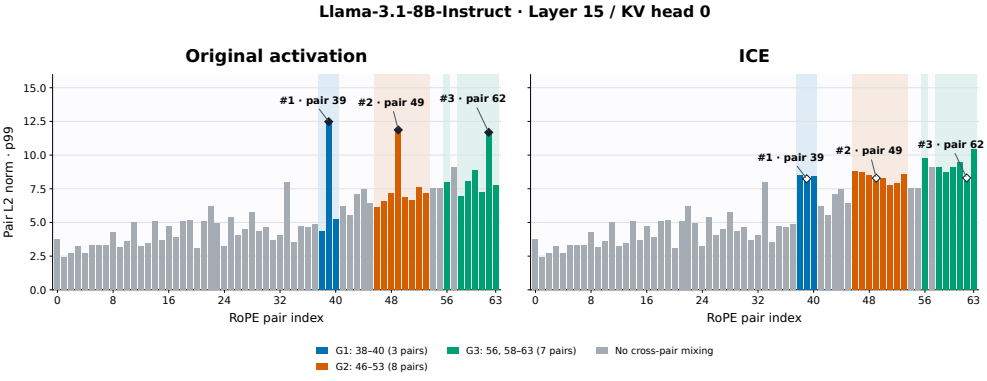
Figure 9 illustrates representative grouping patterns produced by the ICE group search in Algorithm 1 across different layers and KV heads. ICE first selects the pair with the largest outlier score as the seed, then adds unassigned candidates in ascending order of angular-frequency distance from the seed. Each time a candidate is added, the attention-output loss $L_{\text{out}}$ is evaluated, and the group configuration achieving the lowest loss during the search is selected. The pairs in the selected group are then excluded from subsequent searches, so repeating this process can produce multiple groups with different sizes and compositions within the same KV head.




(a) Layer 5 / KV head 7




(b) Layer 21 / KV head 5




(c) Layer 15 / KV head 0

Figure 9: Representative ICE group formation patterns in Llama-3.1-8B-Instruct. In each example, the left and right panels show the activation distributions before and after applying ICE, respectively, and pairs shown in the same color belong to the same ICE group. (a) shows a case where large activations are redistributed within a relatively broad group; (b) shows a case where multiple groups of different sizes coexist within a single KV head; and (c) shows a case where a non-contiguous group is formed while skipping some pairs.

**Distributing large activations across broad groups.** Figure 9a shows the case of Layer 5, KV head 7. Pairs 53 and 38 exhibit noticeably larger activation magnitudes than their neighboring pairs and are therefore treated as major seeds. Through frequency-neighbor expansion, pair 53 forms a group with 19 pairs spanning 45–63, while pair 38 forms a group with 9 pairs spanning 36–44. After applying ICE, the large activation peaks associated with the two seed pairs are reduced, while the activations of other pairs in the same groups increase. This illustrates how activations concentrated in a small number of dominant pairs can be redistributed across multiple frequency-compatible pairs.

**Coexistence of multiple groups with different sizes.** Figure 9b shows the case of Layer 21, KV head 5. A total of six ICE groups are formed within a single head, with group sizes ranging from 2 to 14 pairs. Pair 62 belongs to a relatively large group spanning 50–63, whereas pairs 39 and 35 belong to smaller groups spanning 38–41 and 35–37, respectively. Because Algorithm 1 sequentially expands candidates for each seed and selects the configuration that minimizes $L_{\text{out}}$, it does not use a fixed group size. As a result, groups of different sizes can coexist within the same KV head.

**Formation of groups that skip intermediate pairs.** Figure 9c shows the case of Layer 15, KV head 0. Three groups containing pairs 39, 49, and 62 are formed. In particular, the third group contains pair 56 and pairs 58–63, while excluding pair 57. ICE does not use pair-index contiguity as either a grouping constraint or a stopping criterion, and instead searches the remaining candidates according to their frequency distance from the seed. If the search were terminated simply because an intermediate pair was not included in the group, later candidates that could further reduce $L_{\text{out}}$ would never be evaluated. ICE therefore does not enforce contiguous groups, allowing it to search more flexibly for candidates that reduce the output error. The non-contiguous group in Figure 9c illustrates this search behavior in practice.

# D RoPE Compatibility

This section explains how differences in frequency across pairs constrain the space of rotations that are compatible with RoPE. To fold a rotation applied after RoPE directly into the Q/K projection weights, applying RoPE and the rotation in either order must produce the same result for every position $t$. Therefore, the following condition is required:

$$D_t U = U D_t \qquad \forall t. \tag{7}$$

## D.1 Pair-wise Rotation

Within an individual pair, the rotation angles of two 2-D rotations simply add:

$$\begin{aligned} R(t\omega_p)R(\phi_p) &= R(t\omega_p + \phi_p) \\ &= R(\phi_p)R(t\omega_p) \end{aligned} \tag{8}$$

Thus, pair-wise rotation commutes with RoPE and can be folded into the Q/K projection weights.

However, since $R(\phi_p)$ is orthogonal,

$$\begin{aligned} \|R(\phi_p)k_p\|_2^2 &= k_p^\top R(\phi_p)^\top R(\phi_p)k_p \\ &= \|k_p\|_2^2 \end{aligned} \tag{9}$$

holds. That is, pair-wise rotation can redistribute values between the two components within a pair, but cannot redistribute a norm concentrated in one pair to other pairs.

## D.2 Cross-pair Rotation with Distinct Frequencies

For two RoPE pairs $p$ and $q$, let $R_p = R(t\omega_p)$ and $R_q = R(t\omega_q)$. The RoPE operator applied to the two pairs and an example rotation that redistributes values across the pairs can be written as

$$D_t^{(p,q)} = \begin{bmatrix} R_p & 0 \\ 0 & R_q \end{bmatrix}, \qquad U_{pq} = \begin{bmatrix} \cos\alpha\, I_2 & -\sin\alpha\, I_2 \\ \sin\alpha\, I_2 & \cos\alpha\, I_2 \end{bmatrix} \tag{10}$$

where $I_2$ is the $2 \times 2$ identity matrix and $\alpha$ controls the degree of mixing between the two pairs. $U_{pq}$ is orthogonal and satisfies $U_{pq}^\top U_{pq} = I_4$.

Applying the two operations in the two possible orders gives

$$\begin{aligned} D_t^{(p,q)} U_{pq} &= \begin{bmatrix} \cos\alpha\, R_p & -\sin\alpha\, R_p \\ \sin\alpha\, R_q & \cos\alpha\, R_q \end{bmatrix}, \\ U_{pq} D_t^{(p,q)} &= \begin{bmatrix} \cos\alpha\, R_p & -\sin\alpha\, R_q \\ \sin\alpha\, R_p & \cos\alpha\, R_q \end{bmatrix}. \end{aligned} \tag{11}$$

Therefore, their difference is

$$D_t^{(p,q)} U_{pq} - U_{pq} D_t^{(p,q)} = \sin\alpha \begin{bmatrix} 0 & R_q - R_p \\ R_q - R_p & 0 \end{bmatrix}. \tag{12}$$

When actual cross-pair mixing occurs, i.e., $\sin\alpha \neq 0$, this difference becomes zero only if $R_p = R_q$. At a particular position, different frequencies may occasionally yield the same rotation angle modulo $2\pi$, but weight folding requires compatibility at all positions. Under the standard RoPE frequency range, two pairs with different frequencies do not receive identical rotations at every integer position. Therefore, the above cross-pair rotation does not generally satisfy Eq. (7).

### D.3 Cross-pair Rotation with Shared Frequencies

If two pairs share a common frequency $\nu$, then $R_p = R_q = R(t\nu)$ holds at every position. Accordingly, the right-hand side of Eq. (12) becomes zero, allowing cross-pair rotations that commute with RoPE.

Extending this to multiple pairs, let $m = |G|$ denote the number of pairs in an isofrequency group $G$ with shared frequency $\nu_G$. The corresponding RoPE operator is

$$\widetilde{D}_t^G = I_m \otimes R(t\nu_G). \tag{13}$$

Let

$$J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \tag{14}$$

so that $R(\theta) = \cos\theta\, I_2 + \sin\theta\, J$. A real matrix commutes with $\widetilde{D}_t^G$ for all $t$ if it has the form

$$T_G = A_G \otimes I_2 + B_G \otimes J, \tag{15}$$

where $A_G, B_G \in \mathbb{R}^{m\times m}$. Equivalently, defining the complex matrix

$$U_G = A_G + iB_G, \tag{16}$$

the commuting transformation space is isomorphic to $M_m(\mathbb{C})$.

Restricting $T_G$ to be orthogonal corresponds to requiring $U_G$ to be unitary:

$$U_G^\dagger U_G = I_m. \tag{17}$$

Thus, the orthogonal transformations that commute with the shared-frequency RoPE are isomorphic to $U(m)$. For any such transformation,

$$\widetilde{D}_t^G T_G = T_G \widetilde{D}_t^G, \qquad \forall t, \tag{18}$$

and

$$T_G^\top T_G = I_{2m}. \tag{19}$$

Under distinct RoPE frequencies, the foldable orthogonal transformation space is restricted to independent rotations within individual pairs, corresponding to $U(1)^m$ with $m$ degrees of freedom. Frequency tying expands this space to $U(m)$, which has $m^2$ degrees of freedom. This expansion enables ROKI to redistribute activation magnitude across multiple pairs while preserving RoPE compatibility.

During ICE group search, ROKI samples a random $U_G \in U(m)$ using a QR decomposition of a complex Gaussian matrix and converts it to the equivalent real transformation $T_G$ above. This random rotation is used only as a lightweight proxy for candidate group evaluation and is discarded

once the group structure is fixed. For the final learned transformation, ROKI parameterizes the unitary rotation using a matrix exponential,

$$U_G = \exp(H_G), \qquad H_G^\dagger = -H_G, \tag{20}$$

which preserves unitarity throughout optimization.

Moreover, when the same $T_G$ is applied to both Q and K, the group-wise QK inner product between positions $t$ and $j$ is preserved:

$$\begin{aligned}(\widetilde{D}_t^G T_G q_G)^\top (\widetilde{D}_j^G T_G k_G) &= (T_G \widetilde{D}_t^G q_G)^\top (T_G \widetilde{D}_j^G k_G) \\ &= (\widetilde{D}_t^G q_G)^\top (\widetilde{D}_j^G k_G).\end{aligned} \tag{21}$$

This preservation property describes the relationship before and after adding the rotation under the shared-frequency configuration. It does not imply that frequency tying itself preserves the attention scores of the original model using the native frequencies.

# E Additional Evaluation Results

## E.1 Needle-in-a-Haystack

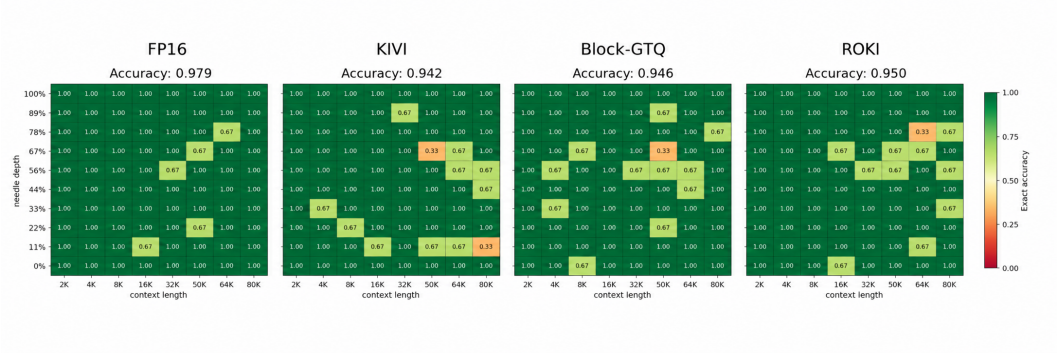



Figure 10: Needle-in-a-Haystack retrieval accuracy on Llama-3.1-8B-Instruct across different context lengths and answer insertion positions. The x-axis denotes context length, while the y-axis indicates the relative position of the answer within the context. Each cell shows the accuracy under the corresponding condition, and the value above each panel reports the average accuracy across all conditions.

## E.2 Additional Accuracy

Table 15: WikiText-2 perplexity on Qwen3-8B under additional KV-cache precision settings. Except for KVTuner, all quantized methods use uniform 3-bit quantization for the V cache.

| Method | Precision setting | PPL ↓ |
|---|---|---|
| FP16 | FP16 | 7.27 |
| ROKI | K 2.5-bit | 7.50 |
| | K 3-bit | 7.31 |
| | K 4-bit | 7.31 |
| TurboQuant | 3-bit | 8.23 |
| KVTuner | KV avg. 3-bit | 8.54 |

TurboQuant (Zandieh et al., 2026) is evaluated in a 3-bit setting that combines 2-bit MSE quantization with 1-bit QJL. KVTuner (Li et al., 2025) selects layer-wise K/V precisions such that the average precision of the entire KV cache is 3 bits, resulting in average precisions of 3.83 bits for K

and 2.17 bits for V. All quantized methods except KVTuner use uniform 3-bit quantization for the V cache.

In Table 15, ROKI achieves a PPL of 7.50 even when the average K-cache precision is reduced to 2.5 bits, outperforming TurboQuant (8.23) and KVTuner (8.54). This shows that ROKI maintains lower perplexity than TurboQuant and KVTuner even with the K-cache precision reduced to 2.5 bits.

# F Implementation and Evaluation Details

## F.1 RoPE Pair Layout

When RoPE pairs are stored separately in the original hidden-dimension layout as $(p, p + d/2)$, we first permute the dimensions such that each pair occupies adjacent positions $(2p, 2p + 1)$. For example, with a head dimension of 128, the original RoPE pairs $(0, 64)$, $(1, 65)$, ... are rearranged into the contiguous layout $(0, 1)$, $(2, 3)$, ....

## F.2 Group Search and Optimization

For each layer and KV head, we use the low-frequency RoPE pairs 32–63 as ICE candidates. We first select pairs with large average pair-wise $\ell_2$ norms of the original FP16 keys as anchors. For each anchor, unassigned pairs are progressively added in ascending order of angular-frequency distance from the anchor. During group search, we evaluate calibration sequences with context lengths of 2K, 8K, 16K, 32K, 64K, and 128K. For each candidate group, we align the group frequency with that of the anchor, apply a random unitary rotation, and measure the attention-output MSE relative to the original FP16 output across these context lengths. The candidate group that achieves the lowest attention-output MSE is selected.

For calibration, we use WikiText-2 and optimize the transformation parameters by minimizing the MSE with respect to the FP16 attention output. With the ICE group assignments fixed, we first optimize the rotations for 1,024 steps using Adam with a learning rate of $5 \times 10^{-4}$. We then compute post-transformation QK importance and permute the pairs into the fixed 2/4-bit mixed-precision layout. With the resulting pair layout fixed, we optimize the scaling parameters for an additional 1,024 steps.

## F.3 Offline Weight Folding and Inference

The learned rotation, permutation, and scaling are fixed linear transformations and can therefore be combined into a single transform. Let $R$, $P$, and $S$ denote the rotation, pair-permutation, and pair-wise scaling matrices, respectively. The final Q/K transformations are

$$q' = SPRq, \qquad k' = S^{-1}PRk, \tag{22}$$

which can be folded into the projection weights as

$$W'_Q = SPRW_Q, \qquad W'_K = S^{-1}PRW_K. \tag{23}$$

The RoPE frequencies are reordered consistently with the pair permutation. Since $R$ commutes with the frequency-tied RoPE and $S$ applies the same scale to both components of each pair, the pre-quantization attention score is preserved under the modified RoPE.

At inference, ROKI uses the folded projection weights and reordered RoPE frequencies, requiring no additional online Q/K transformation.

**Algorithm 1:** ICE Group Search

**Input:** Candidates $\mathcal{P}_{\mathrm{ICE}}$, angular frequencies $\{\omega_p\}$, outlier scores $\{a_p\}$
**Output:** Groups $\mathcal{S}$ and configuration $\mathcal{C}$

$\pi \leftarrow \mathrm{Sort}_{\uparrow}(\mathcal{P}_{\mathrm{ICE}};\ (-a_p, p))$
$\mathcal{S} \leftarrow \emptyset, \quad \mathcal{C} \leftarrow \mathcal{C}_0$
$\ell \leftarrow \mathcal{L}_{\mathrm{out}}(\mathcal{C})$
**foreach** $s \in \pi$, *in order* **do**
  **if** $s \in \mathrm{Pairs}(\mathcal{S})$ **then continue**
  $\mathcal{A} \leftarrow \mathcal{P}_{\mathrm{ICE}} \setminus (\mathrm{Pairs}(\mathcal{S}) \cup \{s\})$
  $\rho \leftarrow \mathrm{Sort}_{\uparrow}(\mathcal{A};\ (|\omega_p - \omega_s|, p))$
  $G \leftarrow \{s\}$
  $(G_*, \mathcal{C}_*, \ell_*) \leftarrow (\emptyset, \mathcal{C}, \ell)$
  **foreach** $p \in \rho$, *in order* **do**
    $G \leftarrow G \cup \{p\}$
    $U_G \leftarrow \mathrm{RandomUnitary}(|G|)$
    $\mathcal{C}' \leftarrow \mathrm{Copy}(\mathcal{C})$
    $\mathcal{C}' \leftarrow \mathrm{ApplyICE}(\mathcal{C}', G, \omega_s, U_G)$
    $\ell' \leftarrow \mathcal{L}_{\mathrm{out}}(\mathcal{C}')$
    **if** $\ell' < \ell_*$ **then**
      $(G_*, \mathcal{C}_*, \ell_*) \leftarrow \mathrm{Copy}(G, \mathcal{C}', \ell')$
  **if** $G_* \neq \emptyset$ **then**
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{G_*\}$
    $\mathcal{C} \leftarrow \mathcal{C}_*, \quad \ell \leftarrow \ell_*$
**return** $\mathcal{S}, \mathcal{C}$

### F.4 Kernel Optimization

ROKI stores the KV cache contiguously in per-token quantization blocks, with a memory layout that enables regular access to the quantized values and their corresponding scales. Rather than dequantizing the entire compressed KV cache in advance, the kernel reconstructs only the values required for the current attention computation on the fly and directly consumes them in the dot-product operation. Under GQA, multiple query heads sharing the same KV head reuse the corresponding KV loads and reconstructed values, reducing redundant memory accesses and dequantization operations. In addition, ROKI exploits the fact that multiple key elements within a quantization block share the same scale. Instead of individually dequantizing each element before computing the dot product, the kernel first accumulates partial dot products over the quantized values and applies the shared scale afterward. This reduces the overhead associated with element-wise dequantization and scale multiplications. For KIVI and Block-GTQ, we use the CUDA kernels provided in their official repositories.

**Latency baseline selection.** We evaluate latency using KIVI, Block-GTQ, and QuaRot, whose runtime KV-cache execution paths differ from that of ROKI. KIVI applies per-channel quantization to the K cache, requiring channel-wise unpacking and dequantization during decoding. Block-GTQ uses layer- and head-specific non-uniform bit allocation together with TurboQuant-based reconstruction, resulting in a more complex decoding path that includes quantized-code lookup and rotation. QuaRot performs an online Hadamard transformation in the KV-cache attention path and reports additional runtime overhead from this operation (Ashkboos et al., 2024).

In contrast, ROKI folds all Q/K transformations into the projection weights, preserves a regular per-token KV-cache layout, and uses an optimized attention kernel for this structure. The corresponding Q/K transformations in FPTQuant and QServe can also be fused into the projection weights; therefore, they are excluded from the latency comparison focused on online transformation overhead.

### F.5 Effective KV-Cache Storage Cost

Table 16 summarizes the effective KV-cache storage cost including quantization metadata and FP16 residual caches.

Table 16: Effective KV-cache storage cost at a 32K context length with head dimension 128, including quantization metadata and FP16 residual caches.

| Method | KV quantization | Metadata / residual | Eff. bits/value |
|---|---|---|---|
| KIVI | K3/V3, group 32 | FP16 scale + zero-point; residual 128 | 4.047 |
| Block-GTQ | Avg. 3-bit K/V | FP16 norms; per-head codebook; no residual | 3.423 |
| ROKI | K2/2/4/4, V3, group 32 | FP16 scale; residual 128 | 3.549 |

Even under the same nominal 3-bit KV-cache setting, the actual storage cost differs due to quantization metadata and residual caches. KIVI stores an FP16 scale and zero-point for each group of 32 values and retains the most recent 128 tokens in an FP16 residual cache. Block-GTQ stores quantized codes according to layer- and head-specific bit allocations, together with multiple FP16 norm values and per-head codebooks, and does not use a residual cache. ROKI stores one FP16 scale for each group of 32 values and retains the most recent 128 tokens in FP16. Including these overheads, the effective storage at a 32K context length is 4.047, 3.423, and 3.549 bits/value for KIVI, Block-GTQ, and ROKI, respectively. Thus, ROKI requires less effective storage than KIVI while maintaining a storage budget comparable to Block-GTQ.

### F.6 LongBench-E

We evaluate long-context understanding on 13 tasks from LongBench-E. The tasks span six categories: Single-document QA, Multi-document QA, Summarization, Few-shot learning, Synthetic, and Code. We use the prompt templates provided by the official LongBench-E implementation and follow the official evaluation metric for each task. The overall score is reported as the equally weighted average across all 13 tasks.

### F.7 Needle-in-a-Haystack (NIAH)

We evaluate long-context information retrieval using a Needle-in-a-Haystack (NIAH) setup with distractors of the same format. Each document contains one target 16-digit access token and 16 distractor tokens, and the model is asked to retrieve the access token corresponding to the specified project. Evaluation is performed over eight context lengths ranging from 2K to 80K tokens and ten insertion depths spanning 0–100%. The maximum generation length is set to 32 tokens, and performance is measured by the retrieval accuracy of the target access token.

### F.8 Reasoning and Code Generation

We evaluate mathematical and scientific reasoning and code generation using `lm-evaluation-harness`. GSM8K is evaluated in an 8-shot chain-of-thought (CoT) setting using exact-match accuracy. MATH-Algebra is evaluated on 1,187 examples in a 4-shot setting using answer exact match. HumanEval is evaluated on 164 problems in a 0-shot setting and reported with pass@1. GPQA-Diamond is evaluated on 196 examples in a 5-shot CoT setting using answer exact match. For all benchmarks, the maximum context length is set to 16,384 tokens and the maximum generation length to 4,096 tokens.