FROM FREQUENCY TO FUNCTION: COMPONENT-SPECIFIC ROPE EXTRAPOLATION FOR LONG-CONTEXT LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Rotary Position Embedding (RoPE)-based language models form position-sensitive attention by superimposing contributions from dimension pairs associated with different rotation frequencies. Long-context extrapolation therefore requires rescaling rotation frequencies. However, common frequency-based methods partition dimensions by the relative ordering of their associated RoPE frequencies and scale attention logits with a single global factor. This design overlooks the distinct functional roles that dimension pairs acquire during pretraining. We group dimension pairs into three classes: positional pairs have concentrated pre-RoPE activation phases and produce stable relative-position periodicity, semantic pairs are dominated by input content variation, and sink pairs make exceptionally large positive contributions to sink-token attention logits. Based on this decomposition, we propose FARoPE (Function-Aware RoPE), a training-free RoPE extrapolation method that first identifies the three pair classes and then applies component-specific transformations. Specifically, FARoPE applies a shared frequency scale to positional pairs to prevent spurious shifts in position-sensitive attention patterns. Sink pairs receive the same frequency reduction to maintain sink alignment, while semantic pairs retain the YaRN ramp. FARoPE separately calibrates the additional attention-logit gain for sink pairs without masking sink tokens. Our evaluation covers two model families and three long-context benchmarks with context windows up to 128K.

1 INTRODUCTION

Long-context understanding determines how much evidence a large language model (LLM) can jointly use for retrieval, synthesis, and reasoning. Most LLMs, however, are optimized within a finite context window, and their perplexity and downstream accuracy often deteriorate sharply once inference exceeds the lengths observed during pretraining (Chen et al., 2023; Hsieh et al., 2024). Long-context extrapolation seeks to extend this usable range without sacrificing the capabilities learned at shorter lengths. Positional encoding is central to this problem because it makes tokens distinguishable by position and directly shapes attention geometry. Rotary Position Embedding (RoPE), now widely used in LLMs, represents relative positions by applying position-dependent rotations to query and key vectors (Su et al., 2024). Beyond the pretrained window, previously unseen rotation phases and phase combinations become out of distribution, making RoPE a primary source of extrapolation failure (Liu et al., 2024).

One approach extends the training distribution through long-sequence continued pretraining or fine-tuning (Chen et al., 2024; Ding et al., 2024). Although effective, it requires substantial compute and suitably long training data. For example, LongRoPE's 128K training stage takes one week on eight A100 GPUs (Ding et al., 2024). Such training moves the model's supported window to the lengths covered by the new optimization but does not directly provide weight-update-free *train-short-test-long* extrapolation. This motivates training-free methods that extend an unchanged pretrained model at inference time.

Within full-context RoPE inference, training-free methods mainly follow two strategies. *Position remapping* changes the position indices supplied to attention. SelfExtend preserves ordinary posi-

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.

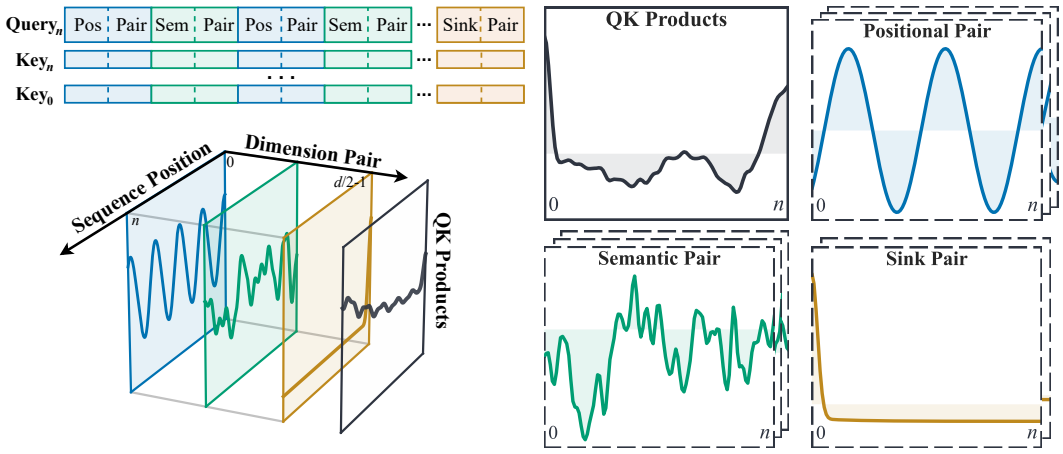

Figure 1: Functional heterogeneity among RoPE dimension pairs within an attention head. Each attention logit sums partial QK products from multiple pairs. Positional pairs produce stable periodic contributions over relative positions, semantic pairs vary mainly with content, and sink pairs make exceptionally large positive contributions to sink-token logits.

tions locally but groups distant positions, while DCA cyclically reuses positions within chunks and constrains cross-chunk positions (Jin et al., 2024; An et al., 2024). These designs keep effective relative distances near the pretrained range, but compressing or reusing coordinates can weaken exact long-range positional geometry. *Rotation-frequency scaling* instead changes how fast RoPE dimension pairs rotate while retaining the original token order, as in dynamic NTK-aware RoPE, YaRN, and MrROPE (Peng et al., 2024; Tian et al., 2026). Our work follows this second strategy because it preserves a distinct coordinate for every token; the remaining question is how each rotary subspace should be extrapolated.

Existing frequency-based methods usually answer this question from the magnitude or relative ordering of RoPE frequencies. YaRN divides the frequency spectrum with a fixed ramp, MrROPE assigns frequency bands different mixed-radix treatments, and LongRoPE searches dimension-wise interpolation factors using perplexity (Peng et al., 2024; Tian et al., 2026; Ding et al., 2024). These methods use different signals, including frequency order, distribution shift, and perplexity; they do not use the joint phase-concentration and sink-bias classification studied here. This omission matters because each attention logit is a sum of partial QK products from many pairs. Giving different scales to multiple position-sensitive pairs changes their relative phases: periodic contributions that previously combined at learned offsets can instead align at unintended distances and form spurious attention-logit peaks. Likewise, p-RoPE removes rotations only from the lowest-frequency dimensions, but frequency rank is merely a proxy for function and cannot separate different behaviors within the same band (Barbero et al., 2025b). Layer-wise RoPE/NoPE hybrids and adaptive head-specific scaling recognize heterogeneity across layers and attention heads (Yang et al., 2025b; Wang et al., 2026). These methods use coarser intervention units than dimension pairs, and their training or optimization procedures also differ from our fixed-weight setting.

Attention-logit temperature scaling introduces a second granularity mismatch. YaRN applies one global factor to the sum of all dimension-pair contributions (Peng et al., 2024). This compensates for length-induced changes in attention sharpness, but it also magnifies any exceptionally large positive contribution to an attention-sink token (Xiao et al., 2024; Yu et al., 2024). These observations motivate testing dimension pairs as a finer intervention unit alongside frequency bands and whole heads. As Figure 1 illustrates, dimension pairs within the same head can exhibit three distinct behaviors: *positional pairs* produce stable periodic partial QK products over relative positions, *semantic pairs* vary primarily with content, and *sink pairs* make exceptionally large positive contributions to sink-token attention logits. The working hypothesis is that pair-specific measurements can select useful extrapolation rules beyond what frequency order alone provides.

We therefore propose FARoPE, a training-free, function-aware RoPE extrapolation method. A *dimension pair* denotes the two coordinates jointly transformed by one frequency-specific 2×2 rotary

block; its partial QK product forms one component of an attention logit. FARoPE uses **Function-Aware Pair Classification** to identify positional, semantic, and sink pairs once from a small calibration corpus and reuses their class labels across inputs. Its **Component-Specific Frequency Scaling** applies a shared frequency scale to positional pairs, preserving a common remapping of their relative-distance contributions. Semantic pairs follow the YaRN ramp, while sink pairs receive the maximum frequency reduction, $1/f$, to limit rotation-induced disruption of sink alignment. Its **Component-Specific Logit Calibration** separately calibrates the additional logit gain applied to sink-pair contributions without masking sink tokens or removing their learned function. The two modules act at the dimension-pair level without updating model weights.

Our main contributions are:

- We formulate behavior-based dimension-pair selection for RoPE extrapolation using phase concentration and sink bias. The resulting positional, semantic, and sink classes define different interventions within the same attention head.
- We introduce FARoPE, which combines Component-Specific Frequency Scaling with Component-Specific Logit Calibration. Pair classes are identified only once and then reused, requiring no pretrained-weight updates or additional online forward passes.
- We compare FARoPE with training-free extension methods on two model families and three long-context benchmarks, covering both $4\times$ and $16\times$ extensions.

A detailed discussion of related work appears in Appendix E.

2 Preliminaries

Let ℓ, h, and d denote the layer, key-value (KV) head, and head dimension. Under grouped-query attention (GQA) (Ainslie et al., 2023), pair classes and tracking ratios are shared within each KV group. Appendix A.3 specifies query-head aggregation.

2.1 RoPE and Dimension Pairs

RoPE partitions each query or key vector into $d/2$ two-dimensional rotary subspaces (Su et al., 2024). We call the two coordinates transformed by the same rotary block a *dimension pair*. Under the split-half implementation used by the studied models, pair p contains $(x_p, x_{p+d/2})$. For token t and $u \in \{q, k\}$, which selects a query or key activation, we write

$$a^u_{\ell,h,p,t} = u_{\ell,h,t,p} + \mathrm{i}\, u_{\ell,h,t,p+d/2}, \qquad \omega_p = \theta^{-2p/d}. \tag{1}$$

Here $a^u_{\ell,h,p,t}$ is the complex representation of the pair, i is the imaginary unit, ω_p is its pretrained rotation frequency, and θ is the RoPE base.

2.2 Dimension-Pair Decomposition of Attention Logits

For query position i and key position j, pair p contributes the partial attention logit

$$z^{(p)}_{ij,\ell,h} = \frac{1}{\sqrt{d}} \mathrm{Re}\left(a^q_{\ell,h,p,i}\overline{a^k_{\ell,h,p,j}} e^{\mathrm{i}(i-j)\omega_p}\right), \qquad z_{ij,\ell,h} = \sum_{p=0}^{d/2-1} z^{(p)}_{ij,\ell,h}. \tag{2}$$

The overline denotes complex conjugation and Re takes the real part. This decomposition shows that changing one rotation frequency changes only the relative-position pattern contributed by its pair, while the final logit combines all pair-wise patterns.

2.3 YaRN Scaling

Let L_0 be the native context length, L_{tgt} the target context length, and $f = L_{\mathrm{tgt}}/L_0$ the extension ratio. Standard YaRN uses a pair-dependent ramp to interpolate between the pretrained frequency and uniform $1/f$ scaling (Peng et al., 2024):

$$r_p = \mathrm{clip}\left(\frac{p - p_{\mathrm{fast}}}{p_{\mathrm{slow}} - p_{\mathrm{fast}}}, 0, 1\right), \qquad \omega^{\mathrm{Y}}_p = \omega_p\left[(1 - r_p) + \frac{r_p}{f}\right]. \tag{3}$$

Here p_{fast} and p_{slow} are YaRN's fixed ramp endpoints, clip clamps its input to $[0, 1]$, and ω_p^{Y} is the resulting YaRN-scaled rotation frequency. For an actual inference length L, FARoPE uses the following length-dependent adaptation of the YaRN attention-logit gain:

$$\gamma(L) = \left(1 + 0.1 \ln \max\left\{1, \frac{L}{L_0}\right\}\right)^2. \tag{4}$$

The squared factor accounts for scaling both queries and keys. Frequencies use the fixed target factor f, while gain uses the actual non-padding length L: prompt length during prefill and current key length during decoding. This differs from fixing both factors at the target length.

Uniform gain can amplify attention sinks, tokens that receive substantial attention across many queries (Xiao et al., 2024; Yu et al., 2024). FARoPE retains these tokens and calibrates the contributing dimension pairs.

3 Methodology

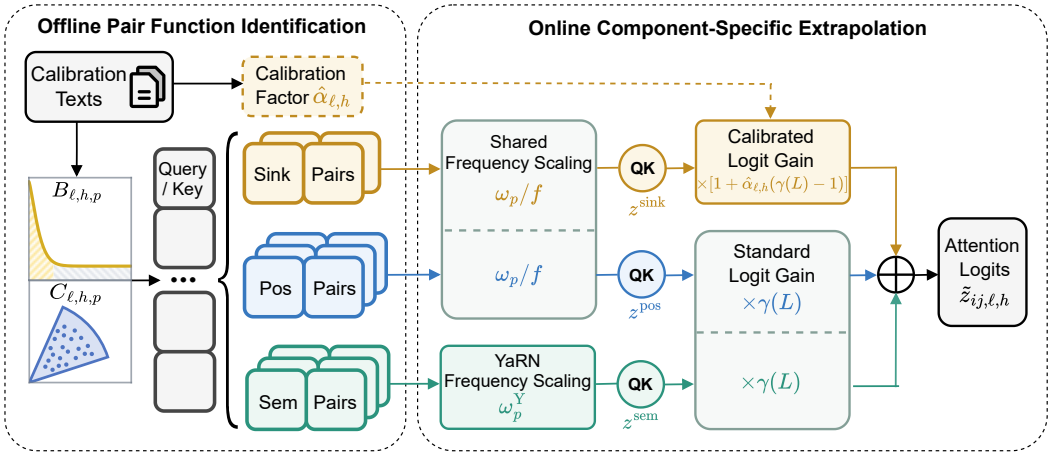

Figure 2: Overview of FARoPE. Offline calibration uses Sink Bias $B_{\ell,h,p}$ and Phase Concentration $C_{\ell,h,p}$ to classify dimension pairs, with sink pairs identified first. Sink and positional pairs share frequency ω_p/f, while semantic pairs use ω_p^{Y}. Class-wise partial logits receive either the calibrated sink gain or the standard gain $\gamma(L)$ before summation. The dashed arrow supplies $\hat{\alpha}_{\ell,h}$, fitted offline after fixing the pair classes and final frequencies. QK denotes the summed, $1/\sqrt{d}$-normalized pair products within each class.

3.1 Overview

FARoPE uses the functional roles of dimension pairs to select component-specific RoPE transformations. Figure 2 summarizes the procedure. Offline, **Function-Aware Pair Classification** uses phase concentration and sink bias to identify positional, semantic, and sink pairs. A subsequent fit determines one sink tracking ratio for each layer–KV-head group. Online, **Component-Specific Frequency Scaling** applies a shared frequency reduction to positional and sink pairs and the YaRN ramp to semantic pairs. **Component-Specific Logit Calibration** applies the standard gain to positional and semantic contributions and a calibrated gain to sink contributions. No pretrained model weight is updated.

3.2 Function-Aware Pair Classification

Figure 3 shows the behavior of each class. Positional pairs have concentrated pre-RoPE phases that provide a stable angular offset to which RoPE adds relative-position rotation. After sink pairs are excluded, semantic pairs are characterized by dispersed pre-RoPE activation phases. Their partial QK products vary primarily with input content, while relative position has a weaker influence. This

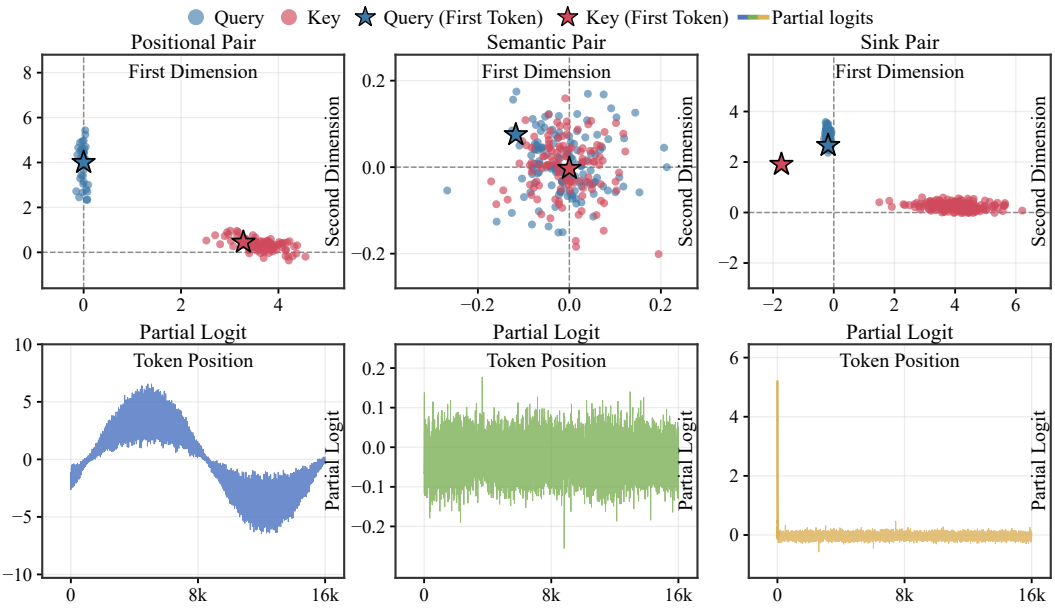

Figure 3: Dimension-pair behaviors. Columns show positional, semantic, and sink pairs, respectively. Top: pre-RoPE query and key activations; stars mark first-token activations. Bottom: partial logits over token positions, showing stable periodicity, irregular content variation, and a pronounced first-token contribution, respectively.

content sensitivity makes these pairs well suited to carrying semantic information about tokens and their context. Sink pairs are identified by their contribution to first-token preference.

3.2.1 Phase Concentration

We estimate phase concentration only from calibration sequences no longer than L_0, so the statistic reflects a pattern learned during pretraining rather than an artifact of extrapolation. For pair p in layer ℓ and head h, its magnitude-weighted phase concentration is

$$C_{\ell,h,p} = \frac{1}{2} \sum_{u \in \{q,k\}} \frac{\left| \sum_t a^u_{\ell,h,p,t} \right|}{\sum_t \left| a^u_{\ell,h,p,t} \right|}. \tag{5}$$

Both sums include valid calibration tokens. The score approaches one when each side has a stable phase direction, even if query and key directions differ. For GQA, we average query-head concentrations within each KV group before combining query and key scores; numerical conventions appear in Appendix A.3.

3.2.2 Sink Bias

Phase concentration cannot distinguish a stable positional pattern from a stable sink contribution. For each calibration query i, let j_0 be the first non-padding position, $\mathcal{S}_i = \{j_0\}$, and $\mathcal{C}_i = \{j : j_0 < j < i,\ j \text{ is non-padding}\}$. Queries with an empty context set are excluded. This protocol targets first-token sinks and does not automatically detect secondary sinks. At each (ℓ, h), the full reference logits define separate softmax distributions $P_{\mathcal{S}}$ and $P_{\mathcal{C}}$ within these two sets. Reference logits use native RoPE at lengths up to L_0 and standard YaRN at longer calibration lengths. The partial logits in the following equation are computed under the same reference configuration. The signed sink bias is

$$B_{\ell,h,p} = \mathbb{E}_{i,L} \left[\mathbb{E}_{s \sim P_{\mathcal{S}}} \left[z^{(p)}_{is,\ell,h} \right] - \mathbb{E}_{c \sim P_{\mathcal{C}}} \left[z^{(p)}_{ic,\ell,h} \right] \right]. \tag{6}$$

The outer expectation averages calibration queries i and sequence lengths L. Separate within-set normalization prevents the number of sink or context positions from determining the comparison. A large positive $B_{\ell,h,p}$ means that pair p consistently favors sink positions in layer ℓ and head h.

Pair classification. Let $\mathcal{P}_{\ell,h} = \{0, \ldots, d/2 - 1\}$ be all pairs in the current layer-head. We set a robust sink-bias threshold $\tau^B_{\ell,h}$ to the larger of a positive floor and the median sink bias plus three raw median absolute deviations. The positive floor is 0.10, and the phase threshold is $\tau_C = 0.96$. Appendix A.1 gives the complete calibration data and parameter settings. The three classes are defined entirely by threshold comparisons:

$$\begin{aligned} \mathcal{P}^{\text{sink}}_{\ell,h} &= \left\{p \in \mathcal{P}_{\ell,h} : B_{\ell,h,p} > \tau^B_{\ell,h}\right\}, \\ \mathcal{P}^{\text{pos}}_{\ell,h} &= \left\{p \in \mathcal{P}_{\ell,h} : B_{\ell,h,p} \le \tau^B_{\ell,h},\ C_{\ell,h,p} \ge \tau_C\right\}, \\ \mathcal{P}^{\text{sem}}_{\ell,h} &= \left\{p \in \mathcal{P}_{\ell,h} : B_{\ell,h,p} \le \tau^B_{\ell,h},\ C_{\ell,h,p} < \tau_C\right\}. \end{aligned} \tag{7}$$

These conditions are mutually exclusive and exhaustive. The sink-first threshold preserves sink pairs as an independent class even when their phases are concentrated, and the robust threshold lets different layer-heads contain different numbers of sink pairs, including none.

3.3 Component-Specific Frequency Scaling

This module selects the frequency transformation for each pair from its functional class. We apply one common $1/f$ frequency scale to every positional pair. In the default configuration, semantic pairs retain the standard YaRN frequency ω^{Y}_p defined in Equation 3, while sink pairs also use the maximum reduction ω_p/f:

$$\omega'_{\ell,h,p} = \begin{cases} \omega_p/f, & p \in \mathcal{P}^{\text{pos}}_{\ell,h}, \\ \omega^{\text{Y}}_p, & p \in \mathcal{P}^{\text{sem}}_{\ell,h}, \\ \omega_p/f, & p \in \mathcal{P}^{\text{sink}}_{\ell,h}. \end{cases} \tag{8}$$

Here $\omega'_{\ell,h,p}$ is the inference frequency; $1/f$ is the strongest reduction in $[1/f, 1]$.

To see what the shared scale preserves, hold pre-RoPE activations fixed and write $b_p = a^q_{p,i}\overline{a^k_{p,j}}/\sqrt{d}$. The positional contribution satisfies

$$F(\delta) = \sum_{p \in \mathcal{P}^{\text{pos}}} \text{Re}(b_p e^{\text{i}\delta\omega_p}), \qquad F_{\text{shared}}(\delta) = F(\delta/f). \tag{9}$$

Thus all selected pairs use one common distance remapping. This preserves the shape of their combined contribution under that remapping, rather than its peak at the original token position. The identity concerns fixed activations; it does not guarantee preservation of the full attention distribution.

Why scale sink frequencies? For a nonzero pair, let $\psi_{ij,p} = \arg a^q_{p,i} - \arg a^k_{p,j}$. Its normalized contribution after RoPE is $\cos(\psi_{ij,p} + (i-j)\omega_p)$. Additional rotation at long distances can disrupt alignment with the first-token key, depending on the learned phase (Wertheimer et al., 2026). Scaling by $1/f$ keeps the target-window rotation range within the native-window range. This frequency adjustment addresses query–sink geometry; the following gain calibration controls additional amplification of sink contributions.

3.4 Component-Specific Logit Calibration

The YaRN logit gain $\gamma(L)$ in Equation 4 can over-amplify sink-pair contributions when applied uniformly, even though positional and semantic pairs still benefit from it. Using the final frequencies $\omega'_{\ell,h,p}$, let $z^{\text{pos}}_{ij,\ell,h}$, $z^{\text{sem}}_{ij,\ell,h}$, and $z^{\text{sink}}_{ij,\ell,h}$ denote the sums of the pair partial logits over the three classes. With an offline-fitted sink tracking ratio $\hat{\alpha}_{\ell,h} \in [0, 1]$, the calibrated attention logit is

$$\widetilde{z}_{ij,\ell,h} = \gamma(L)\left(z^{\text{pos}}_{ij,\ell,h} + z^{\text{sem}}_{ij,\ell,h}\right) + \left[1 + \hat{\alpha}_{\ell,h}\left(\gamma(L) - 1\right)\right] z^{\text{sink}}_{ij,\ell,h}. \tag{10}$$

Thus, sink-pair logits in layer ℓ and head h are scaled by $1 + \hat{\alpha}_{\ell,h}(\gamma(L) - 1)$. When $\hat{\alpha}_{\ell,h} = 1$, all three classes recover YaRN's global gain. When $\hat{\alpha}_{\ell,h} = 0$, sink-pair contributions receive no additional long-context gain, while positional and semantic contributions remain unchanged.

Offline fit. We fit $\hat{\alpha}_{\ell,h}$ after fixing the pair classes and frequencies. Sink preference is measured by the difference between the count-normalized log-mean-exp of sink and context logits. A native-context high quantile provides the reference level. A first-order fit reduces preference above this reference, with $\alpha \in [0, 1]$ and a regularizer $10^{-3}(1 - \alpha)^2$ that returns underdetermined fits toward one. Appendix A.2 gives the objective and closed-form solution.

3.5 Inference

FARoPE reuses fixed pair classes and fitted tracking ratios without weight updates, added tokens, or extra forward passes. Applying the gains to queries lets the gain change during decoding while cached keys retain fixed frequencies. Standard attention kernels can consume the transformed queries and keys. Appendix F gives the equivalent dot-product formulation, cache conditions, and complexity; actual latency and memory depend on the implementation.

4 Experiments

We evaluate whether dimension-pair-specific extrapolation can extend unchanged pretrained models to 128K tokens while preserving long-context retrieval, aggregation, understanding, and reasoning.

4.1 Experimental Setup

Models and context lengths. We evaluate Qwen3-8B and Llama 3 8B Instruct (Yang et al., 2025a; Grattafiori et al., 2024). We extend Qwen3-8B from its 32K native context window to 128K, a $4\times$ extension, and Llama 3 8B Instruct from 8K to 128K, a $16\times$ extension. The two settings test whether a method remains effective under both moderate and large extrapolation ratios.

Benchmarks and metrics. We use three complementary long-context benchmarks. RULER provides 13 controlled retrieval, aggregation, and question-answering tasks (Hsieh et al., 2024). We evaluate 100 examples per task at 4K, 8K, 16K, 32K, 64K, and 128K. We first average each task at each length and then macro-average the 13 tasks and six lengths. Infinite-Bench contains 12 retrieval, summarization, dialogue, reasoning, math, and code tasks beyond 100K tokens (Zhang et al., 2024). We evaluate up to 120 examples per task and macro-average the task-specific accuracy, F1, or ROUGE-L scores (Lin, 2004). LongBench v2 measures realistic long-context understanding with multiple-choice accuracy (Bai et al., 2025). We retain every example whose tokenized prompt fits the 128K window, yielding 304 examples for Qwen3-8B and 321 for Llama 3 8B Instruct.

Baselines. The comparison includes seven training-free baselines. Dynamic NTK-aware RoPE, YaRN, and MrROPE modify rotation frequencies (Peng et al., 2024; Tian et al., 2026); SelfExtend, LaMPE, DCA, and AdaGroPE remap positions through grouping, chunking, or adaptive reuse (Jin et al., 2024; Zhang et al., 2026; An et al., 2024; Xu et al., 2025). All comparisons use a 128K target window with model-specific extension settings.

Implementation details. All methods use the same model weights, tokenizer, prompts, and evaluation examples for each base model. Inference uses `bfloat16` on one NVIDIA A800 GPU with seed 42 and greedy decoding. FARoPE computes pair statistics and fits tracking ratios using 24 PG-19 training books, with four sequences per length from 4K to 128K and seed 42 (Rae et al., 2020). Phase concentration uses only native-length sequences. We jointly tune the three classification hyperparameters on ProofPile, selecting $\tau_C = 0.96$, MAD multiplier 3, and sink floor 0.10 by grid search for both models. We hold these thresholds fixed for RULER, Infinite-Bench, and LongBench v2. Masks and tracking ratios remain fixed during inference. Appendix A.1 gives the calibration settings, and Appendix D.2 summarizes the calibration token counts.

Comparison protocol. Within each model, all methods are evaluated on the same benchmark examples using a common evaluation protocol.

Table 1: Results with a 128K target context window. RULER averages 13 tasks at each length; its Avg. column averages the six lengths. Infinite-Bench averages 12 tasks, and LongBench v2 reports overall multiple-choice accuracy. All scores are percentages; **bold** marks the best reported score per model and column.

Method	RULER							Infinite-Bench	LongBench v2
	Avg.	4K	8K	16K	32K	64K	128K	Avg.	Avg.
Qwen3-8B									
dynamic NTK	93.6	97.5	96.6	96.8	96.9	93.2	80.6	24.4	34.5
YaRN	85.5	90.8	84.6	82.5	87.9	88.2	78.8	23.1	32.9
MrROPE	86.4	87.8	84.2	83.8	89.1	93.6	79.7	23.3	32.9
SelfExtend	90.8	95.6	96.2	89.5	90.8	93.9	79.1	23.1	30.6
LaMPE	91.8	97.8	97.0	92.1	92.4	88.5	83.2	23.5	29.9
DCA	87.5	94.8	93.4	88.1	87.6	84.0	76.9	21.7	26.6
AdaGroPE	84.1	92.5	90.8	84.9	84.1	79.5	72.8	20.4	25.0
FARoPE	**95.2**	**98.1**	**97.6**	**97.2**	**97.4**	**95.4**	**85.5**	**25.3**	**35.2**
Llama 3 8B Instruct									
dynamic NTK	52.7	84.2	71.0	81.2	53.2	25.5	1.4	10.7	12.5
YaRN	60.7	79.6	79.8	72.9	55.2	46.5	30.3	20.7	17.1
MrROPE	66.9	76.8	82.7	71.8	68.2	56.2	46.0	20.3	17.1
SelfExtend	73.9	80.8	78.8	76.8	77.7	**75.1**	53.8	30.1	16.5
LaMPE	75.9	87.9	83.2	79.4	78.7	67.8	58.2	25.4	15.9
DCA	69.1	82.6	77.9	72.8	71.5	61.0	48.7	22.6	14.0
AdaGroPE	64.0	78.4	73.2	68.1	66.7	55.4	42.1	20.8	12.8
FARoPE	**79.5**	**89.2**	**86.1**	**82.8**	**81.0**	74.5	**63.4**	**31.2**	**18.1**

4.2 Main Results

Table 1 reports FARoPE and baseline scores; Appendix D.1 gives the result configuration. Qwen3-8B reaches a RULER average of 95.2 and 85.5 at 128K, compared with $93.6/80.6$ for dynamic NTK and $91.8/83.2$ for LaMPE. Under the $16\times$ Llama 3 extension, its corresponding scores are 79.5 and 63.4. The six-length RULER average includes native and extrapolated lengths.

The Infinite-Bench scores are 25.3 and 31.2 for Qwen and Llama, with margins of 0.9 and 1.1 points over the strongest reported baselines. LongBench v2 scores are 35.2 and 18.1, compared with 34.5 and 17.1 for the strongest reported baselines. These small score differences alone do not establish statistical significance. The Llama score is also below the nominal 25% uniform-choice reference, so its relative improvement does not establish strong long-context reasoning.

Appendix C presents the module ablations and comparisons with alternative frequency schedules.

Evaluation scope. At $L \leq L_0$, the gain is one, but the target-factor frequency transformation remains active when $f > 1$. A unit gain therefore does not recover the unmodified Base model. Likewise, compatibility with standard attention does not establish equal latency or memory use.

4.3 Sensitivity to the Calibration Corpus

We compare PG-19 training books with the English C4 training split (Raffel et al., 2020), using matched calibration token budgets and length distributions with fixed hyperparameters. Each configuration is recalibrated once with seed 42. A PG-19 packing control separates sequence construction from corpus choice. Appendix D.3 gives the full protocol and results.

Table 4 reports Macro-Jaccard, the mean overlap of the three pair classes, of 0.963–0.983; mean absolute changes in $\hat{\alpha}$ are 0.012–0.026. Semantic pairs are most stable, and even the lowest sink overlap is 0.945. All absolute downstream changes are below one percentage point (Table 5). Switching to C4 slightly lowers every point estimate, but all paired 95% bootstrap intervals include zero. These

results show limited observed sensitivity in the evaluated settings without establishing performance equivalence.

4.4 Analysis

Positional attention. Figure 4 compares last-query attention for a 32K-token input to Qwen3-8B. The largest attention weights under Base, YaRN, and FARoPE occur at positions 32,766, 10,650, and 32,767, with weights 4.45%, 4.55%, and 4.20%. FARoPE retains a peak next to Base's, consistent with the shared distance remapping in Section 3.3. This single-head example does not establish retrieval accuracy or preservation of the full attention distribution.

Sink calibration. Figure 5 reports first-token attention mass for eight query heads in layer 16 of Qwen3-8B. Base averages 8K–32K inputs; YaRN and FARoPE average 8K–128K inputs. Mean mass across heads is 13.78%, 29.04%, and 13.71%, respectively. FARoPE assigns less sink mass than YaRN in every displayed head while retaining nonzero sink attention. Its mean is close to the native-length Base reference. However, Base uses a shorter averaging range, and raw mass depends on the number of competing tokens. Proximity to Base alone therefore does not establish calibration quality for the count-normalized preference targeted by the fit (Section 3.4).

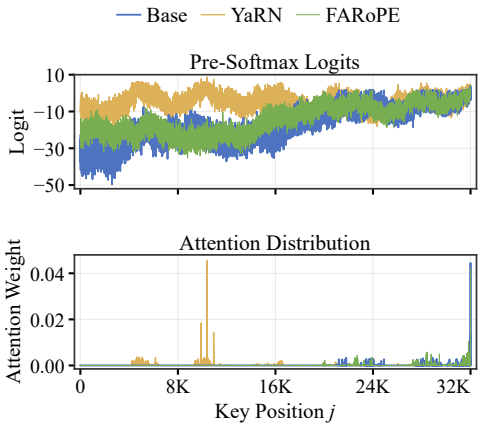

Figure 4: Last-query logits (top) and attention weights (bottom) for Qwen3-8B on a 32K-token input, at layer 0 and query head 28. YaRN and FARoPE use the 4× target extension setting. FARoPE's largest attention weight remains near the query position, close to the Base peak.

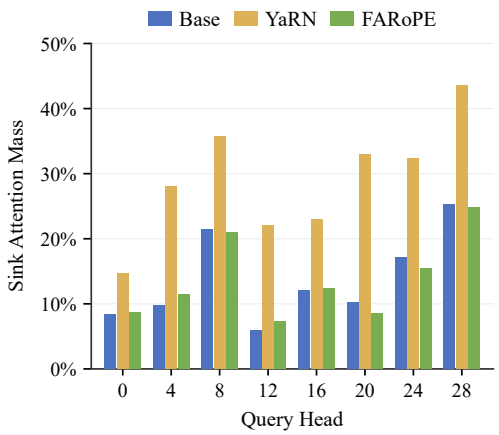

Figure 5: Attention mass on the first non-padding token for eight query heads in layer 16 of Qwen3-8B. Base averages 8K–32K inputs; YaRN and FARoPE average 8K–128K inputs. Means across heads are 13.78%, 29.04%, and 13.71% for Base, YaRN, and FARoPE, respectively.

5 Conclusion

FARoPE selects RoPE interventions from the measured behavior of dimension pairs. Shared frequency scaling gives positional contributions a common distance remapping and limits sink-pair rotation, while separate gain calibration controls additional sink amplification. FARoPE has higher average scores than the reported baselines on all three benchmarks, with small margins on InfiniteBench and LongBench v2. These aggregate scores measure downstream performance but do not isolate the role of pair classes or establish transfer beyond the evaluated settings.

AI USE STATEMENT

We used generative AI tools to assist with experiment-design discussion, coding, and manuscript translation. The authors are responsible for verifying AI-assisted content and for the final manuscript, code, and scientific claims.

ETHICS STATEMENT

This work studies context-window extension using publicly available pretrained language models and established datasets and benchmarks. It involves no new personal-data collection or human-subject experiments. Extending context length does not remove the biases, privacy risks, or potential harmful outputs of the underlying models.

REPRODUCIBILITY STATEMENT

An anonymized implementation of FARoPE is provided in the supplementary material. Section 3 describes the method, and Section 4.1 specifies the evaluation setup. Appendices A and D provide the calibration procedure and additional configuration details.

REFERENCES

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 4895–4901, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.298.

Chenxin An, Fei Huang, Jun Zhang, Shansan Gong, Xipeng Qiu, Chang Zhou, and Lingpeng Kong. Training-Free Long-Context Scaling of Large Language Models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 1493–1510. PMLR, 2024.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3639–3664, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.183.

Federico Barbero, Álvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Michael Bronstein, Petar Veličković, and Razvan Pascanu. Why do LLMs attend to the first token? In *Second Conference on Language Modeling*, 2025a.

Federico Barbero, Alex Vitvitskyi, Christos Perivolaropoulos, Razvan Pascanu, and Petar Veličković. Round and Round We Go! What makes Rotary Positional Encodings useful? In *International Conference on Learning Representations*, pp. 92408–92438, 2025b.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending Context Window of Large Language Models via Positional Interpolation. *arXiv preprint arXiv:2306.15595*, 2023.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. LongLoRA: Efficient Fine-tuning of Long-Context Large Language Models. In *International Conference on Learning Representations*, pp. 8220–8238, 2024.

Yuan Cheng, Fengzhuo Zhang, Yunlong Hou, Cunxiao Du, Chao Du, Tianyu Pang, Aixin Sun, and Zhuoran Yang. Demystifying the Slash Pattern in Attention: The Role of RoPE. *arXiv preprint arXiv:2601.08297*, 2026.

Tri Dao. FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning. In *International Conference on Learning Representations*, pp. 35549–35562, 2024.

Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. LongRoPE: Extending LLM Context Window Beyond 2 Million Tokens. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 11091–11104. PMLR, 2024.

Zican Dong, Junyi Li, Xin Men, Wayne Xin Zhao, Bingning Wang, Zhen Tian, Weipeng Chen, and Ji-Rong Wen. Exploring Context Window of Large Language Models via Decomposed Positional Vectors. In *Advances in Neural Information Processing Systems*, volume 37, pp. 10320–10347. Curran Associates, Inc., 2024. doi: 10.52202/079017-0330.

Zizhuo Fu, Wenxuan Zeng, Runsheng Wang, and Meng Li. Attention Sink Forges Native MoE in Attention Layers: Sink-Aware Training to Address Head Collapse. *arXiv preprint arXiv:2602.01203*, 2026.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, et al. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*, 2024.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When Attention Sink Emerges in Language Models: An Empirical View. In *International Conference on Learning Representations*, pp. 97114–97144, 2025.

Chi Han, Qifan Wang, Hao Peng, Wenhan Xiong, Yu Chen, Heng Ji, and Sinong Wang. LM-Infinite: Zero-Shot Extreme Length Generalization for Large Language Models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3991–4008, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.222.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the Real Context Size of Your Long-Context Language Models? In *First Conference on Language Modeling*, 2024.

Hongye Jin, Xiaotian Han, Jingfeng Yang, Zhimeng Jiang, Zirui Liu, Chia-Yuan Chang, Huiyuan Chen, and Xia Hu. LLM Maybe LongLM: SelfExtend LLM Context Window Without Tuning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 22099–22114. PMLR, 2024.

Siquan Li, Kaiqi Jiang, Jiacheng Sun, and Tianyang Hu. The Structural Origin of Attention Sink: Variance Discrepancy, Super Neurons, and Dimension Disparity. *arXiv preprint arXiv:2605.06611*, 2026.

Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, 2004. Association for Computational Linguistics.

Xiaoran Liu, Hang Yan, Chenxin An, Xipeng Qiu, and Dahua Lin. Scaling Laws of RoPE-based Extrapolation. In *International Conference on Learning Representations*, pp. 48160–48185, 2024.

Yui Oka, Kentaro Hanafusa, Taku Hasegawa, Kyosuke Nishida, and Kuniko Saito. Probing Rotary Position Embeddings through Frequency Entropy. In *International Conference on Learning Representations*, pp. 6139–6167, 2026a.

Yui Oka, Itsumi Saito, Kyosuke Nishida, and Kuniko Saito. Frequency Bands in RoPE: Base Frequency and Context Length Shape the Interpolation–Extrapolation Trade-off. In *International Conference on Learning Representations*, pp. 94947–94975, 2026b.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient Context Window Extension of Large Language Models. In *International Conference on Learning Representations*, pp. 31932–31951, 2024.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. Compressive Transformers for Long-Range Sequence Modelling. In *International Conference on Learning Representations*, 2020.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Valeria Ruscio, Umberto Nanni, and Fabrizio Silvestri. What are you sinking? A geometric approach on attention sink. In *Advances in Neural Information Processing Systems*, volume 38, pp. 123482–123521. Curran Associates, Inc., 2025. doi: 10.52202/085713-4116.

Ning Shang, Li Lyna Zhang, Siyuan Wang, Gaokai Zhang, Gilsinia Lopez, Fan Yang, Weizhu Chen, and Mao Yang. LongRoPE2: Near-Lossless LLM Context Window Scaling. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 54203–54218. PMLR, 2025.

Jianlin Su, Ahmed Murtadha, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced Transformer with Rotary Position Embedding. *Neurocomputing*, 568:127063, 2024. doi: 10.1016/j.neucom.2023.127063.

Zunhai Su, Hengyuan Zhang, Wei Wu, Yifan Zhang, Yaxiu Liu, He Xiao, Qingyao Yang, Yuxuan Sun, Rui Yang, Chao Zhang, Jing Xiong, Hui Shen, Keyu Fan, Weihao Ye, Chaofan Tao, Taiqiang Wu, Zhongwei Wan, Tiantian Zhang, Bowen Yan, Zhen Li, Yiming Zhang, Congkai Xie, Yulei Qian, Yuchen Xie, Yik-Chung Wu, Hongxia Yang, and Ngai Wong. Attention Sink in Transformers: A Survey on Utilization, Interpretation, and Mitigation. *arXiv preprint arXiv:2604.10098*, 2026.

Qingyuan Tian, Wenhong Zhu, Xiaoran Liu, Xiaofeng Wang, and Rui Wang. MrRoPE: Mixed-radix Rotary Position Embedding. In *International Conference on Learning Representations*, pp. 64365–64380, 2026.

Shaowen Wang, Yuke Zheng, Tansheng Zhu, Shuang Chen, Shaofan Liu, Suncong Zheng, and Jian Li. AdaRoPE: Not All Attention Heads Should Rotate and Scale Equally. *arXiv preprint arXiv:2607.19363*, 2026.

Davis Wertheimer, Aozhong Zhang, Derrick Liu, Penghang Yin, and Naigang Wang. Frayed RoPE and Long Inputs: A Geometric Perspective. In *International Conference on Learning Representations*, pp. 156004–156035, 2026.

Jeffrey T. H. Wong, Cheng Zhang, Louis Mahon, Wayne Luk, Anton Isopoussu, and Yiren Zhao. On the Existence and Behavior of Secondary Attention Sinks. *arXiv preprint arXiv:2512.22213*, 2025.

Yingsheng Wu, Yuxuan Gu, Xiaocheng Feng, Weihong Zhong, Dongliang Xu, Qing Yang, Hongtao Liu, and Bing Qin. Extending Context Window of Large Language Models from a Distributional Perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7288–7301, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.414.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient Streaming Language Models with Attention Sinks. In *International Conference on Learning Representations*, pp. 21875–21895, 2024.

Xinhao Xu, Jiaxin Li, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Extending LLM Context Window with Adaptive Grouped Positional Encoding: A Training-Free Method. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 573–587, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.28.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, 2025a.

Bowen Yang, Bharat Venkitesh, Dwaraknath Gnaneshwar Talupuru, Hangyu Lin, David Cairuz, Phil Blunsom, and Acyr Locatelli. Rope to Nope and Back Again: A New Hybrid Attention Strategy. In *Advances in Neural Information Processing Systems*, volume 38, pp. 64133–64157. Curran Associates, Inc., 2025b. doi: 10.52202/085713-2150.

Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. Unveiling and Harnessing Hidden Attention Sinks: Enhancing Large Language Models without Training through Attention Calibration. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57659–57677. PMLR, 2024.

Sikui Zhang, Guangze Gao, Ziyun Gan, Chunfeng Yuan, Zefeng Lin, Houwen Peng, Bing Li, and Weiming Hu. LaMPE: Length-aware Multi-grained Positional Encoding for Adaptive Long-context Scaling Without Training. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 32134–32149, San Diego, California, United States, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.1608.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. ∞Bench: Extending Long Context Evaluation Beyond 100K Tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15262–15277, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.814.

A Calibration and Pair Classification

This appendix gives the information needed to reproduce the offline calibration used by FARoPE. We first describe the data and pair-classification rule, then specify the fit of the sink tracking ratio and grouped-query aggregation. Pair classes and fitted gains remain fixed during inference.

A.1 Offline Calibration Protocol

Positions and data. For a calibration sequence, let j_0 be its first non-padding position. Every valid query i uses only this position as its candidate sink: $\mathcal{S}_i = \{j_0\}$. The ordinary context $\mathcal{C}_i$ contains every non-padding position strictly between j_0 and i. We neither subsample this set nor remove positions near the query. Queries with an empty ordinary context are omitted from statistics that compare the two sets. Delimiters and dynamically detected secondary sinks are not included, and the same position sets are used for both pair classification and sink-gain calibration.

We use the PG-19 training split (Rae et al., 2020) with seed 42. The corpus contains 24 distinct books, with four sequences at each of 4K, 8K, 16K, 32K, 64K, and 128K. This yields 24 sequences and 1,032,192 non-padding calibration tokens per model. Both models use the same books, with sequence lengths measured using each model's tokenizer. Phase concentration uses only lengths within the native context window L_0. Sink bias and sink-gain calibration use all six lengths, with equal total weight assigned to each length so that the longest sequences do not dominate the estimates.

This is the default calibration corpus. The corpus-sensitivity comparison in Appendix D.3 keeps this token budget and length grid while varying the source corpus and sequence construction.

Pair classification. For each layer-head, we summarize the sink biases $\{B_{\ell,h,p}\}_p$ by their median $\widetilde{B}_{\ell,h}$ and raw median absolute deviation (MAD). The MAD is the median of $|B_{\ell,h,p} - \widetilde{B}_{\ell,h}|$ across dimension pairs, so it provides a robust within-head measure of dispersion. The sink threshold is

$$\tau^B_{\ell,h} = \max\Big\{0.10,\ \widetilde{B}_{\ell,h} + 3\,\mathrm{MAD}_{\ell,h}\Big\}\,. \tag{11}$$

The floor 0.10 is measured in the same partial-logit units as the sink bias and prevents a negligible positive bias from becoming a sink criterion when within-head dispersion is small. We use the raw MAD, without the 1.4826 conversion to a Gaussian standard-deviation estimate.

Classification then follows one sink-first rule. A pair is a sink pair when $B_{\ell,h,p} > \tau^{B}_{\ell,h}$. Among the remaining pairs, it is positional when its phase concentration satisfies $C_{\ell,h,p} \geq 0.96$, and semantic otherwise. Thus, 0.96 is the phase-concentration threshold, not a sink threshold. The three classes are mutually exclusive and determine the frequency and gain transformations applied to each pair. Semantic pairs have dispersed pre-RoPE phases, and their partial QK products vary primarily with input content, with a weaker dependence on relative position. These content-dependent interactions allow them to carry semantic information about tokens and their context.

A pair with both concentrated phases and a large sink bias receives the sink intervention, combining frequency scaling by $1/f$ with separate gain calibration.

A.2 Fitting the Sink Tracking Ratio

The fit targets excess sink preference rather than the total amount of sink attention learned within the native window. For any set of logits z, we measure the count-normalized sink preference of query i as

$$\kappa_i(z) = \log\left(\frac{1}{|\mathcal{S}_i|}\sum_{s\in\mathcal{S}_i} e^{z_{is}}\right) - \log\left(\frac{1}{|\mathcal{C}_i|}\sum_{c\in\mathcal{C}_i} e^{z_{ic}}\right). \tag{12}$$

The corresponding attention-mass ratio is $\sum_{s\in\mathcal{S}_i} p_{is} / \sum_{c\in\mathcal{C}_i} p_{ic} = e^{\kappa_i}|\mathcal{S}_i|/|\mathcal{C}_i|$. Thus a fixed count-normalized preference does not imply fixed sink mass as the context grows. For each layer-head, the native reference $\tau^{\mathrm{ref}}_{\ell,h}$ is the weighted empirical 95th percentile of κ_i under native RoPE over lengths no greater than L_0. Each native length receives equal total weight, distributed uniformly over its eligible calibration rows. We normalize these weights to sum to one, pool the rows across native lengths, and sort their κ values in ascending order. The reference is the first observed value whose cumulative weight reaches or exceeds 0.95, without interpolation. This reference records how much sink preference already exists without extrapolation; the fit corrects only the amount above it.

After fixing the pair classes and their rotation frequencies, let z^{pos}, z^{sem}, and z^{sink} denote the corresponding sums of pair partial logits. For a candidate $\alpha \in [0,1]$, the calibrated logit is

$$\widetilde{z}_{ij,\ell,h}(\alpha, L) = \gamma(L)\big(z^{\mathrm{pos}}_{ij,\ell,h} + z^{\mathrm{sem}}_{ij,\ell,h}\big) + \big[1 + \alpha(\gamma(L)-1)\big] z^{\mathrm{sink}}_{ij,\ell,h}. \tag{13}$$

Uniform YaRN attention-logit gain under the final pair frequencies is the special case $\alpha = 1$. Let $\kappa_{i,\ell,h}(\alpha, L)$ be Equation 12 evaluated with these logits, and let $\kappa^{\mathrm{Y}}_{i,\ell,h}(L) = \kappa_{i,\ell,h}(1, L)$. We define the excess target $R^{+}_{i,\ell,h}(L) = [\kappa^{\mathrm{Y}}_{i,\ell,h}(L) - \tau^{\mathrm{ref}}_{\ell,h}]_{+}$. The local sensitivity $\xi_{i,\ell,h}(L)$ is the positive part of $\partial\kappa_{i,\ell,h}(\alpha, L)/\partial\alpha$ at $\alpha = 1$. When that derivative is non-positive, we set $\xi = 0$, because reducing α would not move sink preference in the desired direction.

The first-order surrogate used for fitting is

$$\kappa_{i,\ell,h}(\alpha, L) \approx \kappa^{\mathrm{Y}}_{i,\ell,h}(L) - (1-\alpha)\xi_{i,\ell,h}(L). \tag{14}$$

This is a Taylor approximation for positive derivatives. When the derivative is non-positive, clipping to $\xi = 0$ makes the row constant in the surrogate. We fit one ratio per layer-head by

$$\begin{aligned} \widehat{\alpha}_{\ell,h} &= \operatorname*{argmin}_{0\leq\alpha\leq 1} \mathcal{J}_{\ell,h}(\alpha), \\ \mathcal{J}_{\ell,h}(\alpha) &= \sum_{L,i} w_{i,L}\left[(1-\alpha)\xi_{i,\ell,h}(L) - R^{+}_{i,\ell,h}(L)\right]^2 + 10^{-3}(1-\alpha)^2. \end{aligned} \tag{15}$$

For each fit, let N_L be the number of eligible calibration rows at length L, pooled across its sequences. For GQA, a row includes both the query position and its query-head index within the KV group. We set $w_{i,L} = 1/(6N_L)$ over the six calibration lengths, so each length has total weight $1/6$ and $\sum_{L,i} w_{i,L} = 1$. The data term therefore averages within each length and then across lengths. The final term is the complete regularizer, including its coefficient. It returns an under-determined fit toward $\alpha = 1$, which is the unmodified YaRN gain, without changing the excess-preference target. This objective also makes clear that $\widehat{\alpha}_{\ell,h}$ is a fitted tracking ratio rather than another pair-classification threshold.

The exact derivative is available without finite differences. Let $P^{(1)}_{\mathcal{S}}$ and $P^{(1)}_{\mathcal{C}}$ be the separately normalized softmax distributions under the final frequencies at $\alpha = 1$. Then

$$\left.\frac{\partial\kappa_i}{\partial\alpha}\right|_{\alpha=1} = (\gamma(L)-1)\left(\mathbb{E}_{s\sim P^{(1)}_{\mathcal{S}}} z^{\mathrm{sink}}_{is} - \mathbb{E}_{c\sim P^{(1)}_{\mathcal{C}}} z^{\mathrm{sink}}_{ic}\right). \tag{16}$$

With $\lambda = 10^{-3}$, the constrained quadratic has the closed form

$$\widehat{\alpha}_{\ell,h} = \mathrm{clip}_{[0,1]}\left(1 - \frac{\sum_{L,i} w_{i,L}\xi_{i,\ell,h}(L)R^{+}_{i,\ell,h}(L)}{\lambda + \sum_{L,i} w_{i,L}\xi_{i,\ell,h}(L)^2}\right). \tag{17}$$

An empty sink class or zero sensitivity yields $\widehat{\alpha} = 1$. At native lengths $\gamma = 1$, sensitivity is zero. Eligible rows with zero sensitivity retain their weights in the six-length average. The coefficient $\lambda = 10^{-3}$ applies to this unit-total-weight objective.

The closed form minimizes the surrogate rather than the nonlinear sink preference. The multiplier lies between one and γ, so calibration reduces only the added gain; it neither masks sink tokens nor guarantees a particular attention mass or improvement for every query.

A.3 Grouped-Query Attention and Numerical Conventions

Let $\mathcal{G}_h$ denote the query heads that share KV head h. For query head $r \in \mathcal{G}_h$, compute its own magnitude-weighted concentration $C^q_{\ell,r,p}$, then use

$$C_{\ell,h,p} = \frac{1}{2}\left(\frac{1}{|\mathcal{G}_h|}\sum_{r\in\mathcal{G}_h} C^q_{\ell,r,p} + C^k_{\ell,h,p}\right). \tag{18}$$

A pair with zero total activation magnitude receives concentration zero. Each query- or key-side concentration is at most one. Therefore, the threshold $C \geq 0.96$ requires both the group-averaged query concentration and key concentration to be at least 0.92.

Concentration is computed before averaging across query heads, avoiding cancellation between distinct head directions. Sink bias likewise averages the signed per-query-head margins after computing each head's within-set softmax. Averaging logits before softmax defines a different statistic. For the reference threshold, we pool the native-length κ rows from all query heads in the KV group with equal query-head weight and apply the weighted quantile rule in Appendix A.2. We do not average separate per-head quantiles. The gain fit likewise pools the per-head objective rows to estimate one ratio for the group, rather than averaging separately fitted ratios. All query heads use the same eligible query positions, so counting the head index in N_L gives equal head weight within each length without another division by $|\mathcal{G}_h|$. The group shares its pair assignments, reference threshold, and fitted ratio. Query-head identifiers in figures refer to r, while masks and ratios are indexed by h.

Accumulate phase sums and softmax/log-mean-exp statistics in fp32 or higher precision, using numerically stable log-sum-exp and explicit causal/padding masks. Empty comparison sets are excluded rather than assigned zero preference. The reference code uses $\epsilon = 10^{-8}$ for numerical denominators. Pair-logit diagnostics require blockwise extraction at long lengths; the model-agnostic diagnostic tensor is not evidence of efficient 128K inference.

B A Local Dynamical View of Attention-Sink Formation

We next revisit the mechanism that motivates sink-specific calibration. The analysis describes one local and sufficient optimization pathway, not a universal account of attention sinks. It does not require a sink token to be semantically empty, and it does not imply that the first token becomes a sink in every head. During each infinitesimal step, we hold head inputs fixed and analyze direct gradient flow through W_V, W_Q, and W_K.

B.1 A Local Feedback Mechanism

Consider one causal attention head. For query m, its output is

$$o_m = \sum_{n\leq m} p_{mn} v_n, \qquad p_{mn} = \frac{\exp(a_{mn})}{\sum_{r\leq m}\exp(a_{mr})}, \tag{19}$$

where a_{mn} is the attention logit and v_n includes a locally fixed output projection. We focus on query positions for which the residual path already carries most of the needed representation and this head

is locally encouraged to make only a small update. Around $o_m = 0$, we express this no-op regime as $\mathcal{L}_m(o_m) = \mathcal{L}_m(0) + \lambda_m \|o_m\|^2/2 + o(\|o_m\|^2)$, with $\lambda_m > 0$. Hence the local output gradient is $g_m = \lambda_m o_m + o(\|o_m\|)$.

Let x_n be the input to the value projection, $v_n = W_V x_n$, and $\overline{x}_m = \sum_{n \le m} p_{mn} x_n$. Because the attention probabilities do not depend on W_V, gradient flow gives

$$\begin{aligned} \dot{v}_s &= -\eta \sum_m g_m \langle \overline{x}_m, x_s \rangle, \\ \dot{v}_s &= -c_s(t) v_s + e_s(t), \\ c_s(t) &= \eta \|x_s\|^2 \sum_{m \in \mathcal{M}_s} \lambda_m p_{ms}^2 > 0. \end{aligned} \tag{20}$$

The first line retains the cross-token Gram terms, which disappear if v_s is incorrectly treated as an independent parameter. $\mathcal{M}_s$ contains queries for which candidate position s already receives non-negligible attention, and $e_s(t)$ collects other tokens, queries outside this set, and higher-order terms.

The contraction is conditional. If $\|e_s(t)\| \le \epsilon c_s(t) \|v_s(t)\|$ for some $0 \le \epsilon < 1$, then

$$\frac{d}{dt} \frac{\|v_s\|^2}{2} \le -(1-\epsilon) c_s(t) \|v_s\|^2. \tag{21}$$

Thus, the squared sink-value norm is bounded by its initial value times $\exp[-2(1-\epsilon)\int_0^t c_s(u)\,du]$. Exponential decay in wall time additionally requires a positive lower bound on the time-averaged rate. Convergence to zero requires $\int_0^\infty c_s(t)\,dt = \infty$. These explicit conditions avoid the stronger and generally unjustified claim that every activated sink value must unconditionally vanish.

B.2 From Value Contraction to Logit Growth

The softmax derivative links the value-side dynamics to the sink logit. Define the negative-gradient drive $\Delta_{mj} = -\partial \mathcal{L}/\partial a_{mj}$. In the local no-op regime,

$$\Delta_{ms} = p_{ms} \langle g_m, o_m - v_s \rangle \approx \lambda_m p_{ms} \big(\|o_m\|^2 - \langle o_m, v_s \rangle \big). \tag{22}$$

The direct drive is positive when $\langle o_m, v_s \rangle < \|o_m\|^2$, a condition that becomes easier to satisfy as v_s contracts. If logits were independent parameters, this would be enough for gradient descent to increase a_{ms}.

Transformer logits are not independent because they share W_Q and W_K. Let $\vartheta = (W_Q, W_K)$ and define the logit tangent kernel $K_{(m,s),(i,j)} = \langle \nabla_\vartheta a_{ms}, \nabla_\vartheta a_{ij} \rangle$. The exact first-order parameter-flow relation and a sufficient condition for positive sink-logit growth are

$$\begin{aligned} \dot{a}_{ms} &= \eta \sum_{(i,j) \in \mathcal{A}} \Delta_{ij} K_{(m,s),(i,j)}, \\ \Delta_{ms} K_{(m,s),(m,s)} &> \left| \sum_{(i,j) \ne (m,s)} \Delta_{ij} K_{(m,s),(i,j)} \right|. \end{aligned} \tag{23}$$

The second line states the needed diagonal-dominance condition explicitly. Without sufficient alignment or dominance, cross terms from the shared projections can reverse the direct drive, so monotonic logit growth is not guaranteed.

Under this condition, value contraction and logit growth form a positive feedback loop. A smaller v_s makes attention to s a lower-cost way to keep the head output small. The resulting positive drive increases p_{ms}, and the larger attention probability strengthens both the contraction rate, which contains p_{ms}^2, and the logit drive, which contains p_{ms}. This loop explains how a weak initial preference can be amplified. It does not identify which token breaks symmetry first; causal visibility, token frequency, learned geometry, and the rest of the network can all affect that selection.

B.3 RoPE Coherence and the Calibration Rule

RoPE affects whether sink-directed gradients from different query positions reinforce or cancel within a dimension pair. The pair contribution contains the rotation factor $e^{\mathrm{i}(m-s)\omega_p}$. For non-negative query weights w_m that sum to one, define coherence over the sink domain as

$$\rho_{s,p} = \left| \sum_{m \in \mathcal{M}_s} w_m e^{\mathrm{i}(m-s)\omega_p} \right|, \qquad \rho_{s,p}^2 = 1 - \omega_p^2 \operatorname{Var}(m-s) + \mathcal{O}(\omega_p^4 \mu_4). \tag{24}$$

where the expansion holds for small ω_p, and μ_4 is the fourth moment of the difference between two independently drawn offsets. The coherence isolates cancellation caused by RoPE rotations on the observed offsets; activation magnitudes and pre-RoPE directions can still cancel independently.

When those activation-side factors remain aligned, slow-rotating pairs retain coherent gradient accumulation over a broader range of offsets. Faster pairs, or pairs whose observed phases span many cycles, can cancel more strongly. This provides a conditional explanation for the empirical concentration of sink behavior in slowly rotating dimensions, but frequency rank alone is not a sufficient sink classifier.

The first non-padding token is causally visible to every later query, so it has repeated opportunities to participate in the feedback loop. The same argument can describe a localized secondary sink, but the calibration protocol intentionally keeps $\mathcal{S}_i = \{j_0\}$ and does not add dynamic sink candidates.

The theory motivates the intervention without determining the pair classes. FARoPE measures a pair's sink contribution directly through $B_{\ell,h,p}$. It then replaces the uniform sink multiplier $\gamma(L)$ by $1 + \widehat{\alpha}_{\ell,h}(\gamma(L) - 1)$. Relative to YaRN, this removes the fraction $1 - \widehat{\alpha}_{\ell,h}$ of the additional extrapolation-induced gain while preserving the base multiplier of one. The method therefore limits excess long-context amplification without masking the sink token or declaring its learned function useless.

C Ablation Study

We examine the contributions of Component-Specific Frequency Scaling (Section 3.3) and Component-Specific Logit Calibration (Section 3.4), then compare alternative frequency schedules for semantic pairs.

C.1 Configurations and Evaluation

Table 2 defines the six variants. Removing the frequency module restores YaRN frequencies for all pairs; removing logit calibration applies the uniform gain $\gamma(L)$. Removing both combines YaRN frequencies with FARoPE's length-dependent gain $\gamma(L)$, evaluated separately from the YaRN baseline in Table 1. The last two variants replace only the semantic-pair frequency schedule with dynamic NTK-aware RoPE or MrROPE; positional and sink pairs retain ω_p/f.

Table 2: Frequency and sink-gain settings for the six ablation variants. "Fitted" denotes $1+\hat{\alpha}[\gamma(L)-1]$, fitted under each variant's final frequencies. Non-sink logits use $\gamma(L)$ throughout.

Variant	Positional frequency	Semantic frequency	Sink frequency	Sink gain
FARoPE	ω_p/f	ω_p^{Y}	ω_p/f	Fitted
w/o frequency module	ω_p^{Y}	ω_p^{Y}	ω_p^{Y}	Fitted
w/o logit calibration	ω_p/f	ω_p^{Y}	ω_p/f	$\gamma(L)$
w/o both modules	ω_p^{Y}	ω_p^{Y}	ω_p^{Y}	$\gamma(L)$
FARoPE + Dynamic NTK	ω_p/f	ω_p^{N}	ω_p/f	Fitted
FARoPE + MrROPE	ω_p/f	ω_p^{M}	ω_p/f	Fitted

Here ω_p^{Y}, ω_p^{N}, and ω_p^{M} denote the YaRN, dynamic NTK-aware RoPE, and MrROPE frequencies, respectively. All schedules use the 128K target window and remain fixed within a request, including

the dynamic NTK schedule in the combination variant. Holding $\gamma(L)$ fixed across the two combination variants isolates the semantic frequency schedule; these variants differ from the standalone baselines in Table 1.

The comparisons use the same calibration data, pair classes, and evaluation examples. When a variant retains logit calibration, we recompute its fitting statistics under the final frequencies and refit one $\hat{\alpha}$ per layer–KV-head group using Appendix A.2. Calibration documents, length weights, and fitting hyperparameters remain fixed. The reference logits for Sink Bias and the resulting pair classes also remain fixed; only the statistics used for gain fitting are recomputed. Uniform gain sets $\hat{\alpha} = 1$ and requires no fit. These comparisons assess the transformations applied to fixed classes, rather than alternative pair classification rules.

C.2 Results and Analysis

Table 3: Module ablations and combinations with other frequency-scaling methods. RULER averages six lengths from 4K to 128K; Infinite-Bench and LongBench v2 use the same aggregates as Table 1. All scores are percentages (higher is better); **bold** marks the best score per column. The variant without both modules uses YaRN frequencies with FARoPE's length-dependent gain $\gamma(L)$.

Variant	Qwen3-8B			Llama 3 8B Instruct		
	RULER	Infinite-Bench	LongBench v2	RULER	Infinite-Bench	LongBench v2
FARoPE	**95.2**	**25.3**	**35.2**	**79.5**	**31.2**	**18.1**
w/o frequency module	91.7	23.6	32.6	74.8	26.4	15.9
w/o logit calibration	93.8	24.4	33.9	77.6	29.2	17.1
w/o both modules	86.1	23.4	32.2	62.2	22.7	17.1
FARoPE + Dynamic NTK	94.2	24.7	34.5	76.5	28.4	16.5
FARoPE + MrROPE	92.9	24.1	33.9	78.0	29.6	17.4

Contributions of the two modules. Table 3 shows that full FARoPE has the highest score on each benchmark for both models. Removing frequency scaling lowers RULER by 3.5 points for Qwen and 4.7 points for Llama; removing logit calibration lowers it by 1.4 and 1.9 points, respectively. Both removals also lower Infinite-Bench and LongBench v2 scores. Relative to removing both modules, the full method improves RULER by 9.1 and 17.3 points and Infinite-Bench by 1.9 and 8.5 points for Qwen and Llama, respectively.

The effect of gain calibration depends on both the model and the frequency setting. On LongBench v2, adding calibration to YaRN frequencies raises Qwen's score from 32.2 to 32.6 but lowers Llama's from 17.1 to 15.9. With FARoPE frequencies, it raises both scores, from 33.9 to 35.2 for Qwen and from 17.1 to 18.1 for Llama. These results show that sink-gain calibration does not improve every configuration, while it improves all reported model–benchmark scores when combined with FARoPE frequencies.

Alternative semantic frequency schedules. The default YaRN schedule gives higher scores than the dynamic NTK and MrROPE combinations on all six model–benchmark comparisons. For Qwen, their RULER scores are 94.2 and 92.9, compared with 95.2 for the default schedule; for Llama, they are 76.5 and 78.0, compared with 79.5. These results favor the YaRN semantic-pair schedule among the tested FARoPE configurations, while showing that the relative ordering of the two alternatives depends on the model.

D Configuration and Reproducibility Details

D.1 Result Configuration

All reported full-FARoPE results use the frequency rule in Section 3.3: positional and sink pairs use ω_p/f, while semantic pairs use the YaRN ramp. Sink gains are fitted under these final frequencies. The native and YaRN reference logits used to compute Sink Bias belong to the preceding classifica-

tion stage; they do not specify the final sink frequencies. Ablation variants use the module settings listed in Table 2.

D.2 Reproducibility Details

Appendix A.1 specifies the reported calibration settings. The sequence grid contains 1,032,192 non-padding token positions per model; the native-length subset used for phase concentration contains 245,760 tokens for Qwen and 49,152 for Llama. These counts describe one traversal of the calibration corpus, not repeated forward work.

D.3 Calibration Corpus Sensitivity

Data and controls. The comparison includes three calibration configurations for each model: PG-19 with one contiguous book span per sequence, PG-19 with packed text, and English C4 with packed text. We denote them by PG-19 (single), PG-19 (packed), and C4 (packed), abbreviated as P_s, P_p, and C_p in the tables. All use training data, four sequences at each of 4K, 8K, 16K, 32K, 64K, and 128K, and 1,032,192 non-padding tokens per calibration set. Each configuration uses one calibration set constructed with seed 42. The PG-19 (single) construction follows Appendix A.1, with 24 distinct books per set. Its packed counterpart uses the same book pool.

Packing uses blocks of 1,023 consecutive content tokens followed by one tokenizer EOS token. We sample these spans without reusing token positions within a calibration set and join blocks from different source documents at each boundary. A source document may contribute multiple non-overlapping blocks. C4 documents must contain at least 1,023 content tokens under the model's tokenizer. Both packed corpora therefore have the same block length and delimiter placement. Sequence-level special tokens follow the same model-specific policy in all three configurations and count toward the length budget; only the last block is trimmed if necessary. Position indices continue across blocks, and attention remains causal across document boundaries. Delimiters are ordinary context tokens, and only the first non-padding position is the candidate sink.

The PG-19 (single)–C4 (packed) comparison measures the overall effect of replacing the default corpus. The two PG-19 configurations measure the effect of changing sequence construction within the same book pool. Comparing the packed corpora then tests the source change with block length and boundary placement matched. These controls do not equate the number or diversity of source documents. PG-19 and C4 calibration texts are separate from benchmark examples. Threshold selection follows Section 4.1.

Independent recalibration. Each configuration recomputes phase concentration, Sink Bias, the adaptive sink thresholds, the native sink-preference reference, pair classes, and $\hat{\alpha}$. The fixed hyperparameters remain $\tau_C = 0.96$, MAD multiplier 3, sink floor 0.10, and $\lambda = 10^{-3}$. Phase concentration uses only native lengths; Sink Bias and the gain fit use the six-length grid with equal total weight per length. The gain-fit weight normalization and reference-quantile convention follow Appendix A.2; model revisions and the 128K target window are identical across configurations. After classification, positional and sink pairs use ω_p/f, and semantic pairs use the YaRN ramp. The gain fit uses these final frequencies; neither masks nor fitted ratios are transferred between corpora. Pair classes and fitted gains remain fixed during benchmark inference.

Classification and gain stability. For two calibration outputs A and B of the same model, let $\mathcal{P}_c^A$ be the set of layer–KV-head–pair indices assigned to class $c \in \{\mathrm{pos}, \mathrm{sem}, \mathrm{sink}\}$. We compute

$$J_c(A,B) = \frac{|\mathcal{P}_c^A \cap \mathcal{P}_c^B|}{|\mathcal{P}_c^A \cup \mathcal{P}_c^B|}, \qquad J_{\mathrm{macro}}(A,B) = \frac{1}{|\mathcal{C}_*|} \sum_{c \in \mathcal{C}_*} J_c(A,B), \tag{25}$$

where $\mathcal{C}_*$ contains classes with a non-empty union. A class absent from both outputs is reported as N/A and omitted from the macro average; one-sided absence gives zero overlap. When all three classes are present, each receives weight $1/3$. This statistic measures pair identity rather than merely agreement in class proportions.

For gain stability, let $\mathcal{H}_*$ contain layer–KV-head groups with at least one sink pair in either output. We use

$$D_\alpha(A, B) = \frac{1}{|\mathcal{H}_*|} \sum_{(\ell,h)\in\mathcal{H}_*} |\hat{\alpha}^A_{\ell,h} - \hat{\alpha}^B_{\ell,h}|. \tag{26}$$

An empty sink class in one output retains its default $\hat{\alpha} = 1$. If $\mathcal{H}_*$ is empty, D_α is N/A. Restricting the average to these groups prevents heads with no sink pairs in either output from artificially reducing the difference. Each comparison uses the two calibration outputs constructed with seed 42 and reports one value per metric. Each metric is computed within one model; masks from different models are never compared.

Downstream comparison. All calibrated configurations use the same benchmark examples, prompts, decoding settings, and score aggregation as Section 4.1. For benchmark b, the reported change is $\Delta_b = S_b(B) - S_b(A)$, expressed in percentage points for the fixed calibration outputs. RULER uses the six-length average from 4K to 128K. Positive values favor B. We estimate paired 95% percentile bootstrap intervals with 2,000 replicates and bootstrap seed 42. Each replicate resamples evaluation examples, using the same sampled indices for both configurations. Example resampling is stratified by task and, for RULER, by length; repeated source-example identities are kept together. The original benchmark aggregation is applied within each replicate. These intervals describe evaluation-sample uncertainty conditional on the selected thresholds and fixed calibration outputs; they do not include uncertainty from hyperparameter selection or calibration-data sampling.

Table 4: Calibration-output stability across corpus configurations. PG-19 and C4 give similar pair assignments and fitted sink gains in these comparisons. Jaccard scores closer to 1 and D_α closer to 0 indicate more similar outputs. Each configuration is calibrated once with seed 42. P_s: PG-19 (single); P_p: PG-19 (packed); C_p: C4 (packed).

Comparison $A \to B$	$J_{\text{pos}} \uparrow$	$J_{\text{sem}} \uparrow$	$J_{\text{sink}} \uparrow$	$J_{\text{macro}} \uparrow$	$D_\alpha \downarrow$
Qwen3-8B					
$P_s \to C_p$	0.969	0.987	0.954	0.970	0.021
$P_s \to P_p$	0.982	0.994	0.973	0.983	0.012
$P_p \to C_p$	0.976	0.989	0.960	0.975	0.017
Llama 3 8B Instruct					
$P_s \to C_p$	0.960	0.984	0.945	0.963	0.026
$P_s \to P_p$	0.975	0.991	0.962	0.976	0.016
$P_p \to C_p$	0.966	0.986	0.955	0.969	0.021

Calibration results. Table 4 shows that both changing the corpus and changing the packing rule preserve most pair assignments. Under matched packing ($P_p \to C_p$), Macro-Jaccard is 0.975 for Qwen3-8B and 0.969 for Llama 3 8B Instruct, with $D_\alpha = 0.017$ and 0.021. The PG-19 packing control ($P_s \to P_p$) gives higher overlap and smaller changes in $\hat{\alpha}$ for both models. Thus, the corpus change has a somewhat larger effect than packing alone, while both effects remain small in these measurements. Semantic pairs are the most stable class (Jaccard 0.984–0.994); sink pairs are the least stable (0.945–0.973), but still retain high overlap. The corresponding changes in fitted tracking ratios remain small across all comparisons (D_α ranges from 0.012 to 0.026), so the agreement extends beyond class labels.

Downstream results. Table 5 shows small observed score changes across all three benchmarks. Replacing the default PG-19 configuration with packed C4 ($P_s \to C_p$) lowers scores by 0.20–0.33 percentage points for Qwen3-8B and 0.30–0.62 points for Llama 3 8B Instruct. Packing PG-19 alone changes scores by at most 0.33 points in absolute value. With packing matched ($P_p \to C_p$), C4 gives lower point estimates on every benchmark for both models; the largest drop is 0.93 points on LongBench v2 for Llama 3 8B Instruct. All 18 paired 95% bootstrap intervals include zero. The LongBench v2 intervals are wider than those for the other benchmarks; for example, the largest observed drop has an interval of $[-3.43, 1.56]$ points. The observed scores therefore remain close across corpora, but these intervals do not establish performance equivalence.

Table 5: Downstream sensitivity to calibration data. Each cell gives the score difference $B - A$ in percentage points and its paired 95% bootstrap interval. All absolute point differences are below one percentage point, and every interval includes zero. All configurations use the current FARoPE frequency rule and independently fitted sink gains. RULER, Infinite-Bench, and LongBench v2 retain the main-table aggregation. P_s, P_p, and C_p denote PG-19 (single), PG-19 (packed), and C4 (packed), respectively.

Comparison $A \to B$	RULER	Infinite-Bench	LongBench v2
Qwen3-8B			
$P_s \to C_p$	-0.20 $[-0.48, 0.08]$	-0.20 $[-0.94, 0.55]$	-0.33 $[-2.30, 1.64]$
$P_s \to P_p$	$+0.10$ $[-0.14, 0.34]$	$+0.10$ $[-0.51, 0.72]$	$+0.33$ $[-1.32, 1.97]$
$P_p \to C_p$	-0.30 $[-0.62, 0.02]$	-0.30 $[-1.12, 0.52]$	-0.66 $[-2.96, 1.64]$
Llama 3 8B Instruct			
$P_s \to C_p$	-0.40 $[-0.91, 0.12]$	-0.30 $[-1.18, 0.61]$	-0.62 $[-2.80, 1.56]$
$P_s \to P_p$	-0.10 $[-0.54, 0.35]$	$+0.10$ $[-0.69, 0.88]$	$+0.31$ $[-1.56, 2.18]$
$P_p \to C_p$	-0.30 $[-0.81, 0.21]$	-0.40 $[-1.38, 0.58]$	-0.93 $[-3.43, 1.56]$

E Related Work

E.1 RoPE-Based Context Extension

RoPE-based context extension methods differ in the granularity at which they modify the pretrained positional mechanism. Global methods apply one rule across the RoPE spectrum. Position Interpolation compresses long positions into the pretrained range, dynamic NTK-aware RoPE adjusts the frequency schedule as context length grows, and YaRN combines a frequency-dependent interpolation ramp with a single attention-logit temperature scale (Chen et al., 2023; Peng et al., 2024). These methods offer simple extension rules, but their treatment of a dimension pair is determined by its RoPE frequency rather than its behavior after pretraining.

Non-uniform methods use finer frequency- or dimension-level transformations. Distributional methods estimate rotary-angle shift and choose different interpolation or extrapolation strategies across the spectrum (Wu et al., 2024). LongRoPE searches dimension-wise interpolation factors, while LongRoPE2 attributes residual OOD behavior to under-trained higher-index RoPE dimensions and uses needle-driven perplexity to search their rescaling factors before mixed context window training (Ding et al., 2024; Shang et al., 2025). MrRoPE instead divides RoPE frequencies into bands and applies mixed-radix conversion (Tian et al., 2026).

A separate line remaps relative positions or constrains attention without updating model weights. SelfExtend and DCA reuse or bound relative positions through grouped or chunked attention (Jin et al., 2024; An et al., 2024). AdaGroPE progressively increases the reuse of distant relative positions as input length grows, whereas LaMPE adapts the mapping length to each input and allocates different positional resolutions across sequence regions (Xu et al., 2025; Zhang et al., 2026). LM-Infinite restricts attention to keep inference within the pretrained distance range (Han et al., 2024). Training-based approaches such as LongLoRA instead extend the training distribution through efficient long-context fine-tuning (Chen et al., 2024). FARoPE does not remap token coordinates. It defines frequency groups from observed dimension-pair behavior and separately controls the temperature gain applied to sink-pair contributions.

E.2 RoPE Dimension Analysis

Recent studies show that RoPE components do not play uniform roles. Barbero et al. (2025b) connect different parts of the frequency spectrum to positional and semantic behavior, motivating selective use of RoPE dimensions. Frequency-band analysis further shows that high-norm bands emerge during pretraining and that their locations depend on the RoPE base and training length (Oka et al., 2026b). Frequency Entropy probes individual RoPE frequencies, while slash-pattern analysis links particular frequency components to heads that attend at stable relative offsets (Oka

et al., 2026a; Cheng et al., 2026). Decomposed positional vectors provide another view by separating and analyzing the positional component of hidden representations (Dong et al., 2024).

Other methods study heterogeneity across layers and attention heads. RoPE/NoPE hybrids alternate RoPE and NoPE across layers, and AdaRoPE learns head-specific choices of whether and how strongly to rotate and scale (Yang et al., 2025b; Wang et al., 2026). These studies establish that a uniform positional treatment is restrictive. However, frequency bands, layers, and whole heads remain coarse intervention units when one head contains several kinds of pair-wise behavior. FARoPE instead classifies dimension pairs from pre-RoPE phase concentration and their contributions to sink-token attention logits, then uses these classes directly during extrapolation.

Frayed RoPE also connects long-input failure to query/key geometry and sink functionality, combining channel-selective rotation with input-length temperature scaling (Wertheimer et al., 2026). FARoPE instead selects interventions from measured pair behavior and calibrates the gain only for sink-pair contributions. The settings also differ: RoPE-ID is evaluated in decoders pretrained with its positional design, whereas FARoPE modifies fixed pretrained models.

E.3 Attention Sinks

Attention sinks are tokens that receive large attention mass despite carrying limited semantic information for the current query. StreamingLLM documents this phenomenon and shows that retaining initial sink tokens stabilizes windowed streaming inference (Xiao et al., 2024). Later work identifies hidden sinks beyond the first token and uses attention calibration to control them without weight updates (Yu et al., 2024). Secondary sinks further show that sink behavior can move across positions and depend on the layer or context (Wong et al., 2025).

A growing literature studies why sinks form. Empirical work traces when sinks emerge during training, while first-token and geometric analyses relate the phenomenon to optimization, data, representation geometry, and softmax attention (Gu et al., 2025; Barbero et al., 2025a; Ruscio et al., 2025). Other studies connect sink formation to variance discrepancy, super neurons, dimension disparity, and head collapse (Li et al., 2026; Fu et al., 2026); a recent survey organizes these explanations together with sink utilization and mitigation methods (Su et al., 2026). Most prior methods diagnose or intervene on sink tokens, attention heads, or training dynamics. FARoPE examines a different level: the partial QK products contributed by individual dimension pairs during long-context extrapolation. It retains sink tokens and their learned function, but limits the extra length-dependent temperature gain applied to sink-pair contributions.

F Inference Implementation

After offline calibration, FARoPE stores one class label per pair and one fitted sink tracking ratio per layer–KV-head group. Online inference uses this metadata to set frequencies and gains without weight updates, added tokens, or extra model forward passes.

We apply pair-specific gains to the query vectors, avoiding the need to materialize pair-wise attention matrices. Let $R_{\ell,h,t}$ apply the final pair frequencies at position t, and let $D_{\ell,h}(L)$ be diagonal with the same gain on both coordinates of each pair. Its entry is $\gamma(L)$ for non-sink pairs and $1 + \widehat{\alpha}_{\ell,h}(\gamma(L) - 1)$ for sink pairs. For query head r in KV group h, define

$$\bar{q}_{\ell,r,i} = D_{\ell,h}(L) R_{\ell,h,i} q_{\ell,r,i}, \qquad \bar{k}_{\ell,h,j} = R_{\ell,h,j} k_{\ell,h,j}. \tag{27}$$

The dot product $\bar{q}_{\ell,r,i}^{\top} \bar{k}_{\ell,h,j} / \sqrt{d}$ equals the class-scaled logit in real arithmetic. Standard dot-product attention kernels, including FlashAttention-2 (Dao, 2024), can therefore consume these queries and keys without changing softmax or value aggregation.

Putting the full gain on queries lets $\gamma(L)$ change during decoding without rescaling cached keys. The target frequencies and pair masks remain fixed within each request; changing the frequencies would require consistent re-rotation of cached keys.

The transformations preserve the attention complexity and KV-cache dimensions. The extra query scaling costs $O(LH_q d)$ for prefill and $O(H_q d)$ per decoded token, where H_q is the number of query heads. Offline calibration requires model forward evaluations and stores $O(N_{\text{layers}} H_{\text{KV}} d)$ metadata, independently of sequence length. Actual latency and peak memory depend on the implementation.