FUSED BUT NOT RELATIVE: LOW-RANK ATTENTION BIASES SILENTLY DROP TRANSLATION INVARIANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

A rank-R additive attention bias written as $B \approx \phi_q \phi_k^\top$, with $\phi_q, \phi_k \in \mathbb{R}^{N \times R}$, can be folded into the queries and keys, allowing compatible unmodified FlashAttention kernels to compute biased attention without materializing the $N \times N$ bias. Alternative fused solutions require bias-specific kernel support or compilation. Low-rank factorization supports fused training but does not guarantee relativity: $\phi_q(i)^\top \phi_k(j)$ may depend on (i, j) rather than only on $i - j$. Prior work proposes learning these factors from initialization but does not evaluate this variant; its reported instantiations use closed forms, factorize a pretrained table, or fit an existing bias, thereby enforcing or inheriting relativity. When trained freely in vision, only 37.2% of the learned bias energy remains displacement-dependent, compared with 100% for the unfactorized relative bias it replaces. We restore relativity by generating factors from learned frequencies, whose inner product reduces to $\sum_k a_k b_k \cos\big(\mathbf{w}_k^\top (\mathbf{p}_i - \mathbf{p}_j)\big)$ and therefore depends only on displacement by algebraic identity. A learned phase lifts the resulting even-symmetry restriction, still depending only on displacement, without increasing factor width. Kernel and computational cost are unchanged, while positional parameters become independent of window size and context length; in Swin-T at window 32, they fall from $1{,}277{,}952$ to $4{,}416$. At matched factor width, architecture, and training recipe, restoring relativity reduces language-model perplexity degradation at $16\times$ the training context by roughly $7\times$ across ranks 16–128, while using $683\times$ fewer positional parameters and improving in-distribution perplexity. In vision, it improves transfer to semantic segmentation and detection, and out-of-domain generalization at the larger window, while the phase variant strengthens robustness at both. These results provide a practical route to learned relative biases within fused attention across vision, language, and other position-structured domains.

1 INTRODUCTION

Attention underlies modern Transformer architectures Vaswani et al. (2017) across language Brown et al. (2020); Touvron et al. (2023), vision Cao et al. (2022); Hatamizadeh et al. (2022), multimodal learning Liu et al. (2023), and scientific modeling Wu et al. (2024). Self-attention alone, however, is permutation-equivariant and cannot distinguish token order or spatial location. Transformers therefore inject positional structure additively, through positional encodings or attention biases Vaswani et al. (2017); Liu et al. (2021), or multiplicatively, by transforming queries and keys Su et al. (2024); Heo et al. (2024). Additive biases encode relationships absent from token content, including spatial offsets, token distances, and scientific geometry. This structure is often *relative*: interactions should depend on the displacement between positions rather than on their absolute locations.

Relative biases, however, are not natively supported by standard efficient-attention kernels such as FlashAttention Dao et al. (2022); Dao (2024); Shah et al. (2024). Conventional relative biases are stored as lookup tables and indexed for every token pair Liu et al. (2021; 2022); naively incorporating them therefore requires a dense, head-specific bias matrix. Materializing and transferring this matrix undermines the principal advantage of FlashAttention, which avoids writing the quadratic score and attention matrices to high-bandwidth memory. Exact integration without materialization is possible in two ways: a handwritten kernel can implement a particular bias, as FlashAttention-2 does for ALiBi Dao (2024), or a compiler can fuse a user-defined bias into the tiled score computation Dong et al. (2024). The former is restricted to biases explicitly supported by the kernel, whereas the

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

latter depends on a compatible compilation stack and backend. These dependencies are manageable for a fixed bias and deployment setting, but limit portability when a learned positional mechanism must operate unchanged across resolutions, context lengths, architectures, and domains—including high-resolution images, 3D medical volumes, video, long-context sequences, and large meshes.

Low-rank factorization represents the bias through per-token factors incorporated directly into the query–key product, preserving fused training and inference Philippenko et al. (2025); Wu et al. (2026). This efficiency, however, changes the bias's structure. A relative bias depends only on the displacement between two positions, whereas independently learned factors can depend on both positions arbitrarily. Factorization therefore enables fusion but does not guarantee relativity, removing an inductive bias that has proved valuable in language and vision models Cao et al. (2022); Liu et al. (2022); Shaw et al. (2018); Press et al. (2021); Dai et al. (2019). Prior work mentions optimizing such factors from random initialization but does not evaluate this setting Wu et al. (2026). Its reported variants instead use closed-form factors, decompose a pretrained relative-bias table, or optimize factors to reconstruct an existing relative bias; each therefore enforces or inherits relativity. Whether freely learned factors recover this structure remains unknown. A concurrent approach generates factors with a coordinate network rather than storing them explicitly Lee et al. (2026), but conditions that network on absolute positions and likewise provides no guarantee of relativity. Whenever absolute position is predictive—for example, when objects tend to be centered in an image-classification dataset—the optimizer can exploit this freedom by learning location-specific patterns instead of a displacement-only bias.

We measure displacement dependence by orthogonally projecting each learned bias onto the subspace of relative functions. At $R = 16$ (rank), freely trained factors retain only 37.2% relative energy in vision and 25.2% in language, versus 100% for unfactorized relative biases and closed-form constructions such as ALiBi Press et al. (2021). We restore relativity by generating factors from learned frequencies whose inner product depends only on displacement by algebraic identity. This preserves the factor width, kernel, and fusion cost while using fewer positional parameters, independent of resolution or context length. The difference is negligible in-distribution: less than 0.1 ImageNet-1K accuracy points Deng et al. (2009) and one perplexity point. Under distribution, task, resolution, and context-length shifts, however, it becomes substantial. Extending a language model from 512 to $8{,}192$ tokens increases perplexity by $11.4\times$ for freely learned factors but only $1.7\times$ for ours. This gap persists from $R = 16$ to $R = 128$, indicating a structural rather than capacity limitation. At matched rank, architecture, and training recipe, changing only the factor parameterization recovers 2.16 ImageNet-A points Hendrycks et al. (2021b) and 0.57 mIoU on ADE20K Zhou et al. (2019). ALiBi separates relativity from functional form: although both are exactly relative, its unbounded monotone penalty generalizes better to longer contexts ($0.92\times$ perplexity), whereas bounded cosines can oscillate beyond the training range. Ours still fits the training range at least as well, reaching perplexities of 24.51 at $R = 32$ and 24.31 at $R = 96$, versus ALiBi's 24.63. Together, these results establish a practical way to combine relative biases with fused attention, enabling their use at scale and opening opportunities to explore relative structure across vision, language, and scientific modeling. Our main *contributions* are:

- We show that freely learned low-rank biases preserve fusion but not relativity, and introduce a projection-based measure of displacement dependence.
- We propose a frequency-generated factorization that guarantees relativity at unchanged rank and fusion cost, with resolution- and context-independent parameters, and a learned phase that lifts the induced even symmetry.
- Controlled vision and language experiments isolate the effects of relativity on robustness, transfer, and length extrapolation, including where it provides limited benefit.

2 Related work

Relative position as an additive bias. Adding a term B_{ij} to the attention logits is the standard way to express positional relationships that token content does not carry. Shaw et al. (2018) introduced relative position representations for self-attention; Transformer-XL (Dai et al., 2019) and T5 (Raffel et al., 2020) refined them with, respectively, a decomposed relative term and a bucketed scalar bias shared across layers, and KERPLE (Chi et al., 2022) learns a kernelized distance penalty applied densely. In vision, Swin (Liu et al., 2021) learns a table indexed by the displacement between

tokens within a window, and SwinV2 (Liu et al., 2022) replaces the table with a small network evaluated on log-spaced relative coordinates (log-CPB), improving transfer across window sizes. All of these are relative by construction: B_{ij} is a function of $i - j$, either by indexing or by evaluating a function of the displacement. In their conventional implementations, these biases must be materialized, gathered, or evaluated for every token pair and attention head, putting them in tension with fused attention (Dao et al., 2022; Dao, 2024).

Fused attention and bias fusion. FlashAttention (Dao et al., 2022; Dao, 2024; Shah et al., 2024) avoids materializing the $N \times N$ score matrix, whereas a dense additive bias must still be loaded per head. Existing solutions require either bias-specific kernel support or compilation: FlashAttention-2 implements ALiBi slopes, xFormers (Lefaudeux et al., 2022) streams dense bias tiles, and FlexAttention (Dong et al., 2024) compiles arbitrary score modifications into fused kernels. Low-rank factorization instead writes $B \approx \phi_q \phi_k^\top$, with $\phi_q, \phi_k \in \mathbb{R}^{N \times R}$ and $R \ll N$, allowing the factors to be appended to queries and keys without bias-specific kernel logic. FlashBias (Wu et al., 2026) applies this strategy using closed-form ALiBi slopes (Press et al., 2021), squared distances for neural PDEs (Wu et al., 2024), a precomputed approximation of a pretrained Swin table, and a fit to an existing structure-prediction bias. These instantiations enforce or inherit relativity. It also states, but does not evaluate, learning free factors from initialization, where nothing constrains the learned bias to be relative. RIB (Lee et al., 2026) is closest to our approach: it generates factors from absolute coordinates using an MLP over Fourier features, making parameter count independent of window size. The inner product of those factors, however, need not depend only on displacement; RIB notes that its construction "does not guarantee equivalence across the same relative offsets." Its authors further report that freely learned FlashBias factors fail to converge when trained from scratch in their setting, consistent with the deficit we measure here. Our parameterization instead guarantees relativity algebraically, quantifies the displacement structure lost by freely learned factors, and measures its effect on transfer and context extension.

Low-rank structure in attention. Low-rank and linear approximations of attention are well studied (Wang et al., 2020; Choromanski et al., 2020; Katharopoulos et al., 2020). These methods approximate or restructure the attention computation to reduce its quadratic cost. In contrast, the factorization studied here applies only to the additive bias and leaves attention itself exact. The two directions are therefore orthogonal: our parameterization changes the structure the bias can represent, not how attention is computed, at matched rank, fused kernel, and number of appended channels.

Rotary and other multiplicative encodings. Rotary position embeddings (RoPE) (Su et al., 2024) encode position by rotating queries and keys before attention. Their inner product depends on position only through the difference between rotation angles, making it relative by construction without requiring a bias matrix and directly compatible with fused kernels. Two-dimensional extensions (Heo et al., 2024) adapt this construction to vision using mixed, learnable spatial frequencies. Our method applies the same geometric intuition to an *additive* bias. Rather than rotating content-dependent queries and keys, we generate separate positional factors whose inner product forms the bias. This transfers RoPE's relative-by-construction property to low-rank additive biases while keeping them independent of the query–key content interaction. They can therefore encode standalone relational preferences, such as distance penalties or geometric kernels, while remaining compatible with fused attention. RoPE and our method are complementary: RoPE makes content interactions relative, whereas we use its geometry to make a learned additive bias relative.

3 METHOD

Fused low-rank attention biases. Let $\mathbf{q}_i, \mathbf{k}_j \in \mathbb{R}^{d_h}$ denote the query and key at positions i and j, and let $B \in \mathbb{R}^{N \times N}$ be an additive attention bias. The attention weights are

$$A_{ij} = \mathrm{softmax}_j \left(s\, \mathbf{q}_i^\top \mathbf{k}_j + B_{ij} \right), \tag{1}$$

where $s = 1/\sqrt{d_h}$ is the query–key scaling factor. Fused attention kernels avoid materializing the $N \times N$ score matrix; supplying B as a dense tensor would reintroduce the memory traffic this design avoids. A rank-R factorization, $B = \phi_q \phi_k^\top$, with $\phi_q, \phi_k \in \mathbb{R}^{N \times R}$, allows the bias to be evaluated within the fused kernel. Defining the augmented vectors

$$\widetilde{\mathbf{q}}_i = [\sqrt{s}\, \mathbf{q}_i;\ \phi_q(i)], \qquad \widetilde{\mathbf{k}}_j = [\sqrt{s}\, \mathbf{k}_j;\ \phi_k(j)] \tag{2}$$

gives

$$\widetilde{\mathbf{q}}_i^\top \widetilde{\mathbf{k}}_j = s\,\mathbf{q}_i^\top \mathbf{k}_j + \phi_q(i)^\top \phi_k(j) = s\,\mathbf{q}_i^\top \mathbf{k}_j + B_{ij}. \tag{3}$$

The bias is therefore applied without materialization at the cost of R additional channels per head. Because the augmented vectors have dimension $d_h + R$, the scaling is folded into $\widetilde{\mathbf{q}}_i$ and $\widetilde{\mathbf{k}}_j$ and the kernel's own softmax scale is set to 1; otherwise it would apply $1/\sqrt{d_h + R}$. We begin with the variant in which ϕ_q and ϕ_k are learned freely from initialization. In a windowed vision transformer, each is a tensor of shape (M^2, H, R) per block, where M is the window size and H is the number of attention heads. This parameterization places no constraint on $\phi_q(i)^\top \phi_k(j)$: it may depend independently on the absolute positions $\mathbf{p}_i$ and $\mathbf{p}_j$, rather than only on their displacement $\mathbf{p}_i - \mathbf{p}_j$.

Relativity as an Algebraic Identity. Inspired by the geometric construction of RoPE (Su et al., 2024; Heo et al., 2024), we generate the low-rank bias factors from learned frequencies rather than storing a separate factor for every position. Unlike RoPE, which rotates content-dependent queries and keys, we apply the cosine–sine construction to standalone positional factors whose inner product forms an additive bias. Let $\mathbf{p}_i \in \mathbb{R}^{d_p}$ denote the position of token i — its index in a sequence ($d_p = 1$) or its coordinate within a window ($d_p = 2$) — and let $\{\mathbf{w}_k\}_{k=1}^K \subset \mathbb{R}^{d_p}$ be learned frequencies with learned scalar amplitudes $a_k, b_k \in \mathbb{R}$. For each frequency, we define

$$\phi_q(i)_{2k-1:2k} = a_k \begin{bmatrix} \cos(\mathbf{w}_k^\top \mathbf{p}_i) \\ \sin(\mathbf{w}_k^\top \mathbf{p}_i) \end{bmatrix}, \qquad \phi_k(j)_{2k-1:2k} = b_k \begin{bmatrix} \cos(\mathbf{w}_k^\top \mathbf{p}_j) \\ \sin(\mathbf{w}_k^\top \mathbf{p}_j) \end{bmatrix}. \tag{4}$$

Using the algebraic identity $\cos u \cos v + \sin u \sin v = \cos(u - v)$, their inner product becomes

$$B_{ij} = \phi_q(i)^\top \phi_k(j) = \sum_{k=1}^{K} a_k b_k \cos\big(\mathbf{w}_k^\top (\mathbf{p}_i - \mathbf{p}_j)\big). \tag{5}$$

Every term depends only on $\mathbf{p}_i - \mathbf{p}_j$; hence, B is relative for any $\mathbf{w}_k$, a_k, and b_k. Relativity is therefore a property of the parameterization rather than the learned optimum: it requires no regularizer, target bias, or auxiliary loss, and no parameter assignment can introduce dependence on absolute position. Each frequency contributes one cosine–sine pair, giving a factor width of $R = 2K$ and factors with the same shape as the freely learned baseline. The real-valued amplitudes a_k and b_k allow each coefficient $a_k b_k$ to take either sign, avoiding the explicit sign variable and singular derivative introduced by a signed square-root parameterization. Algorithm 1 in Appendix A.1 gives the generation procedure, including the frequency initialization and the `fp32` evaluation of the phases.

Context extension without interpolation. The factors are generated from token positions at forward time rather than stored in position-indexed tables. Their learned parameters depend on the number of heads, frequencies, and positional dimensions, but not on the sequence length N. Freely learned factors instead define no values beyond the training context and must be resized or extended with untrained parameters. Ours directly evaluates the same frequencies at new positions, ensuring that every factor remains defined without interpolation. This guarantees only that the parameterization is defined, not that performance is preserved under arbitrary context extension. The same property holds when changing the window size in vision.

Lifting the symmetry restriction. Because cosine is even, Equation 5 yields $B_{ij} = B_{ji}$: the bias assigned to displacement $\mathbf{d}$ is identical to that assigned to $-\mathbf{d}$. The parameterization therefore cannot distinguish above from below or left from right within a window. This is a real restriction: in learned Swin and SwinV2 biases, 18.8–30.2% of the relative-bias energy lies in the odd component (Appendix A.6.1). Allowing both cosine and sine components yields the more general form $\sum_k [\alpha_k \cos(\mathbf{w}_k^\top \mathbf{d}) + \beta_k \sin(\mathbf{w}_k^\top \mathbf{d})]$. We recover the odd component by rotating the query-side phase by a learned offset φ_k:

$$a_k \begin{bmatrix} \cos(u + \varphi_k) \\ \sin(u + \varphi_k) \end{bmatrix}^\top b_k \begin{bmatrix} \cos v \\ \sin v \end{bmatrix} = a_k b_k \cos(u - v + \varphi_k). \tag{6}$$

This spans both cosine and sine components using one additional scalar per head and frequency, without increasing the factor width, fusion cost, or number of channels appended to the attention kernel. We initialize $\varphi_k = 0$, making the variant identical to Equation 5 at initialization; any subsequent difference is therefore attributable to the learned phase. We evaluate both forms, using the symmetric parameterization as a control that isolates the contribution of directional asymmetry.

Measuring Relativity. To quantify how nearly a learned bias is relative, we find the closest matrix that depends only on displacement and measure how much of the original bias energy it retains. Let $\mathcal{B}_{\mathbf{d}}$ contain all index pairs (i, j) with displacement $\mathbf{d} = \mathbf{p}_i - \mathbf{p}_j$. The closest relative matrix is obtained by replacing each entry with the mean of its displacement bin:

$$\widehat{B}_{ij} = \frac{1}{|\mathcal{B}_{\mathbf{d}(i,j)}|} \sum_{(i',j') \in \mathcal{B}_{\mathbf{d}(i,j)}} B_{i'j'}, \qquad \rho = \frac{\|\widehat{B}\|_F^2}{\|B\|_F^2}. \tag{7}$$

Because the displacement bins partition the matrix entries, bin averaging is the orthogonal projection onto the space of relative biases. Thus, $\rho \in [0, 1]$ is the fraction of bias energy explained by displacement, with $\rho = 1$ for a bias that is relative by construction. For vision, $\mathbf{d} = (\Delta y, \Delta x)$; for language, $\mathbf{d} = i - j$ is scalar. In causal language models, we compute both the bin means and ρ over the unmasked entries $j \leq i$, since masked entries do not affect attention. For reference, an isotropic random bias has expected relative share $2/(N+1) = 0.39\%$ under causal masking at $N = 512$ and $(2M-1)^2/M^4 = 0.38\%$ in a 32×32 window. We report ρ without centering; Appendix A.3 gives the causal and two-dimensional conventions and examines softmax-invariant corrections.

4 Results and Discussion

Initialization. Frequency magnitudes are initialized on a log-spaced ladder, $\|\mathbf{w}_k\| = \theta^{-k/K}$ for $k = 0, \ldots, K-1$. In vision, orientations are sampled uniformly per head, following mixed-frequency rotary vision transformers (Heo et al., 2024), with $\theta = 10$ for pixel-scale displacements; in language, $\mathbf{w}_k$ is scalar and $\theta = 10^4$ to span thousands of tokens. Amplitudes are initialized independently as $a_k, b_k \sim \mathcal{N}(0, s^2)$ with $s = K^{-1/8}$, so $a_k b_k$ has variance s^4 and the initial bias scale at zero displacement, $\sqrt{K}s^2 = K^{1/4}$, grows mildly with rank. Appendix A.4.6 repeats the sweep with $s = K^{-1/4}$, which fixes this scale, and finds that the rank trend is preserved.

Implementation details. Language models are trained on one NVIDIA H200, with three seeds unless stated otherwise, and evaluated by token-level perplexity (lower is better). Vision models—FlashBias and our variants—are trained on ImageNet-1K for 300 epochs using the same codebase and recipe, with one run per model. *Robustness.* We evaluate without fine-tuning on ImageNet-V2 Recht et al. (2019), -A Hendrycks et al. (2021b), -R Hendrycks et al. (2021a), and -Sketch Wang et al. (2019), reporting top-1 accuracy over the standard 200-class subsets for -A and -R. On ImageNet-C Hendrycks & Dietterich (2019), we report AlexNet-normalized mean corruption error over 15 corruptions and five severities. We resize the released 224×224 images using our 256×256 transform rather than regenerating the corruptions, so absolute mCE is not directly comparable with values from the standard protocol; the transform is nevertheless identical across models. *Downstream transfer.* We evaluate UPerNet Xiao et al. (2018) on ADE20K using mIoU and Mask R-CNN He et al. (2017) on COCO Lin et al. (2014) using box and mask AP. Full training details and pseudocode appear in Appendices A.2 and A.1.

Efficiency results. Table 1 compares throughput and peak memory at matched batch size and resolution, with factorized methods using $R = 16$. At window 32 and patch 2, materializing the bias requires about 139 GiB, compared with 25–33 GiB for kernel-computed and factorized biases. Throughput alone does not motivate factorization: FlexAttention compiles the displacement-table lookup, preserves the full SwinV2 bias, and reaches 819 training img/s at 6.7 GiB for patch 4, within 2.1% of our 836 img/s; at patch 2, it reaches 214 versus our 229 img/s. The augmented-query form instead provides kernel portability, while our frequency generation additionally provides parameter efficiency. Writing the bias as $\phi_q \phi_k^\top$ turns it into ordinary attention over augmented queries and keys, requiring no bias-specific kernel logic. Our method therefore runs on unmodified FlashAttention-2 at 852 training img/s and can inherit future kernel improvements, provided the augmented head dimension is supported. In our setup, the tested FlexAttention backend required a power-of-two head dimension and could not compile a Swin-T variant with head dimension 24. RIB has the same peak memory and comes within 1.7% and 0.4% of our training throughput at patches 4 and 2, respectively. From an efficiency perspective, the principal difference is parameter count: its coordinate MLP stores 157,824 positional parameters, compared with 4,416 for our window-independent frequencies and amplitudes.Appendix A.4.1 reports similar efficiency trends for the language models.

Table 1: Inference/training throughput (images/s), peak training memory, and positional parameter counts for Swin-T at window 32, 256px, batch 64, on one H200. Factorized methods use $R = 16$; patch 2 has four times as many tokens as patch 4. Parameters are counted at window 32.

	Patch 4			Patch 2			
Method	Infer	Train	GiB	Infer	Train	GiB	Pos. params
Bias materialized on the score matrix							
Swin, indexed table	1128	486	45.0	227	103	138.7	151,434
SwinV2, log-CPB	1084	394	38.7	222	87	138.9	89,088
Bias computed inside the kernel							
SwinV2 + FlexAttention	2987	819	6.7	745	214	24.9	89,088
SwinV2 + dense bias, memory-efficient	2663	699	8.4	630	177	33.1	89,088
Bias factorized at rank $R = 16$							
FlashBias, free factors	3140	866	6.7	786	231	26.0	1,277,952
RIB, coordinate MLP	2817	822	6.7	767	228	26.0	157,824
Ours	3042	836	6.7	778	229	26.0	4,416
Ours + phase	3045	836	6.7	778	229	26.0	5,520
Ours, stock FlashAttention-2	2872	852	7.1	738	236	26.5	4,416

The memory-efficient row forces PyTorch's `EFFICIENT_ATTENTION` SDPA backend, derived from xFormers (Lefaudeux et al., 2022), preventing fallback to the math implementation. The final row forces PyTorch's `FLASH_ATTENTION` backend, corresponding to stock FlashAttention-2 without bias-specific logic. Swin, SwinV2, FlexAttention, FlashBias, and RIB follow Liu et al. (2021; 2022); Dong et al. (2024); Wu et al. (2026); Lee et al. (2026).

4.1 Language-Modelling Results

Why language modelling isolates relativity. Context extension provides a controlled stress test of relativity. Freely learned factors are position-indexed $N \times R$ tensors with no parameters beyond the training context; we extend them by linearly resampling ϕ_q and ϕ_k along the position axis with aligned endpoints, stretching the learned grid to the evaluation length. This is the language analogue of Swin's bicubic table resizing and avoids leaving added positions untrained. Ours instead generates factors from position and remains defined at any length. Across parameterizations, we hold rank, attention kernel, optimizer, schedule, and token budget fixed; for each trained model, only the evaluation length changes. We train 51M-parameter decoder-only Transformers on WikiText-103 (Merity et al., 2016) at context 512 and evaluate from 512 to 8,192 tokens, with three seeds per configuration, reporting the mean and one standard deviation. ALiBi (Press et al., 2021) is included as a reference: it is relative by construction, uses no learned positional parameters, and is the language-model instantiation adopted by FlashBias (Wu et al., 2026). We also train RIB (Lee et al., 2026) under the same recipe to separate generating factors from constraining them (Appendix A.5); training cost for every variant is reported in Appendix A.4.1.

Freely learned factors are only partially relative. We measure relativity over the entries causal attention uses ($j \leq i$), excluding logits masked before softmax. At $R = 16$, freely learned factors retain $\rho = 25.2\% \pm 4.0$ of their bias energy in the relative subspace, against 100% for ALiBi and for ours by construction. Chance at $N = 512$ is $2/(N+1) = 0.39\%$, placing the freely learned bias about $65\times$ above chance but far below exact relativity; including masked entries lowers ρ further, so the causally active region is the more relative one. RIB, which generates factors with a coordinate network, lies between the two: pooled over ranks and seeds it retains $62.9\% \pm 9.1\%$, with per-rank means from 55.4% to 71.5% and no systematic improvement with rank (Table 10). Factor generation therefore recovers some displacement structure but leaves the remainder to optimization.

Relativity improves extrapolation without sacrificing in-distribution fit. Figure 1(A) reports perplexity at the training context. For $R \geq 8$, our parameterization matches or improves upon freely learned factors at every rank, lowering perplexity by 0.23–0.70 against seed standard deviations of 0.02–0.32. Once factor width is sufficient, the constraint therefore does not trade in-distribution fit for generalization. Under context extension the parameterizations diverge sharply (Figure 1(B)). At $R = 32$ and context 8,192, freely learned factors reach 286.1 ± 19.2 perplexity against 41.2 ± 5.8 for ours, i.e. degradation factors of 11.42 ± 0.79 and 1.68 ± 0.21 relative to context 512; their ratio stays between $6.8\times$ and $7.9\times$ at every rank from $R = 16$ to $R = 128$ (Table 3). The separation appears after one context doubling: at 1,024 tokens ours is unchanged at $R = 32$ and $R = 128$ ($0.99\times$), whereas freely learned factors degrade by $1.43\times$ and $2.09\times$. Rank does not resolve the deficit — at 8,192 their degradation moves only from $11.42\times$ at $R = 32$ to $11.12\times$ at $R = 128$,

despite fourfold greater positional capacity, while their rank@99% rises from 30 to 69 (Table 5), showing the deficit is structural rather than unused capacity. Training both variants at context 1,024 yields the same qualitative separation (Table 7).

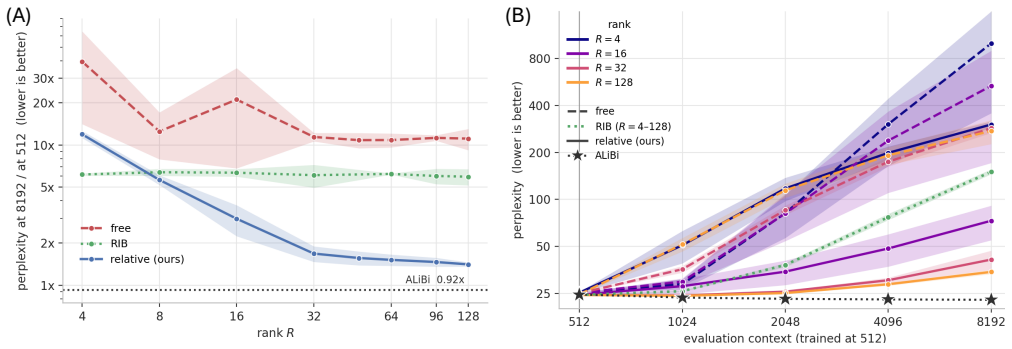

Figure 1: **WikiText-103 language modelling, trained at context 512; lower is better.** Lines are three-seed means ± 1 standard deviation; ALiBi stars denote a single run. **(A)** Training-context perplexity versus rank. Ours matches or improves upon freely learned factors for $R \geq 8$; RIB (Lee et al., 2026) performs similarly. **(B)** Perplexity versus evaluation context. Colour denotes rank; freely learned factors are dashed and ours solid. The RIB band spans $R = 4$–128 because its curves overlap. Freely learned factors degrade beyond 512, higher-rank relative variants remain stable through the first doubling, and RIB remains intermediate and largely insensitive to rank.

Generating factors is insufficient; constraining them is. RIB shares the practical properties of our method: its factors are generated at forward time, its positional parameter count does not grow with context, and its training throughput is within 0.5% of ours (Table 4). However, its factors depend on absolute coordinates, so their inner product need not depend only on displacement. Under our recipe RIB matches our in-distribution perplexity but degrades by $6.35 \pm 0.39\times$ at $R = 16$ and $6.10 \pm 1.13\times$ at $R = 32$, against $2.98 \pm 0.74\times$ and $1.68 \pm 0.21\times$ for ours (Table 10). Across the sweep our degradation falls from $11.93\times$ at $R = 4$ to $1.41\times$ at $R = 128$, whereas RIB stays flat ($6.16\times$ to $5.95\times$, within seed variation), as does its relative share. Additional width helps only when the parameterization ties it to displacement; otherwise it captures structure that does not transfer beyond the training context.

Low ranks are initialization-sensitive, with diminishing returns above $R = 32$. Raising R from 16 to 32 reduces degradation by $1.30\times$, whereas increasing it fourfold to $R = 128$ gains only another $0.27\times$. From $R = 32$ to $R = 128$, rank@99% grows from 9 to 15, reducing utilization from 28% to 12% (Table 5). Because the default initialization increases bias scale with rank, we repeat the sweep at fixed scale. Degradation still falls from $4.05\times$ at $R = 16$ to $1.44\times$ at $R = 128$, confirming that additional frequencies, rather than initialization scale, drive the improvement (Table 9). With one cosine–sine pair, the default $R = 2$ model is degenerate and highly variable under extension, so Figure 1(B) omits it. Placing its single frequency at the centre of the initialization rung largely removes this failure and outperforms matched freely learned factors, indicating optimization sensitivity rather than an inherent representational limit. It remains inferior to ALiBi, whose closed-form monotone function is task-aligned, whereas ours must learn a basis from one frequency. Mid-band initialization does not consistently help at larger ranks, localizing the sensitivity to the extremely low-rank regime. Appendix Tables 6, 7, and 5 provide complete results and bias-shape diagnostics.

The phase does not improve language modelling. Causal attention uses only $j \leq i$ and hence one displacement sign, removing the need to distinguish $\mathbf{d}$ from $-\mathbf{d}$. The phase and symmetric forms reach comparable training-context perplexity, but phase degrades more under extension when rank is limited, and the difference vanishes by $R = 128$. We therefore use the symmetric form in the language experiments and report the ablation in Appendix Table 8.

Relativity does not fully determine extrapolation. ALiBi does not degrade under extension ($0.92\times$) and outperforms every learned parameterization evaluated. We treat it as a closed-form reference rather than a matched competitor: our controlled comparison varies relativity while holding rank, kernel, architecture, and recipe fixed. KERPLE (Chi et al., 2022), Sandwich (Chi et al., 2023), and FIRE (Li et al., 2024) fall outside it because they do not use a fused low-rank bias, and rotary embeddings are not content-independent additive biases representable as $\phi_q \phi_k^\top$. Because AL-

iBi and ours are both exactly relative, functional form explains their remaining difference: ALiBi's unbounded monotone penalty keeps suppressing increasingly distant tokens, whereas our cosine basis is bounded and no head remains monotone across the full evaluation range (Table 5). Ours instead fits the training context at least as well, matching or improving on ALiBi from $R = 16$ onward and reaching 24.31 ± 0.05 at $R = 96$ against 24.63. Relativity therefore explains much of the gap to freely learned factors, while functional form governs the residual extrapolation gap to ALiBi. Learned relative bases combining this in-distribution flexibility with monotone long-range behaviour remain future work.

Positional parameters. At context N with H heads, freely learned factors require $2NHR$ positional parameters, whereas ours requires $3HR/2$. The ratio is $4N/3$, independent of rank and head count, a $683\times$ reduction at $N = 512$.

4.2 Vision Results

Setup. We compare three Swin-T window-bias parameterizations at patch size 4 and resolution 256: freely learned FlashBias factors (Wu et al., 2026) (§3.2), our symmetric relative factors, and the phase variant. All use factor width $R = 16$, the same fused kernel and patch-embedding stem, and the same 300-epoch ImageNet-1K recipe with effective batch 1024. We use unfactorized SwinV2 (Liu et al., 2022) as the window-16 reference and train each factorized variant at windows 16 and 32. RIB (Lee et al., 2026) has no released ImageNet classifier; Appendix A.6.4 instead analyzes its released DF2K super-resolution checkpoint for relative structure, not task performance.

All parameterizations perform similarly in distribution. ImageNet-1K accuracy varies by only 0.4 points across the seven models in Table 2, so the robustness and transfer differences occur among models with comparable in-distribution performance. At window 16, replacing the unfactorized SwinV2 bias with freely learned FlashBias factors reduces performance on every evaluated task: ImageNet-1K ($\downarrow 0.40$), ImageNet-V2 ($\downarrow 0.32$), ImageNet-A ($\downarrow 2.34$), ImageNet-R ($\downarrow 1.55$), ImageNet-Sketch ($\downarrow 1.32$), ADE20K ($\downarrow 0.75$ mIoU), and COCO detection ($\downarrow 1.3$ box AP), while mCE worsens by 0.86 points. The largest deficits occur on ImageNet-A, ImageNet-R, ImageNet-Sketch, and COCO detection.

Generated factors are only partially relative. Freely learned factors retain $\rho = 37.2\%$ relative energy at window 32 and 37.1% at window 16, compared with 100% for ours by construction; chance is 0.379% and 1.47%, respectively. They therefore acquire relative structure but remain strongly dependent on absolute position. RIB's released vision model (Lee et al., 2026) shows the same pattern: its coordinate-generated biases average 54.0% relative energy and range from 4.0% to 91.8% across layers. Because this is a DF2K super-resolution model rather than an ImageNet classifier, the values are diagnostic rather than directly comparable. Full results appear in Appendix A.6.4.

Table 2: **Vision results.** Swin-T models use patch size 4, factor width $R = 16$, and 300-epoch ImageNet-1K pretraining; downstream transfer uses 512×512 inputs. Higher is better except mCE. SwinV2 is the unfactorized reference and is available only at window 16.

	ImageNet						Transfer	
	IN-1K	IN-V2	IN-A	IN-R	Sketch	mCE↓	ADE20K	COCO box
Window 16								
SwinV2 (unfactorized)	82.80	71.40	31.23	44.65	31.47	54.11	45.46	39.0
FlashBias (free)	82.40	71.08	28.89	43.10	30.15	54.97	44.71	37.7
Ours	82.56	70.96	28.79	43.55	29.95	55.20	44.88	37.9
Ours + phase	82.76	71.36	30.15	44.26	30.86	54.14	44.83	38.0
Window 32								
FlashBias (free)	82.65	71.30	28.43	42.98	29.40	54.65	45.15	37.0
Ours	**82.73**	**71.99**	30.59	43.55	30.13	53.55	**45.72**	38.4
Ours + phase	82.69	71.86	**31.33**	**44.06**	**30.52**	**53.32**	45.57	**38.6**

Symmetric relativity improves robustness at window 32, but not window 16. At window 32, replacing freely learned factors with our symmetric relative parameterization improves ImageNet-A by 2.16 points, ImageNet-V2 by 0.69, ImageNet-R by 0.57, and ImageNet-Sketch by 0.73, while reducing mCE by 1.10. At window 16, however, the same intervention decreases ImageNet-A and Sketch by 0.10 and 0.20 points and increases mCE from 54.97 to 55.20. Symmetric relativity alone

therefore does not explain the robustness recovery, even though the freely learned biases lose nearly the same fraction of relative structure at both windows.

Learned relative biases contain substantial directional structure. The symmetric parameterization represents only the even component of a relative bias: each term satisfies $B(\mathbf{d}) = B(-\mathbf{d})$, making the odd component $\frac{1}{2}(B - B^\top)$ unreachable at any rank. We measure this excluded component in the relative biases learned by Swin-T and SwinV2-T after removing each head's softmax-invariant constant offset. The odd component contains 19.1% of the centered bias energy in Swin-T, 18.8% in SwinV2-T at window 8, and 30.2% in the window-16 SwinV2 reference. Individual heads exceed 90% odd energy in every model. Learned relative biases can therefore be strongly directional, while the symmetric form discards this structure by construction. Complete decompositions, including raw values before centering, are reported in Table 11.

Directionality improves robustness at both windows. The phase variant remains exactly relative while distinguishing opposite displacements. It improves ImageNet-A, ImageNet-R, Sketch, and mCE over both the symmetric and freely learned variants at each window. At window 16, its largest effect, it improves the symmetric form by 1.36, 0.71, and 0.91 points on ImageNet-A, ImageNet-R, and Sketch, respectively, while reducing mCE by 1.06. It nearly matches SwinV2 on mCE (54.14 versus 54.11) and ImageNet-V2 (71.36 versus 71.40), but remains 1.08 points lower on ImageNet-A. Both relative variants have $\rho = 100\%$, yet all eight robustness comparisons favor phase, showing that ρ does not capture directional structure. Phase does not improve clean ImageNet-1K or ImageNet-V2 at window 32.

Dense-prediction gains persist without interpolation. At window 32, the symmetric parameterization improves freely learned factors by 0.57 mIoU on ADE20K and 1.4 box AP on COCO. Phase changes these results by only -0.15 mIoU and $+0.2$ box AP, suggesting that directionality matters less for dense prediction than robustness. Symmetric and phase variants improve mask AP by 1.0 and 1.2 points, respectively (Table 12). At window 16, all factorized variants remain below SwinV2; the best trails it by 0.58 mIoU, 1.0 box AP, and 0.7 mask AP. At 512px, FlashBias must interpolate its position-indexed factors because later stages use larger effective windows than during 256px pretraining; ours are generated at the required positions. At 256px, where neither method requires interpolation, ours retains gains of 0.6 box AP and 0.3 mask AP. The benefit therefore persists without interpolation, although its larger magnitude at 512px may partly reflect avoided resizing. Because resolution also changes the detection regime, the two effects cannot be exactly separated; full results appear in Appendix A.6.3.

Limitations and future work. Very low ranks remain sensitive to frequency initialization: performance deteriorates below $R = 8$ and degenerates at $R = 2$, although mid-band initialization largely removes the $R = 2$ failure. Developing adaptive frequency initialization or learning the frequency range may improve this regime. Symmetry is domain-dependent: phase improves vision robustness but harms language extrapolation at limited rank under causal masking, motivating parameterizations that adapt directional capacity to the attention geometry. Our vision experiments use only $R = 16$ and one run per configuration, while the language experiments cover one model scale, dataset, and training context of 512. Future work should test whether the observed rank behavior persists across vision scales, larger language models, and longer training contexts. Finally, the bounded cosine basis extrapolates less effectively than ALiBi's unbounded monotone penalty. Combining learned relative frequencies with a monotone long-range component may retain flexible in-distribution fitting while improving extrapolation.

5 Conclusion

Low-rank factorization makes additive biases compatible with fused attention but can discard relativity. Freely learned factors remain only partially relative, weakening robustness, transfer, and context extrapolation. Inspired by RoPE, we introduced a frequency-generated factorization that guarantees relativity without changing factor width or kernel cost. A learned phase additionally captures directional structure without increasing rank. Across vision and language, restoring relativity improves robustness, dense prediction, and length extrapolation, although bounded cosines extrapolate less effectively than ALiBi. More broadly, computationally efficient factorizations can remove useful inductive biases. Preserving them algebraically enables expressive learned relative biases within fused attention and motivates richer bases across vision, language, and scientific modelling.

AI USE STATEMENT

The core research idea and project conceptualization were developed independently of generative AI. We used generative AI tools to assist with checking numerical calculations; writing and debugging experimental, and analysis code; verifying that methodological descriptions were grounded in the implementation; and editing portions of the manuscript for clarity and organization. The authors designed the experiments, selected the evaluation criteria, launched and supervised all training and evaluation runs, and interpreted the results. Generative AI was not used to generate datasets or autonomously determine the scientific conclusions. The authors verified all mathematical claims, numerical results, and citations against the implementation, experimental outputs, and original sources. All AI-assisted text was reviewed and revised by the authors, who take full responsibility for the final content, claims, and conclusions.

ETHICS STATEMENT

This work studies the computational and representational properties of attention-bias parameterizations using established public benchmarks. It does not involve human subjects, personal or sensitive data, or the collection of new datasets. We are not aware of ethical risks specific to the proposed method beyond those associated with the models and application domains in which attention mechanisms are deployed.

REPRODUCIBILITY STATEMENT

We provide the complete model formulation, including the factor construction, initialization, symmetry extension, and relativity metric, in the Methods section. Experimental architectures, training recipes, hardware, datasets, preprocessing, evaluation protocols, and baseline configurations are described in the main text and supplementary material. The appendix reports complete per-rank and per-context language-modeling results, individual-seed outcomes, additional initialization experiments, and bias diagnostics. We will release the full training and evaluation code, configuration files, preprocessing scripts, random seeds, model checkpoints, and raw results needed to reproduce all vision and language experiments upon publication. During double-blind review, the Pseudo code and supporting materials are provided in appendix A.1-A.6.

REFERENCES

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation. In *European conference on computer vision*, pp. 205–218. Springer, 2022.

Ta-Chung Chi, Ting-Han Fan, Peter J Ramadge, and Alexander Rudnicky. Kerple: Kernelized relative positional embedding for length extrapolation. *Advances in Neural Information Processing Systems*, 35:8386–8399, 2022.

Ta-Chung Chi, Ting-Han Fan, Alexander Rudnicky, and Peter Ramadge. Dissecting transformer length extrapolation via the lens of receptive field analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13522–13537, 2023.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. *Advances in neural information processing systems*, 33:18613–18624, 2020.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 2978–2988, 2019.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*, 2024.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Juechu Dong, Boyuan Feng, Driss Guessous, Yanbo Liang, and Horace He. Flex attention: A programming model for generating optimized attention kernels. *arXiv preprint arXiv:2412.05496*, 2(3):4, 2024.

Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger R Roth, and Daguang Xu. Unetr: Transformers for 3d medical image segmentation. In *2022 IEEE/CVF winter conference on applications of computer vision (WACV)*, pp. 1748–1758. Ieee, 2022.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8320–8329. IEEE, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15262–15271, 2021b.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, pp. 289–305. Springer, 2024.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pp. 5156–5165. PMLR, 2020.

Dongheon Lee, Seokju Yun, Jaegyun Im, and Youngmin Ro. Rank-factorized implicit neural bias: Scaling super-resolution transformer with flashattention. *arXiv preprint arXiv:2603.06738*, 2026.

Benjamin Lefaudeux, Francisco Massa, Diana Liskovich, Wenhan Xiong, Vittorio Caggiano, Sean Naren, Min Xu, Jieru Hu, Marta Tintore, Susan Zhang, Patrick Labatut, Daniel Haziza, Luca Wehrstedt, Jeremy Reizenstein, and Grigory Sizov. xformers: A modular and hackable transformer modelling library. `https://github.com/facebookresearch/xformers`, 2022.

Shanda Li, Chong You, Guru Guruganesh, Joshua Ainslie, Santiago Ontanon, Manzil Zaheer, Sumit Sanghai, Yiming Yang, Sanjiv Kumar, and Srinadh Bhojanapalli. Functional interpolation for relative positions improves long context transformers. In *International Conference on Learning Representations*, volume 2024, pp. 11303–11328, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9992–10002. Ieee, 2021.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11999–12009. IEEE, 2022.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

Constantin Philippenko, Kevin Scaman, and Laurent Massoulié. In-depth analysis of low-rank matrix factorisation in a federated setting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19904–19912, 2025.

Ofir Press, Noah A Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*, 2021.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

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.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pp. 5389–5400. PMLR, 2019.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *Advances in Neural Information Processing Systems*, 37:68658–68685, 2024.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 464–468, 2018.

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

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. *arXiv preprint arXiv:2402.02366*, 2024.

Haixu Wu, Minghao Guo, Yuezhou Ma, Yuanxu Sun, Jianmin Wang, Wojciech Matusik, and Mingsheng Long. Flashbias: Fast computation of attention with bias. *Advances in Neural Information Processing Systems*, 38:20729–20756, 2026.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *European conference on computer vision*, pp. 432–448. Springer, 2018.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International journal of computer vision*, 127(3):302–321, 2019.