Rope to Nope and Back Again: A New Hybrid Attention Strategy

Bowen Yang [1] Bharat Venkitesh [1] Dwarak Talupuru [1] Hangyu Lin [1] David Cairuz [1] Phil Blunsom [1]
Acyr Locatelli [1]

Abstract

Long context large language models (LLMs) have achieved remarkable advancements, driven by techniques like Rotary Position Embedding (RoPE) (Su et al., 2023) and its extensions (Chen et al., 2023; Liu et al., 2024c; Peng et al., 2023). By adjusting RoPE parameters and incorporating training data with extended contexts, we can train performant models with considerably longer input sequences. However, existing RoPE-based methods exhibit performance limitations when applied to extended context lengths. This paper presents a comprehensive analysis of various attention mechanisms, including RoPE, No Positional Embedding (NoPE), and Query-Key Normalization (QK-Norm), identifying their strengths and shortcomings in long context modeling. Our investigation identifies distinctive attention patterns in these methods and highlights their impact on long context performance, providing valuable insights for architectural design. Building on these findings, we propose a novel architecture based on a hybrid attention mechanism that not only surpasses conventional RoPE-based transformer models in long context tasks but also achieves competitive performance on benchmarks requiring shorter context lengths.

Does not harm short content lengths

1. Introduction

Developing language models capable of handling long context lengths poses several challenges. First, as the context length increases, an effective modeling of extended input sequences becomes increasingly critical. This often requires advancements in positional encoding (Su et al., 2023), extrapolation techniques (Ding et al., 2024), or architectural innovations (Tworkowski et al., 2023; Huang et al., 2024). Second, training long context large language models with billions of parameters demands significant computational resources. Overcoming this challenge requires scalable algorithms, high-quality datasets, and robust infrastructure. Lastly, deploying these models in real-world applications demands low latency and low memory usage, which requires meticulous optimization of both the model architecture and the serving infrastructure. Addressing these issues requires a strategic balance between modeling, training, and deployment efficiency, while managing potential trade-offs.

On the modeling front, two components of the transformer architecture are particularly crucial for long context capabilities: the attention mechanism and positional embeddings. Recent research has proposed various methods to enhance these components. For instance, Landmark Attention (Mohtashami & Jaggi, 2023) trains attention modules to select relevant blocks using a representative token, referred to as a "landmark token", for efficient retrieval within extended text corpora. Similarly, Focused Transformer (Tworkowski et al., 2023) adopts a contrastive training approach to prioritize attending the most relevant portions of the input sequence, allowing the model to focus on smaller, contextually significant subsets of tokens. Although these approaches improve long context modeling, stabilizing training remains a key challenge for extending transformer capabilities to longer sequences. Query-Key Normalization (QK-Norm) (Henry et al., 2020; Rybakov et al., 2024) has been introduced to address the stability issue, which normalizes the query-key vectors along the head dimension before computing attention. Although QK-Norm mitigates numerical instability during training and is widely used (Team, 2024; Dehghani et al., 2023; Li et al., 2024b), it may impair long context capabilities.

In addition to the chosen attention mechanism, positional embeddings play a crucial role in long context modeling. Various approaches have been proposed to improve their effectiveness. Popular methods include Absolute Position Embedding (APE) (Vaswani et al., 2017), Relative Position Embedding (Raffel et al., 2023), ALiBi (Press et al., 2022), and Rotary Position Embedding (RoPE) (Su et al., 2023). Among these, RoPE has gained significant adoption in large language models (LLMs) (Dubey et al., 2024; Yang et al., 2024a; Üstün et al., 2024) due to its simplicity and effectiveness. In particular, it has the ability to extrapolate context lengths by adjusting RoPE θ values during training

[1] Cohere. Correspondence to: Bowen Yang <bowen@cohere.com>.

(Liu et al., 2024a; AI et al., 2024; Dang et al., 2024). Other techniques, such as relative bias (Press et al., 2022; Raffel et al., 2023; Chi et al., 2022) and contextualized position embeddings (Golovneva et al., 2024), introduce distance-based bias terms or condition the position information on input semantics. These methods often affect attention distribution by incorporating auxiliary information, such a positional indices or explicit recency bias. However, whether certain information or biases are beneficial to long context modeling or overall performance remains less explored. Additionally, the concept of No Positional Embedding (NoPE) has been explored by (Kazemnejad et al., 2023), suggesting that removing explicit positional embeddings and relying solely on implicit positional information derived from the causal mask can enhance long context performance.

NoPE has some implicit positional info from positional mask .

Though the aforementioned strategies help with long context modeling, training and serving long context models still present notable challenges. The vanilla attention mechanism that most transformers use have quadratic complexity relative to sequence length, making them prohibitively expensive for extended contexts. Techniques such as Sliding Window Attention (SWA) (Jiang et al., 2023; Team et al., 2024a) limit each token's attention to a fixed-size window of neighboring tokens, reducing computational overhead while preserving local contextual understanding. Other methods, like sparse attention (Child et al., 2019; Beltagy et al., 2020; Tay et al., 2020a), generalize SWA by applying various sparsity patterns, including random sparsity (Zaheer et al., 2021) and dilated attention (Ding et al., 2023). More recent strategies, such as activation beacon (Zhang et al., 2024), further optimize attention by compressing local keys and values sequentially, or attention sink (Xiao et al., 2024), which points out that tokens at the beginning of the sequences contribute to a disproportionately large mass of attention score and demonstrating that retaining these tokens and employing a sliding window approach can stabilize loss over longer contexts. On the serving side, methods like KV cache trimming (Zhang et al., 2023; Li et al., 2024a; Cai et al., 2024) selectively reduce KV cache size using certain heuristics, thereby minimizing inference memory usage and improving throughput. While reducing attention span can improve efficiency, it often involves trade-offs in model performance, underscoring the importance of a balanced design.

H_2O, SnapKV, Pyramidal

In this paper, we begin analyzing attention patterns of different attention mechanisms,, NoPE, and QK-Norm and its impacts on long context performance trained up to 750 billion tokens. Building on these insights, we propose a new modeling approach and extensively pretrain up to 5 trillion tokens, followed by supervised fine-tuning on a diverse set of datasets tailored for long context. We show that this approach surpasses existing state-of-the-art extrapolation-based RoPE models (Liu et al., 2024c) by a large margin, striking a balance between efficiency and performance.

2. Observation

In this section, we assess three models with different attention components, use needles-in-a-haystack (Kamradt, 2023) (NIAH) and analyze the attention patterns to understand how these variants affect performance. This analysis guides our architectural design choices throughout this work.

2.1. Experimental Setup

All model variants share a common configuration consisting of 8 billion parameters (including the token embedding parameters), with detailed architectural specifications provided in Table 1. For this set of experiments, the model is trained in two stages: a pretraining stage followed by a supervised fine-tuning (SFT) stage. Previous research shows that the SFT stage is necessary for long context evaluations, as it can reduce variance in long context tasks and enables the emergence of long context capabilities that may not manifest in models trained solely through pretraining (Gao et al., 2024).

We pretrain the model with a batch size of 4 million tokens. We use AdamW with a peak learning rate of $7e^{-3}$, a linear warmup of 2000 steps and a cosine learning rate schedule decaying to $3.5e^{-4}$ over 179,000 steps for a total of 750 billion tokens. For the SFT stage, we adopt an interleaved training strategy: we combine short- and long context data in a 3:1 ratio, with context lengths of 8192 and 65536 tokens, respectively. We use a batch size of 0.5 million tokens.

Parameters	8B
Embedding Dim	4096
FFN Dim	28672
Non-linearity	swiglu
Num Layers	32
Num Heads	32
Num KV Heads	8
Vocab Size	256000

Table 1: Model architecture details

The 3 model variants we test are:

1. **RoPE Model:** For this variant, we employ Rotary Position Embedding (RoPE) to encode positional information. During the pretraining stage, the RoPE parameter θ is set to 10,000. In the subsequent supervised fine-tuning (SFT) stage, θ is increased to 2 million to account for the increased context length. This variant serves as the baseline model configuration, maintaining an architecture similar to that of most existing models (Dubey et al., 2024; Jiang et al., 2023; Cohere For AI, 2024).

2. **QK-Norm Model:** Layer normalization (Ba et al., 2016) is applied to both the query and the key vectors before performing the angular rotation used in RoPE. All other hyperparameters, including the θ values and training methodology, remain identical to those of the RoPE variant.

QK Norm done before RoPE

3. **NoPE Model:** Previous research (Wang et al., 2024; Haviv et al., 2022) has demonstrated that transformer variants trained without positional embeddings (NoPE) can perform effectively on long context tasks. However, these models often exhibit inferior performance in terms of perplexity and downstream task evaluations within the trained sequence length (Haviv et al., 2022). In our study, the NoPE variant does not have QK-Norm. This variant is trained using the same methodology as the other two variants.

2.2. Evaluation and Attention Analysis

2.2.1. Evaluations

We evaluate the variants on a set of core evaluation benchmarks, including MMLU (Hendrycks et al., 2021), HellaSwag (Zellers et al., 2019), CommonsenseQA (Talmor et al., 2019), ARC (Clark et al., 2018) for core model capabilities and NIAH benchmark (Kamradt, 2023) for long context capability. NIAH evaluates a model's ability to retrieve information accurately from a specific sentence (the "needle") embedded within a lengthy document (the "haystack"). The needle is randomly placed at varying depths within the sequence to examine performance across different context lengths. To improve robustness, we modify the original NIAH benchmark, where each context-depth combination is tested 16 times with different random seeds, creating diverse context compositions for comprehensive evaluation. The results of all standard benchmark evaluations and results with 65536 context length needles are presented in Table 2. Although prior research has emphasized the limitations of NIAH (Xu et al., 2024) for evaluating deeper and more general context understanding, our focus is solely on testing basic long context capabilities and gaining insights on model architecture design, for which this benchmark is sufficient.

Table 2 shows that the RoPE and QK-Norm variants exhibit comparable performance on standard benchmarks, while the NoPE variant lags behind. This finding is consistent with previous studies (Kazemnejad et al., 2023; Wang et al., 2024). For long context evaluations, QK-Norm performs the worst among the three variants, despite its decent performance in other capabilities. This observation is consistent with the results from the comparisons between Command R and Command R+, where Command R, despite being a significantly smaller model, outperforms Command R+ overall on longer context benchmarks (Hsieh et al., 2024). Although the NoPE variant has slightly lower needles score compared to the RoPE variant, it is decent given that its base capabilities is relatively low.

2.2.2. Attention Pattern Analysis

To investigate the impacts of different architectures, we also analyze the attention patterns within each model. This approach is inspired by previous studies (Ye et al., 2024; Leviathan et al., 2024), which examined attention scores to gain insight on improving model performance.

We still utilize the NIAH task by dividing the context into four segments: the first 10 tokens (begin), the needle sentence tokens (needle), general context tokens (context), and question/completion tokens (qc). We position the needles at approximately 50% depth within the context to increase the complexity of the task, as most models suffer from the lost-in-the-middle problem, as highlighted in previous works (Liu et al., 2024b; Baker et al., 2024). For each model, we first calculate the attention scores between the query tokens of "qc" and the key tokens of all four segments across all heads and layers. The attention scores are summed within each segment and then aggregated across all heads and layers to obtain the average attention score for each segment. These scores are further averaged across multiple samples at sequence lengths of 8,000 tokens, 32,000 tokens, and 128,000 tokens. We refer to this metric as "attention mass" in the following sections. The results are reported in Table 8.

Table 8 reveals that all variants exhibit decreasing attention mass on needle tokens as the sequence length increases. This indicates that retrieving relevant information becomes more challenging as the sequence grows longer. Additionally, within each context length, the NoPE variant demonstrates the highest attention mass on needle tokens, followed by the RoPE variant, while the QK-Norm variant has the lowest attention mass. We also notice that the QK-Norm variant has extremely low attention mass for the beginning tokens (attention sink (Xiao et al., 2024)) compared to other variants and higher attention mass over the noisy context. This is consistent with QK-Norm's relatively poor performance in the NIAH task. We argue that QK-Norm has this effect because the normalization operation mitigates magnitude information from the dot product of Query and Key vectors which tends to result in attention logits being closer in terms of magnitude and flatter in terms of distribution. A more detailed analysis can be found in Appendix B.

☆ Attention Mass studies

☆ Why QK Norm has low attention mass

2.2.3. Hybrid Model

Building on the findings above, we combine RoPE and NoPE to leverage the strengths of both approaches. NoPE's effective attention mechanism for retrieving information

Model	Val Loss	MMLU	HellaSwag	CommonsenseQA	ARC-E	ARC-C	Needles 65k
RoPE	1.52	48.55	73.74	68.30	81.05	39.13	9.82
QK-Norm	1.53	48.21	73.68	68.23	80.54	38.98	7.93
NoPE	1.58	47.61	72.16	66.42	76.94	37.12	9.03

Table 2: Comparison of models on a variety of benchmarks. All evaluations are based on the performance of the SFT-models.

Context Length	Model Variants	begin	needle	context	qc
8k	RoPE	0.3863	0.0328	0.3809	0.2000
	QK-Norm	0.0242	0.0173	0.8020	0.1565
	NoPE	0.3058	0.0454	0.4501	0.1987
32k	RoPE	0.3541	0.0201	0.4343	0.1915
	QK-Norm	0.0064	0.0056	0.8517	0.1364
	NoPE	0.2807	0.0325	0.4981	0.1886
128k	RoPE	0.3463	0.0010	0.4751	0.1776
	QK-Norm	0.0010	0.0004	0.8993	0.0994
	NoPE	0.0846	0.0073	0.8156	0.0925

Table 3: Needles Attention Pattern: RoPE, QK-Norm and Nope

based on vector similarity and RoPE's explicit modeling of positional information and recency bias can, in combination, enhance general performance. We propose a new variant that alternates between NoPE and Rotary Position Embedding (RoPE) layers. Specifically, the two position-embedding strategies are interleaved, with NoPE applied in one layer and RoPE in the next. To ensure consistency and facilitate comparisons, the RoPE parameter θ is initially set to 10,000 during pre-training. We then conduct multiple fine-tuning runs with varying θ values, including 10,000, 100,000, 2 million, and 4 million, to evaluate the model's performance across different configurations. We refer to this variant as the RNoPE variant and RNoPE-10k, RNoPE-100k, RNoPE-2M, and RNoPE-4M with the numbers indicating the RoPE θ value used during SFT.

We report the needles evaluation score at a sequence length of 128,000 and the attention mass calculations (as in Table 8) for all variants. The results are shown in Table 4. The attention mass is aggregated separately for all RoPE and NoPE layers. For simplicity, we present results based on 32,000 sequence length samples, with the complete table for all sequence lengths provided in Appendix A.

Results in Table 4 reveal that as we increase the RoPE parameter θ during fine-tuning, the model's long context capability diminishes. This finding contradicts previous observations in pure RoPE models (Men et al., 2024; Liu et al., 2024a), where larger RoPE θ value during pretraining or fine-tuning improves long context performance and expands the effective attention receptive field. To explore this discrepancy, we compare the attention mass between different model variants.

First, we observe a clear difference in behavior among the RNoPE variants layers. The NoPE layers exhibit strong retrieval capabilities, characterized by a pronounced spike in attention mass on the needle tokens and a phenomenon of attention sink (Xiao et al., 2024) at the beginning tokens. Furthermore, the NoPE layers show significantly weaker recency bias compared to pure RoPE or pure NoPE models. In contrast, the RoPE layers in RNoPE variants demonstrate extremely weak retrieval capabilities and demonstrate almost no attention sink, as evidenced by their relatively low attention mass on the beginning and needle tokens. However, these RoPE layers exhibit a much stronger recency bias compared to the pure RoPE variant or their NoPE layer counterparts.

Even higher than pure RoPE

When comparing between different θ values, we find that as θ increases, the recency bias of the RoPE layers decreases, as reflected in the decreasing attention mass of the qc tokens. This supports previous findings and theoretical analyses suggesting that increasing θ expands the effective receptive field of the attention mechanism (Men et al., 2024), resulting in a flatter attention distribution. However, our empirical observations suggest that a larger receptive field in the RoPE layers introduces noise, which disrupts the ability of the

Even in RNoPE, high θ not used

- Long content capability ↓
- Recency bias diminishes (good for long ctx) BUT large receptive field introduces noise which disrupts ability of subsequent NoPE layers to compute similarity.

Model	NoPE Layers				RoPE Layers				needles-128k
	begin	needle	context	qc	begin	needle	context	qc	
RoPE	-	-	-	-	0.3541	0.0201	0.4343	0.1915	7.395
RNoPE-10k	0.3275	0.0765	0.5672	0.0287	0.0049	0.0004	0.6805	0.3142	8.036
RNoPE-100k	0.3263	0.0778	0.5633	0.0327	0.0241	0.0005	0.6782	0.2972	7.461
RNoPE-2M	0.3250	0.0712	0.5735	0.0303	0.1111	0.0046	0.6233	0.2611	7.022
RNoPE-4M	0.3486	0.0369	0.5981	0.0165	0.0960	0.0039	0.6774	0.2227	6.203
RNoPE-10k-swa	0.3303	0.0742	0.5634	0.0321	-	-	-	-	9.562

Table 4: Needles Attention Pattern: RoPE and RNoPE variants

subsequent NoPE layers to compute similarities and perform retrieval tasks effectively, ultimately leading to lower needle evaluation scores. For example, the needle attention mass of the NoPE layers dropped from 0.0765 to 0.0369, and the needle evaluation score decreased from 8.036 to 6.203 as θ increased from 10,000 to 4 million.

From these observations, we draw the following insights:

1. Combining NoPE and RoPE layers provides distinct advantages in processing information. NoPE layers excel at information retrieval, while RoPE layers handle local information effectively due to their built-in recency bias.

2. Restricting the effective attention span of the RoPE layers in RNoPE models can denoise the attention mass and ensure that each layer type adheres to its specialized role.

θ = 10k was best so used with SWA as well.

Based on these insights, we fine-tune a new variant, RNoPE-10k-swa, where "swa" stands for sliding window attention. This approach introduces a hard limit on the attention span of the RoPE layers, enforcing the second insight from above. Specifically, we set the sliding window size for the RoPE layers to 8,192, while maintaining the full attention span of the NoPE layers for long context retrieval. The rest of the training process remains identical to the RNoPE-10k variant, with no changes to θ. The evaluation results, reported in Table 4, demonstrate a notable improvement. The RNoPE-10k-swa variant achieves a needles-128k score of 9.562, outperforming both the baseline and the RNoPE-10k model. Additionally, its NoPE layers exhibit a well-structured attention pattern, indicative of strong long context retrieval capabilities.

3. Model Architecture

Based on the analysis above, we make the following design choices: the following design choices are made for the model architecture, building on top of the Command R+ architecture (Cohere For AI, 2024). First, we remove the QK-Norm component due to its poorly shaped attention patterns, which adversely impacted long context performance. Second, NoPE layers with a full attention span are introduced to enhance the model's retrieval capabilities. Third, a sliding window size of 4,096 is applied to RoPE embeddings, leveraging RoPE's inherent recency bias to improve performance on short-to-medium context ranges. In particular, the sliding-window approach has been employed in several prior works (Team et al., 2024a; Jiang et al., 2023; Character.AI, 2024). Regarding the number of layers, we perform an ablation study on the interleaving ratio of full attention and sliding window layers, testing the configurations of 1:1, 1:3, and 1:7. The results show that a 1:3 ratio strikes an optimal balance between computational efficiency and performance. Furthermore, we investigate the sequential ordering of full attention layers and find that it did not affect the results. As a result, we position the full attention layers at the end of each interleaving group, preceded by three sliding window layers. All other hyperparameters for the model architecture remain consistent with those outlined in Table 1.

Full : SWA

Moreover, the combination of RoPE embeddings with sliding window layers effectively reduces computation cost and memory overhead while maintaining a recency bias, which is well suited for short-to-medium context ranges. Meanwhile, NoPE layers with full attention spans are employed to capture long-range dependencies, addressing the limitations of purely local architectures. Earlier transformer variants, such as GPT-J (Wang & Komatsuzaki, 2021) and GPT-NeoX (Black et al., 2022), also explored combining RoPE and NoPE by applying rotational embeddings to a portion of the head dimensions. However, the effectiveness of these approaches in length extrapolation remains an open question and an area of ongoing research.

For additional context, the resulting design shares similarities with other well-explored long-sequence architectures, such as Mega (Ma et al., 2023) and state-space models (SSMs) (Gu et al., 2022; Gu & Dao, 2024). For example,

Mega

(Ma et al., 2023) and (Ma et al., 2024) introduced a multi-dimensional damped version of an exponential moving average component in conjunction with the gated attention unit (GAU) architecture (Hua et al., 2022), aiming to balance local and long-term dependencies, a common challenge in time-series modeling. Similarly, previous studies have proposed a diagonal variant of the S4 architecture (Gu et al., 2022), incorporating an exponentially decaying measure to enhance the model's ability to capture long-range dependencies (Tay et al., 2020b).

4. Experiments

In this section, we detail the experiments conducted on the model architectures, covering the stages of pretraining, cooldown, supervised fine-tuning, and evaluations. Alongside long context evaluations, we provide a comprehensive assessment of short-context benchmarks, a dimension often disregarded in other long context studies. We train two models: one with the RoPE architecture as a baseline and another employing the architecture introduced in Section 3.

Pretraining and Cooldown. The models are pretrained for 5 trillion tokens of diverse data with batch size of 8 million tokens using FP8 precision format (Micikevicius et al., 2022). We use a cosine learning rate schedule of 5e-3 peak learning rate and 5% end learning rate with 8,000 linear warmup steps. From the pre-trained model, we linearly anneal the learning rate from 2.5e-4 to 1e-6 for 50,000 steps in BF16 precision. The context length was initially maintained at 8k for the first 35,000 steps, then extended to 32k and 128k for 10,000 steps and 5,000 steps, respectively. For the baseline model, the RoPE θ values were increased to 1,000,000 for 32k and 8,000,000 for 128k contexts during the length extrapolation phase, while remaining constant for the RNope-SWA variant. Both models utilized the same interleaved training strategy outlined in Section 2.1.

LR annealing dtype needs to be considered as well.

Supervised Finetuning. We supervise fine-tune on top of the pretrained models. As the primary focus of this study is to evaluate the impact of architectural design on downstream tasks, preference training is deferred to future work. To preserve the long context capabilities of the model, the fine-tuning process utilizes interleaved datasets containing 8k and 128k prompt-response pairs. The long context SFT data at 128k includes Retrieval-Augmented Generation (RAG) (Lewis et al., 2021) datasets, multilingual translation datasets with extended passages, long code instruction datasets, and long context retrieval datasets. Training was performed for two epochs across the entire dataset.

5. Experimental Results

Our evaluation contains a comprehensive analysis over standard benchmarks below 8k context length such as MMLU (Hendrycks et al., 2021), HellaSwag (Zellers et al., 2019), ARC (Clark et al., 2018), SAT (Zhong et al., 2023), GSM8k (Cobbe et al., 2021), Winograde (Sakaguchi et al., 2019) and MBPP (Austin et al., 2021), as well as popular long context benchmarks with needles-in-a-haystack (Kamradt, 2023) and the retrieval and QA portion of Ruler (Hsieh et al., 2024). We test the context lengths up to 256k so we can examine the impact of these choices in the extrapolation capability of the model. We denote the baseline model trained with RoPE scaling as "Baseline" and the architecture with interleaved attention span and position embeddings as "RNope-SWA".

5.1. Standard Benchmarks

In this set of evaluations, we evaluate baseline and RNope-SWA on a standard LLM benchmark covering various language, math and code capabilities. The results are shown in Table 5. We observe that the model is better or on-par on most of the metrics compared to the baseline and has some gains over the baseline numbers on certain benchmarks (+2.0% on MMLU and +1.8% on GSM8k). This set of results also indicates that although RNope-SWA explicitly removed position embeddings from 25% of all its layers, positional information is retained by the interleaving attention span and captured by RoPE from previous layers. RNope-SWA does not have the performance loss due to the removal of explicit position embeddings, as previous works have shown (Kazemnejad et al., 2023; Wang et al., 2024).

5.2. Long Context Benchmarks

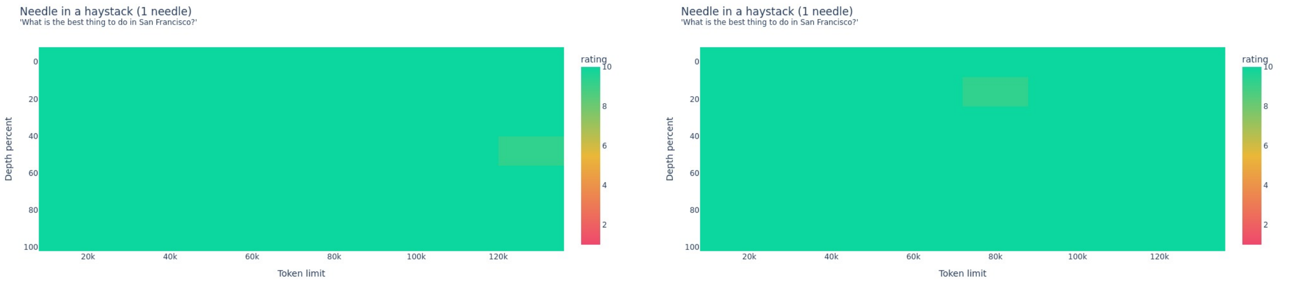
To evaluate the long context performance of these models, we use NIAH and the retrieval and QA components of Ruler (Hsieh et al., 2024). To better understand how architectural choices affect long context performance, we also evaluate with context lengths extending beyond training sequence length. This allows us to assess how well these models can interpolate as well as extrapolate to unseen context lengths and how specific architectural decisions influence these capabilities.

5.2.1. NIAH EVALUATIONS

Following the settings of section 2.2, we run NIAH test twice with 128k and 256k context lengths respectively. The scores are reported in Figure 2. The figure indicates that although both models are able to get close to perfect scores below the context length seen during training, RNope-SWA has better extrapolation capabilities and achieves almost no loss up to 256k context length while the baseline fails to extrapolate well – despite using a high RoPE θ value of 8 million.

Model	MMLU	HellaSwag	ARC-E	ARC-C	SATEn	SATMath	GSM8K	Winogrande	MBPP
Baseline	57.5	75.8	**84.6**	48.5	70.0	**30.9**	40.9	68.5	39.1
RNope-SWA	**59.5**	**76.2**	82.5	**48.8**	**71.9**	30.5	**42.7**	**69.5**	**39.3**

Table 5: Comparison of models on a variety of benchmarks. All the evaluations are based on the performance of the SFT-models.

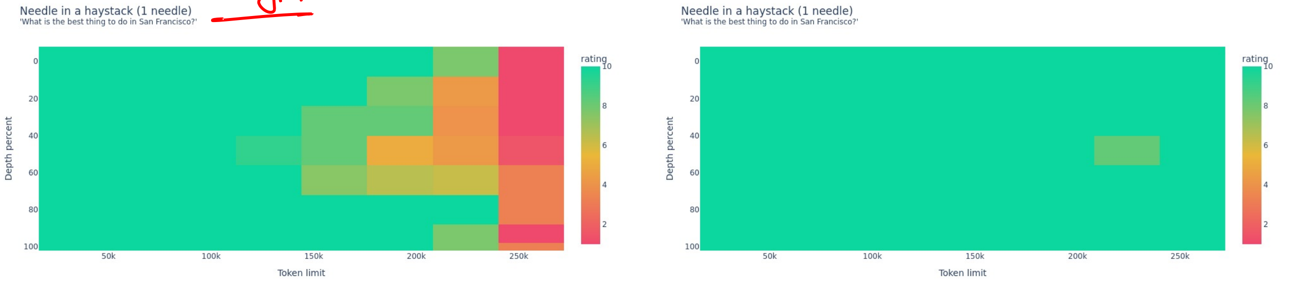


(a) Baseline model up to 128k. The raw score is 9.99

(b) RNope-SWA up to 128k. The raw score is 9.99

Trained till this length

Figure 1: NIAH evaluation at 128k



(a) Baseline model up to 256k. The raw score is 8.25

(b) RNope-SWA up to 256k. The raw score is 9.97

Figure 2: NIAH evaluation at 128k and 256k

Model	8k	16k	32k	64k	128k	256k
Baseline	**96.6**	94.4	**95.1**	89.1	83.0	57.1
RNope-SWA	96.1	**96.1**	94.9	**92.0**	**90.0**	**74.8**

Table 6: Ruler Retrieval Evaluation

Model	8k	16k	32k	64k	128k	256k
Baseline	53.5	50.0	52.5	45.5	36.0	30.0
RNope-SWA	**55.5**	**52.5**	**55.5**	**49.0**	**46.0**	**42.5**

Table 7: Ruler QA Evaluation

5.2.2. Ruler Evaluations

First introduced in (Hsieh et al., 2024), the Ruler benchmark aims to provide a set of more difficult tasks than NIAH. It covers a wider range of retrieval under a Multi-Query/Key/Value settings, more realistic tasks with a long-context Question-Answering format and more. Although our modification of NIAH introduced more context variants and proves to be more difficult than the vanilla version, it still cannot test the limits of the model. Therefore, we evaluate our models on the retrieval and QA portion of the Ruler so we can better separate their performance.

From the results, we can observe that the baseline model with RoPE θ scaling approach suffers from a sharp drop in the longer context range, especially 64k and longer. Comparing the difference between the scores obtained at 8k and 256k, the baseline model dropped from 96.6 to 57.1 (about 41%) on retrieval and from 53.5 to 30.0 (about 44%) on QA, whereas the RNope-SWA model dropped from 96.1 to 74.8 (about 22.1%) on retrieval and from 55.5 to 42.5 (about 23.4%) on QA respectively. From the original Ruler Paper (Hsieh et al., 2024), models that adopt similar RoPE scaling approaches have shown a similar degradation over longer context lengths (Dubey et al., 2024; Yang et al., 2024a;

Much lower degradation

Abdin et al., 2024) as the baseline.

5.3. Impacts on Training and Inference

S=4096

We also report the differences in training and inference speed, as well as memory requirements, of RNope-SWA compared to the baseline model. Let S denote the sliding window size and L represent the full training context length. During training, 75% of all layers now operate with a computational complexity of $\mathcal{O}(SL)$, rather than the quadratic complexity of $\mathcal{O}(L^2)$. This results in the model being approximately 50% faster than the baseline at a 64k context length and nearly 2x faster at 128k in terms of training throughput, using flash attention (Dao et al., 2022; Dao, 2023; Shah et al., 2024) and a sequence-parallel scheme similar to (Jacobs et al., 2023; Yang et al., 2024b). With alternative implementations, such as Ring Attention (Liu & Abbeel, 2023; Liu et al., 2023) or its variants (Brandon et al., 2023), sliding window adoption can reduce key-value block communication overhead if carefully sharded along the sequence dimension, potentially improving speed.

Interesting Papers

For inference, there is a theoretical upper bound of 75% for memory savings on kv cache and time complexity reductions. Empirically, we observed an approximate 44% reduction in end-to-end latency compared to the baseline model with full attention when using 132k input tokens and 96 output tokens. With 990k input tokens and 8 output tokens, the latency reduction increases to nearly 70%. As the sequence length increases, the latency savings approach the theoretical limit of 75% within our architectural setting. Furthermore, increasing the proportion of sliding window layers relative to full attention layers can further enhance both speed and memory efficiency.

6. Discussions and Future Work

In this paper, we introduced RNope-SWA, an architecture that interleaves NoPE and RoPE position embeddings with varying attention spans (RNope-SWA). RNope-SWA is able to strike a balance between effective attention modeling and computational efficiency, achieving a nearly 4x reduction in KV cache size and significantly boosting both training and inference speeds without compromising performance. The integration of NoPE layers with full attention spans enhances long context capabilities, eliminating the need for RoPE scaling. This simplification improves the stability of training and delivers excellent long context performance.

Our findings align with recent work, such as YoCo (Sun et al., 2024), Jamba-1.5 (Team et al., 2024b), and MiniMax-01 (MiniMax et al., 2025), which demonstrate that hybrid attention mechanisms generally outperform full attention mechanisms in handling long contexts. However, the underlying reasons behind this seemingly counterintuitive observation remain largely unexplored. This opens an intriguing area of study, particularly as models push towards multi-million-token context lengths. Re-visiting and re-thinking the foundational components of transformer architectures, such as attention mechanisms, may become essential to accommodating these extreme requirements. Recent works (Ye et al., 2024; Leviathan et al., 2024; Soh et al., 2024) have begun to explore this direction by focusing on reducing attention noise across large context windows, a promising approach to refine the performance of attention modules.

In the mean time, there is growing interest in more sophisticated techniques for integrating contextual and positional information. These methods have significant potential but introduce challenges in maintaining runtime efficiency and compatibility with optimization frameworks like flash attention (Dao, 2023). Research efforts (Golovneva et al., 2024), have begun exploring these possibilities, offering new directions for advancing both efficiency and performance in long context processing.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning algorithms. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Abdin, M., Aneja, J., Awadalla, H., Awadallah, A., Awan, A. A., Bach, N., Bahree, A., Bakhtiari, A., Bao, J., Behl, H., Benhaim, A., Bilenko, M., Bjorck, J., Bubeck, S., Cai, M., Cai, Q., Chaudhary, V., Chen, D., Chen, D., Chen, W., Chen, Y.-C., Chen, Y.-L., Cheng, H., Chopra, P., Dai, X., Dixon, M., Eldan, R., Fragoso, V., Gao, J., Gao, M., Gao, M., Garg, A., Giorno, A. D., Goswami, A., Gunasekar, S., Haider, E., Hao, J., Hewett, R. J., Hu, W., Huynh, J., Iter, D., Jacobs, S. A., Javaheripi, M., Jin, X., Karampatziakis, N., Kauffmann, P., Khademi, M., Kim, D., Kim, Y. J., Kurilenko, L., Lee, J. R., Lee, Y. T., Li, Y., Li, Y., Liang, C., Liden, L., Lin, X., Lin, Z., Liu, C., Liu, L., Liu, M., Liu, W., Liu, X., Luo, C., Madan, P., Mahmoudzadeh, A., Majercak, D., Mazzola, M., Mendes, C. C. T., Mitra, A., Modi, H., Nguyen, A., Norick, B., Patra, B., Perez-Becker, D., Portet, T., Pryzant, R., Qin, H., Radmilac, M., Ren, L., de Rosa, G., Rosset, C., Roy, S., Ruwase, O., Saarikivi, O., Saied, A., Salim, A., Santacroce, M., Shah, S., Shang, N., Sharma, H., Shen, Y., Shukla, S., Song, X., Tanaka, M., Tupini, A., Vaddamanu, P., Wang, C., Wang, G., Wang, L., Wang, S., Wang, X., Wang, Y., Ward, R., Wen, W., Witte, P., Wu, H., Wu, X., Wyatt, M., Xiao, B., Xu, C., Xu, J., Xu, W., Xue, J., Yadav, S., Yang, F., Yang, J., Yang, Y., Yang, Z., Yu, D., Yuan, L., Zhang, C., Zhang, C., Zhang, J., Zhang, L. L., Zhang, Y., Zhang, Y., Zhang, Y., and Zhou, X. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL https://arxiv.org/abs/2404.14219.

AI, ., :, Young, A., Chen, B., Li, C., Huang, C., Zhang, G., Zhang, G., Li, H., Zhu, J., Chen, J., Chang, J., Yu, K., Liu, P., Liu, Q., Yue, S., Yang, S., Yang, S., Yu, T., Xie, W., Huang, W., Hu, X., Ren, X., Niu, X., Nie, P., Xu, Y., Liu, Y., Wang, Y., Cai, Y., Gu, Z., Liu, Z., and Dai, Z. Yi: Open foundation models by 01.ai, 2024. URL https://arxiv.org/abs/2403.04652.

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., Jiang, E., Cai, C., Terry, M., Le, Q., and Sutton, C. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization, 2016. URL https://arxiv.org/abs/1607.06450.

Baker, G. A., Raut, A., Shaier, S., Hunter, L. E., and von der Wense, K. Lost in the middle, and in-between: Enhancing language models' ability to reason over long contexts in multi-hop qa, 2024. URL https://arxiv.org/abs/2412.10079.

Beltagy, I., Peters, M. E., and Cohan, A. Longformer: The long-document transformer, 2020. URL https://arxiv.org/abs/2004.05150.

Black, S., Biderman, S., Hallahan, E., Anthony, Q., Gao, L., Golding, L., He, H., Leahy, C., McDonell, K., Phang, J., Pieler, M., Prashanth, U. S., Purohit, S., Reynolds, L., Tow, J., Wang, B., and Weinbach, S. GPT-NeoX-20B: An open-source autoregressive language model. In Fan, A., Ilic, S., Wolf, T., and Gallé, M. (eds.), *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pp. 95–136, virtual+Dublin, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.bigscience-1.9. URL https://aclanthology.org/2022.bigscience-1.9/.

Brandon, W., Nrusimha, A., Qian, K., Ankner, Z., Jin, T., Song, Z., and Ragan-Kelley, J. Striped attention: Faster ring attention for causal transformers, 2023. URL https://arxiv.org/abs/2311.09431.

Cai, Z., Zhang, Y., Gao, B., Liu, Y., Liu, T., Lu, K., Xiong, W., Dong, Y., Chang, B., Hu, J., and Xiao, W. Pyramidkv: Dynamic kv cache compression based on pyramidal information funneling, 2024. URL https://arxiv.org/abs/2406.02069.

Character.AI. Optimizing ai inference at character.ai, 2024.

Chen, S., Wong, S., Chen, L., and Tian, Y. Extending context window of large language models via positional interpolation, 2023. URL https://arxiv.org/abs/2306.15595.

Chi, T.-C., Fan, T.-H., Ramadge, P. J., and Rudnicky, A. I. Kerple: Kernelized relative positional embedding for length extrapolation, 2022. URL https://arxiv.org/abs/2205.09921.

Child, R., Gray, S., Radford, A., and Sutskever, I. Generating long sequences with sparse transformers, 2019. URL https://arxiv.org/abs/1904.10509.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Cohere For AI. c4ai-command-r-plus (revision 432fac1), 2024. URL https://huggingface.co/CohereForAI/c4ai-command-r-plus.

Dang, J., Singh, S., D'souza, D., Ahmadian, A., Salamanca, A., Smith, M., Peppin, A., Hong, S., Govindassamy, M., Zhao, T., Kublik, S., Amer, M., Aryabumi, V., Campos, J. A., Tan, Y.-C., Kocmi, T., Strub, F., Grinsztajn, N., Flet-Berliac, Y., Locatelli, A., Lin, H., Talupuru, D., Venkitesh, B., Cairuz, D., Yang, B., Chung, T., Ko, W.-Y., Shi, S. S., Shukayev, A., Bae, S., Piktus, A., Castagné, R., Cruz-Salinas, F., Kim, E., Crawhall-Stein, L., Morisot, A., Roy, S., Blunsom, P., Zhang, I., Gomez, A., Frosst, N., Fadaee, M., Ermis, B., Üstün, A., and Hooker, S. Aya expanse: Combining research breakthroughs for a new multilingual frontier, 2024. URL https://arxiv.org/abs/2412.04261.

Dao, T. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023. URL https://arxiv.org/abs/2307.08691.

Dao, T., Fu, D. Y., Ermon, S., Rudra, A., and Ré, C. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022. URL https://arxiv.org/abs/2205.14135.

Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A., Caron, M., Geirhos, R., Alabdulmohsin, I., Jenatton, R., Beyer, L., Tschannen, M., Arnab, A., Wang, X., Riquelme, C., Minderer, M., Puigcerver, J., Evci, U., Kumar, M., van Steenkiste, S., Elsayed, G. F., Mahendran, A., Yu, F., Oliver, A., Huot, F., Bastings, J., Collier, M. P., Gritsenko, A., Birodkar, V., Vasconcelos, C., Tay, Y., Mensink, T., Kolesnikov, A., Pavetić, F., Tran, D., Kipf, T., Lučić, M., Zhai, X., Keysers, D., Harmsen, J., and Houlsby, N. Scaling vision transformers to 22 billion parameters, 2023. URL https://arxiv.org/abs/2302.05442.

Ding, J., Ma, S., Dong, L., Zhang, X., Huang, S., Wang, W., Zheng, N., and Wei, F. Longnet: Scaling transformers to 1,000,000,000 tokens, 2023. URL https://arxiv.org/abs/2307.02486.

Ding, Y., Zhang, L. L., Zhang, C., Xu, Y., Shang, N., Xu, J., Yang, F., and Yang, M. Longrope: Extending llm context window beyond 2 million tokens, 2024. URL https://arxiv.org/abs/2402.13753.

Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., and Angela Fan, e. a. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Gao, T., Wettig, A., Yen, H., and Chen, D. How to train long-context language models (effectively), 2024. URL https://arxiv.org/abs/2410.02660.

Golovneva, O., Wang, T., Weston, J., and Sukhbaatar, S. Contextual position encoding: Learning to count what's important, 2024. URL https://arxiv.org/abs/2405.18719.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces, 2024. URL https://arxiv.org/abs/2312.00752.

Gu, A., Goel, K., and Ré, C. Efficiently modeling long sequences with structured state spaces, 2022. URL https://arxiv.org/abs/2111.00396.

Haviv, A., Ram, O., Press, O., Izsak, P., and Levy, O. Transformer language models without positional encodings still learn positional information. In Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 1382–1390, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.99. URL https://aclanthology.org/2022.findings-emnlp.99/.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Henry, A., Dachapally, P. R., Pawar, S. S., and Chen, Y. Query-key normalization for transformers. In Cohn, T., He, Y., and Liu, Y. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4246–4253, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.379. URL https://aclanthology.org/2020.findings-emnlp.379/.

Hsieh, C.-P., Sun, S., Kriman, S., Acharya, S., Rekesh, D., Jia, F., Zhang, Y., and Ginsburg, B. Ruler: What's the real context size of your long-context language models?, 2024. URL https://arxiv.org/abs/2404.06654.

Hua, W., Dai, Z., Liu, H., and Le, Q. V. Transformer quality in linear time, 2022. URL https://arxiv.org/abs/2202.10447.

Huang, Y., Xu, J., Lai, J., Jiang, Z., Chen, T., Li, Z., Yao, Y., Ma, X., Yang, L., Chen, H., Li, S., and Zhao, P. Advancing transformer architecture in long-context large language models: A comprehensive survey, 2024. URL https://arxiv.org/abs/2311.12351.

Jacobs, S. A., Tanaka, M., Zhang, C., Zhang, M., Song, S. L., Rajbhandari, S., and He, Y. Deepspeed ulysses: System optimizations for enabling training of extreme long sequence transformer models, 2023. URL https://arxiv.org/abs/2309.14509.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Kamradt, G. Needle in a haystack–pressure testing llms, 2023.

Kazemnejad, A., Padhi, I., Ramamurthy, K. N., Das, P., and Reddy, S. The impact of positional encoding on length generalization in transformers, 2023. URL https://arxiv.org/abs/2305.19466.

Leviathan, Y., Kalman, M., and Matias, Y. Selective attention improves transformer, 2024. URL https://arxiv.org/abs/2410.02703.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., tau Yih, W., Rocktäschel, T., Riedel, S., and Kiela, D. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021. URL https://arxiv.org/abs/2005.11401.

Li, Y., Huang, Y., Yang, B., Venkitesh, B., Locatelli, A., Ye, H., Cai, T., Lewis, P., and Chen, D. Snapkv: Llm knows what you are looking for before generation, 2024a. URL https://arxiv.org/abs/2404.14469.

Li, Z., Zhang, J., Lin, Q., Xiong, J., Long, Y., Deng, X., Zhang, Y., Liu, X., Huang, M., Xiao, Z., and et al., D. C. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding, 2024b. URL https://arxiv.org/abs/2405.08748.

Liu, H. and Abbeel, P. Blockwise parallel transformer for large context models, 2023. URL https://arxiv.org/abs/2305.19370.

Liu, H., Zaharia, M., and Abbeel, P. Ring attention with blockwise transformers for near-infinite context, 2023. URL https://arxiv.org/abs/2310.01889.

Liu, H., Yan, W., Zaharia, M., and Abbeel, P. World model on million-length video and language with blockwise ringattention, 2024a. URL https://arxiv.org/abs/2402.08268.

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., and Liang, P. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024b. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Liu, X., Yan, H., Zhang, S., An, C., Qiu, X., and Lin, D. Scaling laws of rope-based extrapolation, 2024c. URL https://arxiv.org/abs/2310.05209.

Ma, X., Zhou, C., Kong, X., He, J., Gui, L., Neubig, G., May, J., and Zettlemoyer, L. Mega: Moving average equipped gated attention, 2023. URL https://arxiv.org/abs/2209.10655.

Ma, X., Yang, X., Xiong, W., Chen, B., Yu, L., Zhang, H., May, J., Zettlemoyer, L., Levy, O., and Zhou, C. Megalodon: Efficient llm pretraining and inference with unlimited context length, 2024. URL https://arxiv.org/abs/2404.08801.

Men, X., Xu, M., Wang, B., Zhang, Q., Lin, H., Han, X., and Chen, W. Base of rope bounds context length, 2024. URL https://arxiv.org/abs/2405.14591.

Micikevicius, P., Stosic, D., Burgess, N., Cornea, M., Dubey, P., Grisenthwaite, R., Ha, S., Heinecke, A., Judd, P., Kamalu, J., Mellempudi, N., Oberman, S., Shoeybi, M., Siu, M., and Wu, H. Fp8 formats for deep learning, 2022. URL https://arxiv.org/abs/2209.05433.

MiniMax, Li, A., Gong, B., Yang, B., Shan, B., Liu, C., Zhu, C., Zhang, C., Guo, C., Chen, D., Li, D., Jiao, E., Li, G., Zhang, G., Sun, H., Dong, H., Zhu, J., Zhuang, J., Song, J., Zhu, J., Han, J., Li, J., Xie, J., Xu, J., Yan, J., Zhang, K., Xiao, K., Kang, K., Han, L., Wang, L., Yu, L., Feng, L., Zheng, L., Chai, L., Xing, L., Ju, M., Chi, M., Zhang, M., Huang, P., Niu, P., Li, P., Zhao, P., Yang, Q., Xu, Q., Wang, Q., Wang, Q., Li, Q., Leng, R., Shi, S., Yu, S., Li, S., Zhu, S., Huang, T., Liang, T., Sun, W., Sun, W., Cheng, W., Li, W., Song, X., Su, X., Han, X., Zhang, X., Hou, X., Min, X., Zou, X., Shen, X., Gong, Y., Zhu, Y., Zhou, Y., Zhong, Y., Hu, Y., Fan, Y., Yu, Y., Yang, Y., Li, Y., Huang, Y., Li, Y., Huang, Y., Xu, Y., Mao, Y., Li, Z., Li, Z., Tao, Z., Ying, Z., Cong, Z., Qin, Z., Fan, Z., Yu, Z., Jiang, Z., and Wu, Z. Minimax-01: Scaling foundation models with lightning attention, 2025. URL https://arxiv.org/abs/2501.08313.

Mohtashami, A. and Jaggi, M. Landmark attention: Random-access infinite context length for transformers, 2023. URL https://arxiv.org/abs/2305.16300.

Peng, B., Quesnelle, J., Fan, H., and Shippole, E. Yarn: Efficient context window extension of large language models, 2023. URL https://arxiv.org/abs/2309.00071.

Press, O., Smith, N. A., and Lewis, M. Train short, test long: Attention with linear biases enables input length extrapolation, 2022. URL https://arxiv.org/abs/2108.12409.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL https://arxiv.org/abs/1910.10683.

Rybakov, O., Chrzanowski, M., Dykas, P., Xue, J., and Lanir, B. Methods of improving llm training stability, 2024. URL https://arxiv.org/abs/2410.16682.

Sakaguchi, K., Bras, R. L., Bhagavatula, C., and Choi, Y. Winogrande: An adversarial winograd schema challenge at scale, 2019. URL https://arxiv.org/abs/1907.10641.

Shah, J., Bikshandi, G., Zhang, Y., Thakkar, V., Ramani, P., and Dao, T. Flashattention-3: Fast and accurate attention with asynchrony and low-precision, 2024. URL https://arxiv.org/abs/2407.08608.

Soh, Y. J., Huang, H., Tian, Y., and Zhao, J. You only use reactive attention slice for long context retrieval, 2024. URL https://arxiv.org/abs/2409.13695.

Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding, 2023. URL https://arxiv.org/abs/2104.09864.

Sun, Y., Dong, L., Zhu, Y., Huang, S., Wang, W., Ma, S., Zhang, Q., Wang, J., and Wei, F. You only cache once: Decoder-decoder architectures for language models, 2024. URL https://arxiv.org/abs/2405.05254.

Talmor, A., Herzig, J., Lourie, N., and Berant, J. Commonsenseqa: A question answering challenge targeting commonsense knowledge, 2019. URL https://arxiv.org/abs/1811.00937.

Tay, Y., Bahri, D., Yang, L., Metzler, D., and Juan, D.-C. Sparse sinkhorn attention, 2020a. URL https://arxiv.org/abs/2002.11296.

Tay, Y., Dehghani, M., Abnar, S., Shen, Y., Bahri, D., Pham, P., Rao, J., Yang, L., Ruder, S., and Metzler, D. Long range arena: A benchmark for efficient transformers, 2020b. URL https://arxiv.org/abs/2011.04006.

Team, C. Chameleon: Mixed-modal early-fusion foundation models, 2024. URL https://arxiv.org/abs/2405.09818.

Team, G., Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., and Bobak Shahriari, e. a. Gemma 2: Improving open language models at a practical size, 2024a. URL https://arxiv.org/abs/2408.00118.

Team, J., Lenz, B., Arazi, A., Bergman, A., Manevich, A., Peleg, B., Aviram, B., Almagor, C., Fridman, C., Padnos, D., Gissin, D., Jannai, D., Muhlgay, D., Zimberg, D., Gerber, E. M., Dolev, E., Krakovsky, E., Safahi, E., Schwartz, E., Cohen, G., Shachaf, G., Rozenblum, H., Bata, H., Blass, I., Magar, I., Dalmedigos, I., Osin, J., Fadlon, J., Rozman, M., Danos, M., Gokhman, M., Zusman, M., Gidron, N., Ratner, N., Gat, N., Rozen, N., Fried, O., Leshno, O., Antverg, O., Abend, O., Lieber, O., Dagan, O., Cohavi, O., Alon, R., Belson, R., Cohen, R., Gilad, R., Glozman, R., Lev, S., Meirom, S., Delbari, T., Ness, T., Asida, T., Gal, T. B., Braude, T., Pumerantz, U., Cohen, Y., Belinkov, Y., Globerson, Y., Levy, Y. P., and Shoham, Y. Jamba-1.5: Hybrid transformer-mamba models at scale, 2024b. URL https://arxiv.org/abs/2408.12570.

Tworkowski, S., Staniszewski, K., Pacek, M., Wu, Y., Michalewski, H., and Miłoś, P. Focused transformer: Contrastive training for context scaling, 2023. URL https://arxiv.org/abs/2307.03170.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Wang, B. and Komatsuzaki, A. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

Wang, J., Ji, T., Wu, Y., Yan, H., Gui, T., Zhang, Q., Huang, X., and Wang, X. Length generalization of causal transformers without position encoding, 2024. URL https://arxiv.org/abs/2404.12224.

Xiao, G., Tian, Y., Chen, B., Han, S., and Lewis, M. Efficient streaming language models with attention sinks, 2024. URL https://arxiv.org/abs/2309.17453.

Xu, X., Ye, Q., and Ren, X. Stress-testing long-context language models with lifelong icl and task haystack, 2024. URL https://arxiv.org/abs/2407.16695.

Yang, A., Yang, B., Hui, B., Zheng, B., Yu, B., Zhou, C., Li, C., Li, C., Liu, D., Huang, F., Dong, G., Wei, H., Lin, H., Tang, J., Wang, J., Yang, J., Tu, J., Zhang, J., Ma, J., Yang, J., Xu, J., Zhou, J., Bai, J., He, J., Lin, J., Dang, K., Lu, K., Chen, K., Yang, K., Li, M., Xue, M., Ni, N., Zhang, P., Wang, P., Peng, R., Men, R., Gao, R., Lin, R., Wang, S., Bai, S., Tan, S., Zhu, T., Li, T., Liu, T., Ge, W., Deng, X., Zhou, X., Ren, X., Zhang, X., Wei, X., Ren, X., Liu, X., Fan, Y., Yao, Y., Zhang, Y., Wan, Y., Chu, Y., Liu, Y., Cui, Z., Zhang, Z., Guo, Z., and Fan, Z. Qwen2 technical report, 2024a. URL https://arxiv.org/abs/2407.10671.

Yang, A., Yang, J., Ibrahim, A., Xie, X., Tang, B., Sizov, G., Reizenstein, J., Park, J., and Huang, J. Context parallelism for scalable million-token inference, 2024b. URL https://arxiv.org/abs/2411.01783.

Ye, T., Dong, L., Xia, Y., Sun, Y., Zhu, Y., Huang, G., and Wei, F. Differential transformer, 2024. URL https://arxiv.org/abs/2410.05258.

Zaheer, M., Guruganesh, G., Dubey, A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., and Ahmed, A. Big bird: Transformers for longer sequences, 2021. URL https://arxiv.org/abs/2007.14062.

Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., and Choi, Y. Hellaswag: Can a machine really finish your sentence?, 2019. URL https://arxiv.org/abs/1905.07830.

Zhang, P., Liu, Z., Xiao, S., Shao, N., Ye, Q., and Dou, Z. Long context compression with activation beacon, 2024. URL https://arxiv.org/abs/2401.03462.

Zhang, Z., Sheng, Y., Zhou, T., Chen, T., Zheng, L., Cai, R., Song, Z., Tian, Y., Ré, C., Barrett, C., Wang, Z., and Chen, B. H_2o: Heavy-hitter oracle for efficient generative inference of large language models, 2023. URL https://arxiv.org/abs/2306.14048.

Zhong, W., Cui, R., Guo, Y., Liang, Y., Lu, S., Wang, Y., Saied, A., Chen, W., and Duan, N. Agieval: A human-centric benchmark for evaluating foundation models, 2023. URL https://arxiv.org/abs/2304.06364.

Üstün, A., Aryabumi, V., Yong, Z.-X., Ko, W.-Y., D'souza, D., Onilude, G., Bhandari, N., Singh, S., Ooi, H.-L., Kayid, A., Vargus, F., Blunsom, P., Longpre, S., Muennighoff, N., Fadaee, M., Kreutzer, J., and Hooker, S. Aya model: An instruction finetuned open-access multilingual language model, 2024. URL https://arxiv.org/abs/2402.07827.

A. Attention Distribution of all lengths

This table contains the attention distribution of RoPE and RNoPE variants over 8k, 32k and 128k sequence lengths.

Context Length	Model	NoPE Layers				RoPE Layers			
		begin	needle	context	qc	begin	needle	context	qc
8k	RoPE	-	-	-	-	0.3863	0.0328	0.3809	0.2000
	RNoPE-10k	0.3900	0.0952	0.4736	0.0412	0.1255	0.0102	0.5340	0.3302
	RNoPE-100k	0.3854	0.0932	0.4783	0.0430	0.2135	0.0136	0.4558	0.3171
	RNoPE-2M	0.3775	0.0902	0.4874	0.0449	0.2041	0.0126	0.4952	0.2881
	RNoPE-4M	0.4153	0.0546	0.5072	0.0229	0.1389	0.0136	0.6162	0.2313
	RNoPE-10k-swa	0.3830	0.1025	0.4702	0.0443	0.2040	0.0110	0.5938	0.1911
32k	RoPE	-	-	-	-	0.3541	0.0201	0.4343	0.1915
	RNoPE-10k	0.3275	0.0765	0.5672	0.0287	0.0049	0.0004	0.6805	0.3142
	RNoPE-100k	0.3263	0.0778	0.5633	0.0327	0.0241	0.0005	0.6782	0.2972
	RNoPE-2M	0.3250	0.0712	0.5735	0.0303	0.1111	0.0046	0.6233	0.2611
	RNoPE-4M	0.3486	0.0369	0.5981	0.0165	.0960	0.0039	0.6774	0.2227
	RNoPE-10k-swa	0.3303	.0742	0.5634	0.0321	-	-	-	-
128k	RoPE	-	-	-	-	0.3463	0.0010	0.4751	0.1776
	RNoPE-10k	0.2991	0.0444	0.6430	0.0135	.0000	0.0001	0.7230	0.2769
	RNoPE-100k	0.2454	0.0419	0.7016	0.0111	0.0001	0.0000	0.7749	0.2250
	RNoPE-2M	0.2600	0.0401	0.6836	0.0162	.0417	0.0008	0.7516	0.2059
	RNoPE-4M	0.2949	0.0307	0.6635	0.0109	0.0663	0.0022	0.7115	0.2230
	RNoPE-10k-swa	0.2760	0.0467	0.6615	0.0159	-	-	-	-

Table 8: Needles Attention Pattern: RoPE and RNoPE variants

B. Attention Distribution of RoPE and QK-Norm variants

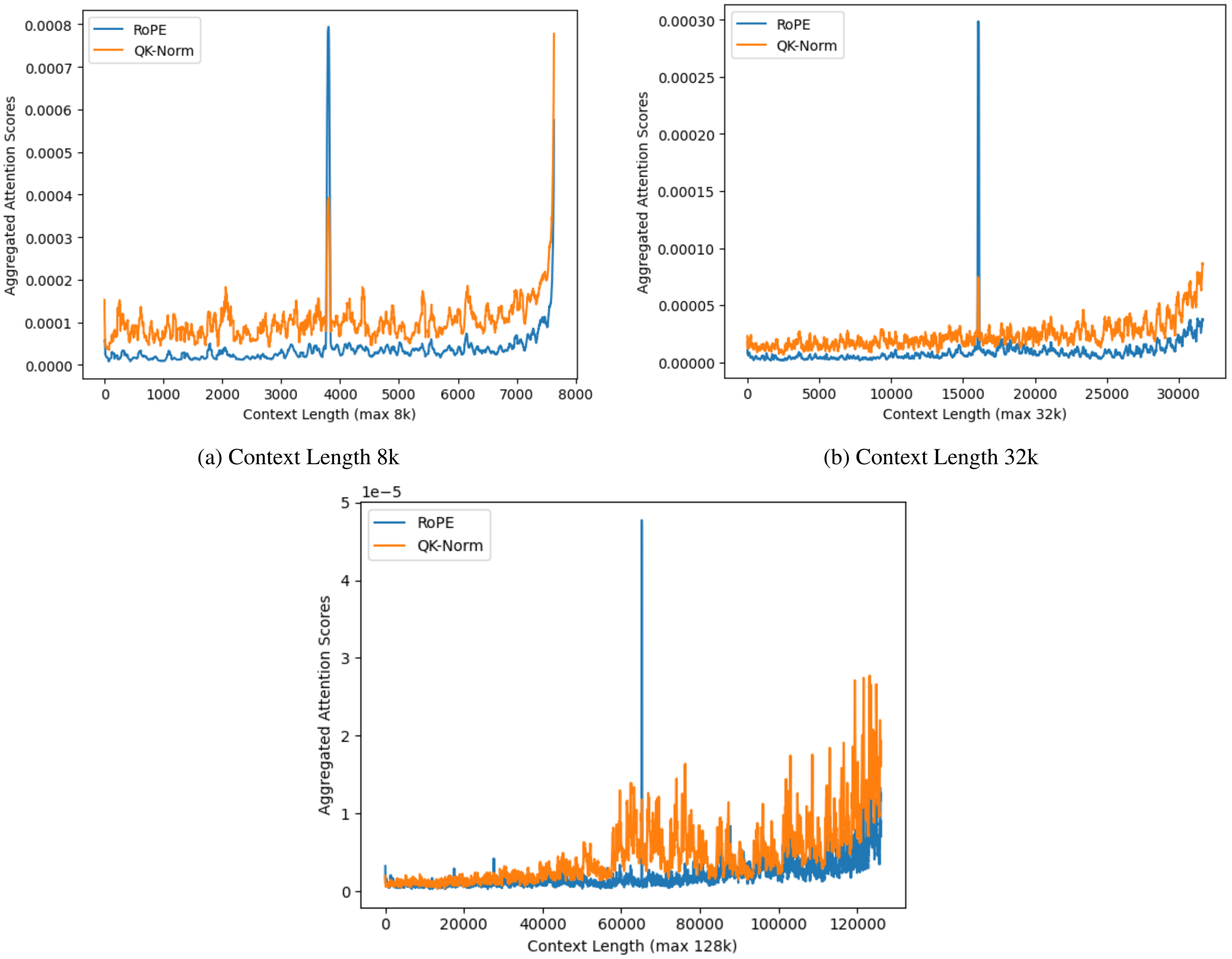
In this section, we present three plots comparing the attention distribution of RoPE and QK-Norm variants across sequence lengths of 8k, 32k, and 128k on needle samples, following the setup described in 2.2.2. Additionally, we provide the aggregated attention entropy for each variant to quantitatively validate the arguments made in 2.2.2.

To improve the clarity of the distribution plots, we preprocess the attention distribution array by removing the first 10 tokens and the last 3% of tokens at the end of each sequence. This step eliminates the disproportionate attention mass caused by the attention sink effect and the recency bias observed of RoPE, making the attention patterns more legible. We then compute a moving average with a window size of 100 tokens and average across all samples and layers to generate the final distributions.

Model	8k	32k	128k
RoPE	**6.02**	**6.95**	**7.62**
QK-Norm	10.71	12.46	14.14

Table 9: Entropy values of aggregated attention distribution

From Figure 3, we observe that the QK-Norm variant exhibits a lower spike on needle tokens but distributes more attention mass across context tokens. However, it also demonstrates a stronger recency bias compared to the RoPE variant. This characteristic results in a lower signal-to-noise ratio for the QK-Norm variant, which hampers its ability to effectively retrieve relevant information from long contexts. To further quantify this observation, we calculate the entropy values of the attention distributions for both variants, averaging across samples and layers at each sequence length. The results, listed in Table 9, show that the QK-Norm variant has significantly higher entropy values than the RoPE variant. This aligns with its weaker performance in long context retrieval tasks, as higher entropy reflects a more dispersed and less focused attention distribution.



(a) Context Length 8k

(b) Context Length 32k

(c) Context Length 128k

Figure 3: Attention Distribution Across Sequence lengths