applied sciences

MDPI

Article

Extending Context Window in Large Language Models with Segmented Base Adjustment for Rotary Position Embeddings

Rongsheng Li [1], Jin Xu [1], Zhixiong Cao [1], Hai-Tao Zheng [1,2,*] and Hong-Gee Kim [3]

[1] Shenzhen International Graduate School, Tsinghua University, Shenzhen 518071, China; lrs21@mails.tsinghua.edu.cn (R.L.); xj21@mails.tsinghua.edu.cn (J.X.); caozx21@mails.tsinghua.edu.cn (Z.C.)
[2] Pengcheng Laboratory, Shenzhen 518055, China
[3] School of Dentistry, Seoul National University, Seoul 03080, Republic of Korea; hgkim@snu.ac.kr
* Correspondence: zheng.haitao@sz.tsinghua.edu.cn

Abstract: In the realm of large language models (LLMs), extending the context window for long text processing is crucial for enhancing performance. This paper introduces SBA-RoPE (Segmented Base Adjustment for Rotary Position Embeddings), a novel approach designed to efficiently extend the context window by segmentally adjusting the base of rotary position embeddings (RoPE). Unlike existing methods, such as Position Interpolation (PI), NTK, and YaRN, SBA-RoPE modifies the base of RoPE across different dimensions, optimizing the encoding of positional information for extended sequences. Through experiments on the Pythia model, we demonstrate the effectiveness of SBA-RoPE in extending context windows, particularly for texts exceeding the original training lengths. We fine-tuned the Pythia-2.8B model on the PG-19 dataset and conducted passkey retrieval and perplexity (PPL) experiments on the Proof-pile dataset to evaluate model performance. Results show that SBA-RoPE maintains or improves model performance when extending the context window, especially on longer text sequences. Compared to other methods, SBA-RoPE exhibits superior or comparable performance across various lengths and tasks, highlighting its potential as an effective technique for context window extension in LLMs.

Keywords: large language models (LLMs); rotary position embeddings (RoPE); long text processing; context window extension; segmented base adjustment

Citation: Li, R.; Xu, J.; Cao, Z.; Zheng, H.-T.; Kim, H.-G. Extending Context Window in Large Language Models with Segmented Base Adjustment for Rotary Position Embeddings. *Appl. Sci.* **2024**, *14*, 3076. https://doi.org/10.3390/app14073076

Academic Editor: Douglas O'Shaughnessy

Received: 23 February 2024
Revised: 29 March 2024
Accepted: 30 March 2024
Published: 6 April 2024

1. Introduction

The Transformer model, since its inception by [1], has revolutionized the field of natural language processing (NLP) with its unparalleled ability to capture the intricacies of language through self-attention mechanisms. A pivotal feature of Transformer-based large language models (LLMs) is their capability for in-context learning (ICL) [2], enabling them to adapt to new tasks without explicit retraining, merely by conditioning on few-shot examples provided within their input context. This ability not only showcases the flexibility of Transformer-based models, but also underscores the importance of the context window—the span of tokens a model can consider at any given time. The size of this context window directly influences the number of examples that can be included for in-context learning, thereby impacting the model's performance on tasks requiring understanding and synthesis of information spread across longer texts.

The concept of the context window is foundational to understanding how Transformers operate. In essence, it determines the maximum scope of direct relationships and dependencies that the model can learn and leverage for prediction. A larger context window allows the inclusion of more examples for in-context learning, facilitating a richer understanding of context and enabling the model to make more informed predictions. Conversely, a smaller context window restricts the model's ability to capture long-range dependencies, potentially limiting its effectiveness in tasks that necessitate a comprehensive grasp of extended narratives or arguments.

Position encoding plays a crucial role in enabling Transformers to process sequential data. Unlike traditional sequential models such as RNN [3] and LSTM [4], Transformers do not inherently process data in sequence. Instead, they treat input as a set of tokens without any inherent order. Position encoding injects this missing sequence information, allowing the model to differentiate between the same word appearing in different positions within the text. The evolution from absolute to relative position encoding [5] has been a significant milestone in the development of Transformers, allowing models to better generalize to different sequence lengths and more effectively capture the relational dynamics within sequences.

While techniques such as ALiBi [6] and LeX [7] enable length extrapolation, they risk insufficient long-range dependencies due to their explicit long-range decay. Many LLMs, including LLaMA [8], GPT-NeoX [9], and PaLM [10], utilize RoPE [11] for their positional encoding. RoPE, without explicit long-range decay, is crucial for models targeting long contexts. It distinguishes between long and short ranges through varying frequencies of trigonometric functions, akin to hierarchical positional encoding, which is vital for long context processing. RoPE's direct application to Q and K, compatibility with Flash Attention, and scalability underscore the importance of finding an effective method to extend its context window.

Despite the advances in position encoding techniques, extending the effective context window of Transformers, especially for large language models (LLMs) such as GPT-NeoX, LLaMA, and PaLM, remains a significant challenge. These models, employing RoPE for their position encoding, must balance the need for long-context understanding with the computational and memory constraints inherent in processing large sequences.

Current approaches for extending the context window, including Positional Interpolation (PI) [12], Neural Tangent Kernel (NTK) [13], and YaRN [14], tackle various facets of this issue, yet each has its own drawbacks. PI, for example, compresses the space between tokens, potentially distorting the model's understanding of local context—a critical aspect for language models, given their reliance on local relationships for prediction accuracy. NTK, while offering a mathematical framework for extending context windows, can suffer from practical issues such as out-of-bounds rotation angles, leading to suboptimal extrapolation performance. YaRN attempts to mitigate some of these issues by partitioning the NTK approach, but it introduces additional complexity and necessitates fine-tuning of hyperparameters for each specific model.

To address these limitations, we propose Segmental Base Adjustment for RoPE (SBA-RoPE), a novel technique aimed at expanding the context window of pre-trained LLMs by strategically adjusting the base values used in RoPE. By selectively extrapolating high-frequency dimensions and interpolating those with maximum angles less than 2π, we treat length extrapolation as a prediction-stage Out-Of-Distribution (OOD) problem. RoPE allocates different angles to different dimensions, with some high-frequency dimensions having fully learned all angles within 0 to 2π. Extrapolating these dimensions does not degrade performance, as these angles have been thoroughly trained during pre-training. However, some low-frequency dimensions have only learned partial angles within 0 to 2π, making them unable to extrapolate on longer texts and only able to interpolate. Thus, due to the periodic nature of trigonometric functions, even with high-frequency dimensions extrapolated, the thoroughly trained angles within 0 to 2π during pre-training do not cause a perplexity explosion problem. Meanwhile, by selecting interpolation for low-frequency dimensions, OOD is avoided for these dimensions. This process is illustrated in Figure 1. SBA-RoPE facilitates an efficient extension of the context window with minimal fine-tuning. This method not only preserves the model's performance for tasks within the original context window, but also enhances its adaptability to tasks that demand longer contexts.

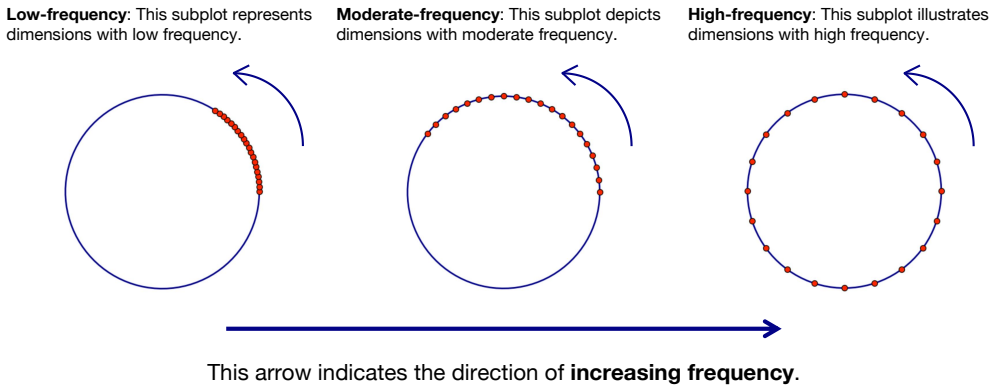

Figure 1. Schematic representation of the angles learned by high-frequency and low-frequency dimensions during pre-training in SBA-RoPE. The subplots on the (**left**) and (**middle**) depict low-frequency and moderate-frequency dimensions, respectively, which did not cover the entire range of 0 to 2π during pre-training, thus necessitating interpolation rather than extrapolation. The subplot on the (**right**) represents high-frequency dimensions, which thoroughly learned the entire range of 0 to 2π during pre-training, enabling extrapolation.

Our contributions are as follows:

- We introduce SBA-RoPE, a novel method for extending the context window of LLMs by segmentally adjusting the base of Rotary Position Embeddings, requiring only minimal fine-tuning steps.
- For tasks within the original context window, our method minimally impacts model performance, showing minimal degradation compared to the original Pythia-2.8B model based on the GPT-NeoX architecture.
- For tasks in the extended context window, our method achieves comparable or superior performance on passkey and perplexity tasks, indicating our model's ability to generalize to longer lengths without sacrificing performance.

2. Backgrounds and Methods

2.1. Background: Rotary Position Embedding

In transformer models, positional information needs to be provided in some way, which is typically achieved through positional encoding. The positional encoding used in models such as GPT-NeoX and LLaMA is the Rotary Position Embedding (RoPE). Given a position index $m \in [0, c)$ and an embedding vector $x := [x_0, x_1, \ldots, x_{d-1}]^T$, where d is the dimension of the attention heads, RoPE defines the complex function as Equation (1):

$$\mathbf{f}(\mathbf{x}, m) = [(x_0 + ix_1)e^{im\theta_0}, (x_2 + ix_3)e^{im\theta_1}, \ldots, (x_{d-2} + ix_{d-1})e^{im\theta_{d/2-1}}]^T \tag{1}$$

where $i := \sqrt{-1}$ denotes the imaginary unit and $\theta_j = base^{-2j/d}$, where in RoPE, *base* is typically set to 10,000. With RoPE, the calculation of self-attention scores is performed as Equation (2):

$$\begin{aligned} a(m,n) &= \mathrm{Re}\langle \mathbf{f}(\mathbf{q}, m), \mathbf{f}(\mathbf{k}, n)\rangle \\ &= \mathrm{Re}\left[\sum_{j=0}^{d/2-1} (q_{2j} + q_{2j+1})(k_{2j} - k_{2j+1})e^{(m-n)\theta_j}\right] \\ &= \sum_{j=0}^{d/2-1} (q_{2j}k_{2j} + q_{2j+1}k_{2j+1})\cos((m-n)\theta_j) + (q_{2j}k_{2j+1} - q_{2j+1}k_{2j})\sin((m-n)\theta_j) \\ &=: a(m-n) \end{aligned} \tag{2}$$

The value of $a(m,n)$ depends only on the relative position $m-n$, where $\mathbf{q}$ and $\mathbf{k}$ are query and key vectors. In Cartesian coordinates, RoPE can be written as Equation (3):

$$\mathbf{f}(\mathbf{x}_m, m, \theta_j) = \begin{pmatrix} \cos m\theta_1 & -\sin m\theta_1 & 0 & 0 & \cdots & 0 & 0 \\ \sin m\theta_1 & \cos m\theta_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos m\theta_2 & -\sin m\theta_2 & \cdots & 0 & 0 \\ 0 & 0 & \sin m\theta_2 & \cos m\theta_2 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & \cos m\theta_{d/2-1} & -\sin m\theta_{d/2-1} \\ 0 & 0 & 0 & 0 & \cdots & \sin m\theta_{d/2-1} & \cos m\theta_{d/2-1} \end{pmatrix} \mathbf{x}_m \tag{3}$$

It should be noted that this transformation of RoPE is equivalent to rotating $[x_{2i}, x_{2(i+1)}]$ (where $i \in [0, d/2-1]$) in complex space. This is why this type of positional encoding is named Rotary Position Encoding, with the corresponding rotation angles being the focus of extrapolation research on out-of-distribution in longer texts.

2.2. Positional Interpolation

Since Large Language Models (LLMs) are typically pre-trained with fixed context lengths, such as 2048 for LLaMA, 4096 for LLaMA-2, and 2048 for Pythia models (based on GPT-NeoX), it is a natural idea to extend the context length by fine-tuning with a small amount of data. The first method to extend the existing LLMs using fine-tuning was Position Interpolation (PI) [12], where it was found that directly extrapolating the angles of RoPE through fine-tuning did not yield satisfactory results. However, the effect of fine-tuning with Position Interpolation showed promising results in extending the context window. They modified the RoPE method as Equation (4):

$$\tilde{\mathbf{f}}(\mathbf{x}, m, \theta_j) = \mathbf{f}(\mathbf{x}, \frac{mL}{L'}, \theta_j) \tag{4}$$

where L represents the maximum context length during pre-training and L' is the larger context length used during fine-tuning. They empirically demonstrated that decent results could be achieved on the new context length with just 1000 steps of fine-tuning.

2.3. NTK Interpolation

Although Position Interpolation has shown some effectiveness in extending the context window, models fine-tuned using this method suffer performance degradation within the original context window [14]. The reason for this performance degradation is apparent: while position interpolation avoids the issue of RoPE rotation angle out-of-bounds at large indices, it compresses the distance between tokens, severely disrupting the model's local resolution. Given that language modeling heavily relies on local relationships, disrupting local relationships inevitably leads to inaccurate predictions.

NTK [13] addresses the "local distortion" problem caused by stretching all dimensions equally in PI. They derive NTK Interpolation using Neural Tangent Kernel (NTK) theory as Equations (5) and (6):

$$base'^{\frac{d-2}{d}} = s \cdot base^{\frac{d-2}{d}} \tag{5}$$

$$base' = base \cdot s^{\frac{d}{d-2}} \tag{6}$$

where $s = \frac{L'}{L}$ represents the scaling factor, $base$ is the value used in RoPE during pre-training, $base'$ is the new corresponding value used during fine-tuning, and d denotes the embedding dimension of the attention head. This adjustment avoids uniformly stretching all dimensions, instead dispersing interpolation pressure across multiple dimensions. After this adjustment, the lowest-frequency dimensions are scaled similarly to PI, while the highest-frequency dimensions remain unchanged (i.e., unscaled). It is noteworthy that this adjustment offers an ability to extend the context window without the need for fine-tuning. However, when used for fine-tuning, this method may lead to out-of-bounds values in

some dimensions compared to the maximum rotation angles during pre-training, resulting in performance degradation.

YaRN [14] proposes segment-wise adjustment of θ values in RoPE based on the period of different dimensions for interpolation. This method addresses the issue of out-of-bounds values in some dimensions in NTK and achieves better results in extending the context window compared to NTK. However, this method introduces additional hyperparameters that need to be adjusted for different models to achieve particularly good results. Additionally, we argue that interpolation for certain high-frequency dimensions may lead to performance degradation in these dimensions, as we believe these dimensions have fully learned the $[0, 2\pi]$ angle during pre-training and do not require interpolation.

2.4. Our Proposal: SBA-RoPE

By observing the out-of-bounds values in certain dimensions of the NTK Interpolation, it becomes evident that for low-frequency dimensions, the maximum rotation angles learned during the pre-training process do not exceed 2π. These dimensions carry absolute position information, as the rotation angle corresponding to each position is unique. Extrapolating these angles would lead to significantly increased model perplexity, since the model has not been trained on these extrapolated angles. On the other hand, high-frequency dimensions have been thoroughly trained within the $[0, 2\pi]$ range, and the sine and cosine values of these angles reoccur in RoPE due to the periodicity of trigonometric functions. Thus, we can simply extrapolate these angles, as they carry relative position information.

To delineate the dimensions requiring interpolation from those needing extrapolation, we define the following notation as Equation (7) for the rotation angle at position pos and dimension dim:

$$\theta_{pos,dim,base} = pos \cdot base^{\frac{-2 \cdot dim}{d}} \tag{7}$$

where $pos \in [0, L-1]$ represents the position index corresponding to the current rotation angle, $dim = 0, 1, \ldots, d/2-1$ represents the dimension corresponding to the current rotation angle, and d represents the embedding dimension of the attention head. Based on our discussion, we need to identify the smallest dimension dim' such that it is just less than 2π; this dimension marks the boundary as Equation (8) between extrapolation and interpolation:

$$dim' = \min\{dim \in [0, 1, \ldots, d/2-1], \theta_{L-1,dim,base} < 2\pi\} \tag{8}$$

where $base$ and L are, respectively, the $base$ and the maximum context window used in RoPE during LLM pre-training. For $dim > dim'$, we employ an interpolation method similar to NTK, ensuring the scaling at dimension dim' matches that of the Positional Interpolation (PI) method. The $base'$ for these low-frequency dimensions dim can be calculated under the new fine-tuning length L' as Equations (9)–(11):

$$\theta_{L'-1,dim',base'} = \theta_{L-1,dim',base} \tag{9}$$

$$(L'-1) \cdot base'^{\frac{-2 \cdot dim'}{d}} = (L-1) \cdot base^{\frac{-2 \cdot dim'}{d}} \tag{10}$$

$$base' = base \cdot \left(\frac{L'-1}{L-1}\right)^{\frac{d}{2 \cdot dim'}} \tag{11}$$

This calculation determines the $base$ used for interpolating low-frequency dimensions. Consequently, we can express the rotation angles in Segmental Base Adjustment for RoPE (SBA-RoPE) as Equation (12):

$$\theta = \begin{cases} \theta_{pos,dim,base}, & \text{if } dim < dim' \\ \theta_{pos,dim,base'}, & \text{if } dim \geq dim'. \end{cases} \tag{12}$$

Through our derivation, we have demonstrated that this approach yields rotation angles that, under the new fine-tuning length L', do not produce out-of-bounds values

for high-frequency dimensions $dim < dim'$ nor low-frequency dimensions $dim > dim'$, maintaining consistency with the PI method. For high-frequency dimensions, we directly perform extrapolation to avoid the loss of high-frequency information, as seen with PI. For low-frequency dimensions, we adopt a method similar to NTK, distributing the interpolation load across all low-frequency dimensions.

3. Experiments and Results

We demonstrate that SBA-RoPE effectively extends the context window of Large Language Models (LLMs) with just 10,000 fine-tuning data points, a quantity negligible compared to the data volume used in model pre-training stages. We evaluated the model's perplexity on long texts and conducted passkey retrieval experiments, proving that SBA-RoPE surpasses all previous methods for extending the context window.

3.1. Setup

Baselines. We compared SBA-RoPE against three methods: Positional Interpolation (PI), Neural Tangent Kernel (NTK), and YaRN. Additionally, to assess performance within the original context window of LLMs, we included baselines of LLMs without fine-tuning.

Considering computational costs, we selected Pythia-2.8b as our fine-tuning starting point. This model size is sufficient to highlight the performance differences between methods. Pythia [15] includes LLMs ranging from 14M to 12B parameters, utilizing the same architecture as GPT-NeoX [9]. It was pre-trained on the Pile [16] dataset by EleutherAI (https://www.eleuther.ai, accessed on 1 March 2024) for research into the interpretability analysis and scaling laws of LLMs, aiming to understand how knowledge develops and evolves during autoregressive transformer training. The model checkpoint used in our experiments is available on Hugging Face (https://huggingface.co/EleutherAI/pythia-2.8b accessed on 1 March 2024). Other than adjusting the implementation of position embeddings using various methods, we also employed Memory-Efficient Attention from xFormers [17], provided by PyTorch [18], to replace the native attention mechanism of GPT-NeoX, thereby accelerating the training and inference processes on NVIDIA V100 GPU. Apart from these modifications, we did not modify the GPT-NeoX model architecture in any way.

Training. We fine-tuned all model variants using the next-token prediction objective and cross-entropy [19] loss function, combined with various methods. We employed the AdamW [20] optimizer, with β_1 and β_2 set to 0.9 and 0.95, respectively. For the scheduler, we used a linear warmup of 60 steps. The maximum learning rate was set to 2×10^{-5}, with weight decay set to zero. Utilizing eight NVIDIA V100 GPUs, we set the global batch size to 8 and fine-tuned each model variant for 1250 steps. FP16 mixed precision training [21] was enabled. All models were trained using PyTorch and DeepSpeed Zero-3 [22]. For the training dataset, we fine-tuned models using data from PG-19 [23], truncating texts to lengths of 4 k tokens and appending BOS and EOS tokens at the beginning and end, respectively. We fine-tuned two variants of each method for scaling factors $s = 2$ and $s = 4$. Given that the original context window of the Pythia model is 2048, the fine-tuned models have expanded context windows of 4096 and 8192, respectively.

3.2. Long Sequence Language Modeling

To evaluate performance in long sequence language modeling, we utilized the Proof-pile [24] dataset, which comprises numerous long sequence texts, specifically employing its test split for our analysis. We adopted the sliding window technique [6] to assess perplexity across various context window sizes, setting the stride S to 256.

Initially, we assessed how model variants, fine-tuned using different methods, performed as the context window size increased. Following the approach used by YaRN, we selected 10 random samples from the Proof-pile dataset, each containing at least 10 k tokens. For the scaling factor $s = 2$, we evaluated the perplexity for sequence lengths from 1 k to 5 k tokens, in 1 k token steps.

Table 1 presents a comparison of perplexity for the Pythia model, expanded from an original context window of 2048 to 4096, utilizing the original model, PI, NTK, YaRN, and SBA-RoPE. From the experimental results, it is evident that, initially, within the original context lengths (1 k and 2 k), all models fine-tuned with various context window extension methods experienced an increase in perplexity to varying degrees. The PI method saw the most significant increase within this range, likely due to the equal stretching of all dimensions, resulting in the loss of high-frequency information.

In contrast, the model variants fine-tuned with our method exhibited the smallest increase in perplexity, indicating minimal performance degradation. This minimal increase is attributed to our method's direct extrapolation of high-frequency dimensions, preventing the loss of this part of the pre-trained information. In the extended context window (3 k to 4 k), the original model's perplexity significantly increased, indicating that the original model became impractical for extended windows. Our method achieved lower perplexity in the extended window compared to all other methods. Furthermore, at the 5 k window, which exceeds the extended context window (4 k), our method still maintained the lowest value, signifying superior extrapolation performance.

Table 1. Perplexity comparison across expanded context window sizes, contrasting the original Pythia-2.8b model with variants fine-tuned via different methods. The scaling factor for all variants is $s = 2$, doubling the original context window size from 2 k to 4 k tokens. Evaluations of perplexity range from 1 k to 5 k tokens, with increments of 1 k. The lowest and second-lowest perplexity (PPL) values at each length are highlighted in **bold** for the lowest and underlined for the second-lowest.

Method	1024	2048	3072	4096	5120
Original	**5.617**	**4.785**	25.844	79.499	146.776
PI	6.185	5.227	4.756	4.622	10.543
NTK	6.064	5.134	4.678	4.595	11.173
YaRN	6.061	5.134	4.678	4.552	11.016
SBA-RoPE	6.036	5.016	**4.596**	**4.363**	**10.233**

We further increased the scaling factor s to 4, applying fine-tuning to the original model through various methods. The outcomes are presented in Table 2. Despite the maximum length of the fine-tuning data being 4 k, adjusting the scaling factor s to 4 allows us to infer that the model's maximum context length has now been extended to 8 k. This extension is noteworthy because, although the model has never been exposed to context lengths between 4 k and 8 k, it still demonstrates a degree of transfer learning capability.

Table 2. Perplexity comparison for the Pythia-2.8b model and its variants fine-tuned with a scaling factor of $s = 4$, extending the context window to 8 k and beyond. The evaluated perplexities range from 1 k to 10 k tokens, in increments of 1 k. Values are adjusted to highlight the lowest (in **bold**) and second-lowest (in underlined) perplexity at each token length.

Model	1024	2048	3072	4096	5120	6144	7168	8192	9216	10,240
Original	**5.617**	**4.785**	25.844	79.499	146.776	247.862	397.569	582.819	767.523	952.580
PI	6.731	5.682	5.155	5.007	5.081	4.944	4.719	4.647	5.761	10.304
NTK	6.070	5.148	4.691	4.554	4.629	4.687	9.119	18.220	30.109	44.844
YaRN	6.103	5.166	4.709	4.572	4.652	4.532	**4.353**	4.577	**5.333**	8.801
SBA-RoPE	6.047	5.148	**4.493**	**4.455**	**4.538**	**4.418**	4.424	**4.506**	5.381	**8.037**

Data presented in Table 2 show performance within the 1 k to 4 k context window that is similar to the behavior observed with a scaling factor $s = 2$. Notably, even within the 7 k to 8 k length, which fal ls within the extended context window of NTK, this method exhibits significantly higher perplexity compared to others. This higher perplexity can be attributed to the NTK method's adjusted rotation angles producing out-of-bounds values.

Conversely, our method maintains superior performance in the previously unseen 5 k to 10 k window range, consistently demonstrating the most favorable outcomes.

3.3. Passkey Retrieval

To investigate the effective context window size of models after extension—that is, the maximum distance of tokens that can be effectively attended to during the inference process—we adhered to the passkey retrieval task as defined by [25]. In this task, the model is required to recover a hidden five-digit passkey embedded within a context of largely nonsensical text. The specific format of the prompt used for this task is detailed in Figure 2.

```
There is an important info hidden inside a lot of irrelevant text. Find it and
memorize them. I will quiz you about the important information there.
The grass is green. The sky is blue. The sun is yellow. Here we go. There and
back again. (repeat X times)
The pass key is 12345. Remember it. 12345 is the pass key.
The grass is green. The sky is blue. The sun is yellow. Here we go. There and
back again. (repeat Y times)
What is the pass key? The pass key is
```

Figure 2. We adopt the identical prompt structure for passkey recovery as suggested by [25]. In this setup, the specific passkey 12345 is substituted with randomly generated five-digit numerals for the evaluation phase.

To evaluate the models, we conducted the passkey retrieval task 20 times for each model variant, positioning the key as close to the beginning of the context as possible to more accurately reflect the model's capability to attend to the longest distance. The accuracy of models fine-tuned with a scaling factor $s = 2$ on the passkey retrieval task is shown in Table 3. It is observed that all fine-tuned model variants exhibit high accuracy in the extended window. Notably, PI outperforms NTK and YaRN across all context window sizes on this task, suggesting that evenly stretching all embedding dimensions, despite causing "local distortion" do not significantly impact the dependency relationships between local tokens in the context of passkey retrieval. Our SBA-RoPE achieves the best results in all context windows except for the 3 k window, by combining the advantages of PI's non-exceeding bounds and NTK's distribution of interpolation stress across all dimensions.

Table 3. Accuracy of passkey retrieval across different context window sizes for the original Pythia-2.8b model and its variants fine-tuned with a scaling factor of $s = 2$. Accuracy is measured by the model's ability to correctly retrieve the hidden passkey within various context lengths. The highest and second-highest accuracy values at each length are highlighted in **bold** for the highest and underlined for the second-highest.

Method	1024	2048	3072	4096	5120
Original	**1.0**	**1.0**	0.00	0.00	0.00
PI	0.95	0.85	**0.65**	**0.85**	0.00
NTK	0.90	0.75	0.40	0.60	0.00
YaRN	0.70	0.80	0.50	0.55	0.00
SBA-RoPE	0.95	0.90	0.60	**0.85**	0.00

To assess the extended capabilities of the models, we further tested them under the passkey retrieval task with the scaling factor increased to $s = 4$, representing an expanded context window up to 8 k tokens. The outcomes, as documented in Table 4, reveal that the models fine-tuned with this larger scaling factor continue to perform with notable accuracy across extended context lengths. Particularly, the PI method and our SBA-RoPE exhibit remarkably consistent performance, even at the higher context ranges of 5 k to 8 k tokens, underscoring the effectiveness of our approach in managing extended contexts. SBA-RoPE, especially, demonstrates superior adaptability and accuracy, effectively leveraging its

hybrid strategy to maintain high retrieval accuracy, a testament to its robust extrapolation capabilities in previously unseen context lengths.

Table 4. Accuracy of passkey retrieval across different context window sizes for the original Pythia-2.8b model and its variants fine-tuned with a scaling factor of $s = 4$. The highest and second-highest accuracy values at each length are highlighted in **bold** for the highest and underlined for the second-highest.

Method	1024	2048	3072	4096	5120	6144	7168	8192	9216	10240
Original	**1.0**	**1.0**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PI	0.95	**1.0**	**0.90**	**0.95**	**0.95**	**0.95**	0.95	**1.0**	0.00	0.00
NTK	0.85	0.85	0.70	0.60	0.90	0.00	0.00	0.00	0.00	0.00
YaRN	0.80	0.65	0.85	0.80	0.75	0.55	0.60	0.00	0.00	0.00
SBA-RoPE	0.95	0.95	**0.90**	**0.95**	**0.95**	**0.95**	**1.0**	**1.0**	**0.05**	0.00

4. Discussions of the Results

The experiments conducted for SBA-RoPE have demonstrated its effectiveness in extending the context window of LLMs with a relatively small fine-tuning dataset. The comparison of SBA-RoPE with other methods, such as PI, NTK, and YaRN, across different metrics and tasks has provided a comprehensive understanding of its performance and advantages.

4.1. Perplexity Analysis

The perplexity measurements, as presented in Tables 1 and 2, illustrate the capability of SBA-RoPE to maintain lower perplexity across extended context windows, surpassing the baseline methods. This is particularly significant in the context of long sequence language modeling, where maintaining coherence over longer spans of text is crucial. The minimal increase in perplexity within the original context lengths for models fine-tuned with SBA-RoPE suggests that our method successfully preserves the model's original performance while effectively extending its context window. This indicates a balanced approach to extrapolating high-frequency dimensions, which is critical for minimizing the loss of pre-trained information.

Furthermore, the extended context window experiments, especially with the scaling factor $s = 4$, highlight SBA-RoPE's superior extrapolation performance. Despite the models never being exposed to context lengths between 4 k and 8 k during training, SBA-RoPE demonstrates a robust transfer learning capability, effectively leveraging learned representations to adapt to and perform within these expanded context windows. This underscores the potential of SBA-RoPE in enhancing model flexibility and generalization across various context lengths, a key advantage for applications requiring comprehension of long documents or conversations.

4.2. Passkey Retrieval Performance

The passkey retrieval task results, as shown in Tables 3 and 4, further validate the effectiveness of SBA-RoPE in managing extended context windows. The high accuracy of SBA-RoPE in this task across all evaluated context windows, especially with the scaling factor $s = 4$, demonstrates its capability to attend to tokens over long distances without significant loss of performance. This is indicative of SBA-RoPE's efficient handling of the extended context, combining the advantages of PI's non-exceeding bounds and NTK's distribution of interpolation stress across dimensions.

Notably, the performance of SBA-RoPE in the previously unseen 5 k to 10 k window range with high retrieval accuracy is a testament to its robust extrapolation capabilities. This suggests that SBA-RoPE not only effectively extends the context window, but also ensures that the model can maintain functional coherence and understanding over these longer spans, a critical requirement for tasks involving detailed comprehension and retention of information across large text bodies.

4.3. Applications and Implementation

Extending the context window of large language models offers multiple benefits for document summarization or long-document question-answering tasks. These benefits primarily stem from the ability to process longer texts, thereby enhancing the model's understanding and generation quality. Here are some specific benefits:

- Improved Document Understanding: By expanding the context window, models can process and understand longer texts at once. This means that when summarizing documents or answering questions related to long documents, the model can capture the content and structure of the document more comprehensively, thus improving the accuracy of understanding.
- Reduced Information Loss: With long documents, smaller context windows may lead to the loss of important information since the model cannot view the entire document at once. Expanding the context window can reduce this information loss, allowing the model to consider more relevant information when generating summaries or answering questions.
- Enhanced Accuracy and Relevance of Answers: For long-document question-answering tasks, being able to consider more information within the document can help the model generate more accurate and relevant answers. This is because the model has a greater chance of finding the exact answer to a question within the entire document, rather than relying on partial information for inference.
- Improved Handling of Long-Distance Dependencies: In long documents, there may be long-distance dependencies between certain points of information. Expanding the context window allows the model to better capture these dependencies, thus providing more coherent and accurate information when generating summaries or answering questions.
- Enhanced Comprehensive Understanding: In complex document summarization tasks, the model needs to understand not just individual sentences or paragraphs but the main theme and structure of the entire document. A larger context window enables the model to perform this comprehensive understanding over a broader range, thereby generating higher quality summaries.
- Optimized Information Integration for Long Documents: In long-document question-answering or summarization, it is necessary to effectively integrate information scattered across different parts. A larger context window allows the model to identify and integrate this information over a wider range, making the final output more accurate and comprehensive.

Furthermore, adding the SBA-RoPE method to existing models based on rotational position embedding, such as Llama, Llama-2, and GPT-NeoX, is very easy, requiring only a few lines of code. This ease of modification is because the SBA-RoPE method does not require changes to the calculation of attention scores but only modifications to the model's embedding-related code. We will release the source code after the paper is accepted.

5. Conclusions

In this study, we introduced SBA-RoPE, a novel technique designed to extend the context window of pre-trained LLMs by strategically adjusting the base values used in RoPE. Our approach, which selectively extrapolates high-frequency dimensions and interpolates those with maximum angles less than 2π, conceptualizes length extrapolation as a prediction-stage Out-Of-Distribution (OOD) problem. This method leverage the inherent properties of RoPE to maintain performance integrity across extended contexts while efficiently addressing the challenges posed by OOD in low-frequency dimensions.

The empirical results presented in this paper validate the effectiveness of SBA-RoPE, demonstrating its superiority over existing methods, such as PI, NTK, and YaRN, across various context window sizes. Notably, SBA-RoPE's ability to maintain high accuracy and low

perplexity in extended contexts—even those previously unseen during training—highlights its robustness and adaptability.

Looking forward, the promising results achieved by SBA-RoPE open up new avenues for research into extending the capabilities of LLMs. Future studies could explore the integration of SBA-RoPE with other model architectures, delve into the potential of larger context windows, and investigate its applicability to few-shot learning scenarios. Such research could further unravel the complexities of model performance across varying context lengths and contribute to the development of more sophisticated and versatile language models.

Author Contributions: Conceptualization, R.L.; methodology, R.L. and Z.C.; software, R.L. and J.X.; validation, R.L. and J.X.; formal analysis, R.L.; investigation, R.L. and Z.C.; resources, R.L. and J.X.; data curation, R.L.; writing—original draft preparation, R.L. and Z.C.; writing—review and editing, R.L., J.X., H.-T.Z. and H.-G.K.; visualization, R.L.; supervision, H.-T.Z.; project administration, H.-T.Z. and H.-G.K.; funding acquisition, H.-T.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundation of China (Grant No.62276154), the Research Center for Computer Network (Shenzhen) Ministry of Education, the Natural Science Foundation of Guangdong Province (Grant No. 2023A1515012914), the Basic Research Fund of Shenzhen City (Grant No. JCYJ20210324120012033 and JSGG20210802154402007), the Major Key Project of PCL for Experiments and Applications (PCL2021A06), and the Overseas Cooperation Research Fund of Tsinghua Shenzhen International Graduate School (HW2021008).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
2. Dong, Q.; Li, L.; Dai, D.; Zheng, C.; Wu, Z.; Chang, B.; Sun, X.; Xu, J.; Sui, Z. A survey for in-context learning. *arXiv* **2022**, arXiv:2301.00234.
3. Medsker, L.R.; Jain, L. Recurrent neural networks. *Des. Appl.* **2001**, *5*, 2.
4. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
5. Shaw, P.; Uszkoreit, J.; Vaswani, A. 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), New Orleans, LA, USA, 1–6 June 2018; pp. 464–468.
6. Press, O.; Smith, N.; Lewis, M. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation. In Proceedings of the International Conference on Learning Representations, Virtual, 25–29 April 2022.
7. Sun, Y.; Dong, L.; Patra, B.; Ma, S.; Huang, S.; Benhaim, A.; Chaudhary, V.; Song, X.; Wei, F. A Length-Extrapolatable Transformer. *arXiv* **2022**, arXiv:2212.10554.
8. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. Llama: Open and efficient foundation language models. *arXiv* **2023**, arXiv:2302.13971.
9. Black, S.; Biderman, S.; Hallahan, E.; Anthony, Q.; Gao, L.; Golding, L.; He, H.; Leahy, C.; McDonell, K.; Phang, J.; et al. Gpt-neox-20b: An open-source autoregressive language model. *arXiv* **2022**, arXiv:2204.06745.
10. Anil, R.; Dai, A.M.; Firat, O.; Johnson, M.; Lepikhin, D.; Passos, A.; Shakeri, S.; Taropa, E.; Bailey, P.; Chen, Z.; et al. Palm 2 technical report. *arXiv* **2023**, arXiv:2305.10403.
11. Su, J.; Lu, Y.; Pan, S.; Wen, B.; Liu, Y. RoFormer: Enhanced Transformer with Rotary Position Embedding. *arXiv* **2021**, arXiv:2104.09864.
12. Chen, S.; Wong, S.; Chen, L.; Tian, Y. Extending context window of large language models via positional interpolation. *arXiv* **2023**, arXiv:2306.15595.
13. bloc97. NTK-Aware Scaled RoPE Allows LLaMA Models to Have Extended (8k+) Context Size without any Fine-Tuning and Minimal Perplexity Degradation. Available online: https://www.reddit.com/r/LocalLLaMA/comments/14lz7j5/ntkaware_scaled_rope_allows_llama_models_to_have (accessed on 1 March 2024).

14. Peng, B.; Quesnelle, J.; Fan, H.; Shippole, E. Yarn: Efficient context window extension of large language models. *arXiv* **2023**, arXiv:2309.00071.
15. Biderman, S.; Schoelkopf, H.; Anthony, Q.G.; Bradley, H.; O'Brien, K.; Hallahan, E.; Khan, M.A.; Purohit, S.; Prashanth, U.S.; Raff, E.; et al. Pythia: A suite for analyzing large language models across training and scaling. In Proceedings of the International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; pp. 2397–2430.
16. Gao, L.; Biderman, S.; Black, S.; Golding, L.; Hoppe, T.; Foster, C.; Phang, J.; He, H.; Thite, A.; Nabeshima, N.; et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv* **2020**, arXiv:2101.00027.
17. Lefaudeux, B.; Massa, F.; Liskovich, D.; Xiong, W.; Caggiano, V.; Naren, S.; Xu, M.; Hu, J.; Tintore, M.; Zhang, S.; et al. xFormers: A Modular and Hackable Transformer Modelling Library. 2022. Available online: https://github.com/facebookresearch/xformers (accessed on 1 March 2024).
18. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Curran Associates Inc.: Red Hook, NY, USA, 2019.
19. De Boer, P.T.; Kroese, D.P.; Mannor, S.; Rubinstein, R.Y. A tutorial on the cross-entropy method. *Ann. Oper. Res.* **2005**, *134*, 19–67. [CrossRef]
20. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.
21. Micikevicius, P.; Narang, S.; Alben, J.; Diamos, G.; Elsen, E.; Garcia, D.; Ginsburg, B.; Houston, M.; Kuchaiev, O.; Venkatesh, G.; et al. Mixed precision training. *arXiv* **2017**, arXiv:1710.03740.
22. Rasley, J.; Rajbhandari, S.; Ruwase, O.; He, Y. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event , 6–10 July 2020; pp. 3505–3506.
23. Rae, J.W.; Potapenko, A.; Jayakumar, S.M.; Hillier, C.; Lillicrap, T.P. Compressive Transformers for Long-Range Sequence Modelling. In Proceedings of the International Conference on Learning Representations, Virtually, 26 April–1 May 2020.
24. Azerbayev, Z.; Ayers, E.; Piotrowski, B. Proof-Pile. Available online: https://github.com/zhangir-azerbayev/proof-pile (accessed on 1 March 2024).
25. Mohtashami, A.; Jaggi, M. Landmark Attention: Random-Access Infinite Context Length for Transformers. *arXiv* **2023**, arXiv:2305.16300.