3D-RPE: Enhancing Long-Context Modeling Through 3D Rotary Position Encoding

Xindian Ma[1], **Wenyuan Liu**[1], **Peng Zhang**[1*], **Nan Xu**[2]

[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]Beijing Wenge Technology Co.
xindianma@tju.edu.cn, 1wy2020@tju.edu.cn, pzhang@tju.edu.cn, xunan2015@ia.ac.cn

Abstract

An essential component in Large Language Models (LLMs) is Rotary Position Encoding (RoPE) , which efficiently manages positional dependencies in long-context modeling. However, when the number of input tokens surpasses the pretrained capacity of LLMs, their ability to process and generate text is markedly weakened. Although position interpolation techniques for RoPE can mitigate this issue, an increase in interpolations leads to a decrease in positional resolution. To tackle this challenge, drawing inspiration from the Bloch Sphere representation, we propose a novel rotary position encoding on a three-dimensional sphere, named 3D Rotary Position Encoding (3D-RPE). 3D-RPE is an advanced version of the widely used 2D RoPE, with two major advantages for modeling long contexts: controllable long-term decay and improved position resolution. For controllable long-term decay, 3D-RPE allows for the regulation of long-term decay within the chunk size, ensuring the modeling of relative positional information between tokens at a distant relative position. For improved position resolution, 3D-RPE can mitigate the degradation of position resolution caused by position interpolation on RoPE. We have conducted experiments on long-context Natural Language Understanding (NLU) and long sequence Language Modeling (LM) tasks. From the experimental results, 3D-RPE achieved performance improvements over RoPE, especially in long-context NLU tasks.

Introduction

Rotary Position Encoding (RoPE) (Su et al. 2024) is essential in Transformer-based Large Language Models (LLMs), such as the LLaMA models (Touvron et al. 2023). RoPE merges the advantages of absolute and relative positional encoding by using a rotation mechanism to represent each position. Despite its widespread use in LLMs (Touvron et al. 2023; Wang and Komatsuzaki 2021; Chiang et al. 2023), RoPE has notable limitations when extending LLMs with a predefined context window. The long-term decay problem of RoPE limits the model's ability to extend positions outward in long-context tasks. Although the long-context modeling capability of LLMs can be extended through position interpolation, as more positions are inserted, RoPE encounters the challenge of decreased position resolution (An et al. 2024).

*Corresponding Author: Peng Zhang

We propose a novel position encoding mechanism for transformer architecture, called 3D Rotary Position Encoding (3D-RPE), to address challenges in long-context modeling faced by LLMs using RoPE. Inspired by the Bloch Sphere, 3D-RPE applies rotary position encoding on a three-dimensional spherical surface, as illustrated in Figure 1(b). In contrast, RoPE employs a rotation on a 2-dimensional circular path, as depicted in Figure 1(a). This leads that RoPE suffers from long-term decay. As shown in Figure 1(c), with the increase in relative distance, the relative upper bound on token correlations at modeled relative positions will continuously decrease. Our proposed 3D-RPE addresses this issue by segmenting a long sequence into chunks and setting rotation angles within and between the chunks to construct position encoding. As shown in Figure 1(d), 3D-RPE is able to control this relative upper bound through two relative positional dimensions, namely within and between chunks. Compared to the relative upper bound in RoPE shown in Figure 1(c), our method improves the correlation upper bound for long relative distances and effectively mitigates the problem of long-term decay.

Furthermore, our proposed 3D-RPE alleviates the problem of reduced positional resolution caused by Position Interpolation (PI) (Chen et al. 2023a) on RoPE in long-context modeling. PI methods are often employed to extend LLMs for modeling contexts that exceed the pre-training length. These techniques scale the position encoding during inference, allowing the originally out-of-range position encoding to fall within the trained position interval after interpolation. However, as the interpolation factor increases, PI experiences a substantial decline in positional resolution among tokens, detrimentally affecting long-context modeling performance. As illustrated in Figure 1(e), extending the pre-training length L_p to L using linear PI (Chen et al. 2023a) results in the positional resolution transitioning from the original 1 to $\frac{L_p}{L}$. As L increases, the positional resolution decreases accordingly. However, our proposed 3D-RPE employs a 3D rotating sphere for position encoding. Based on the same positional interpolation, our method supports higher positional resolution compared to RoPE's 2D circular rotation, i.e., $\mathcal{E}'_{3d-rpe} > \frac{L_p}{L}$ (See Figure 1(f)). This benefit has been theoretically proven (see Theorem 1) and corroborated by experimental results (see Table 4 in Ablation Study).

We conducted experiments on long-context Natural Lan-

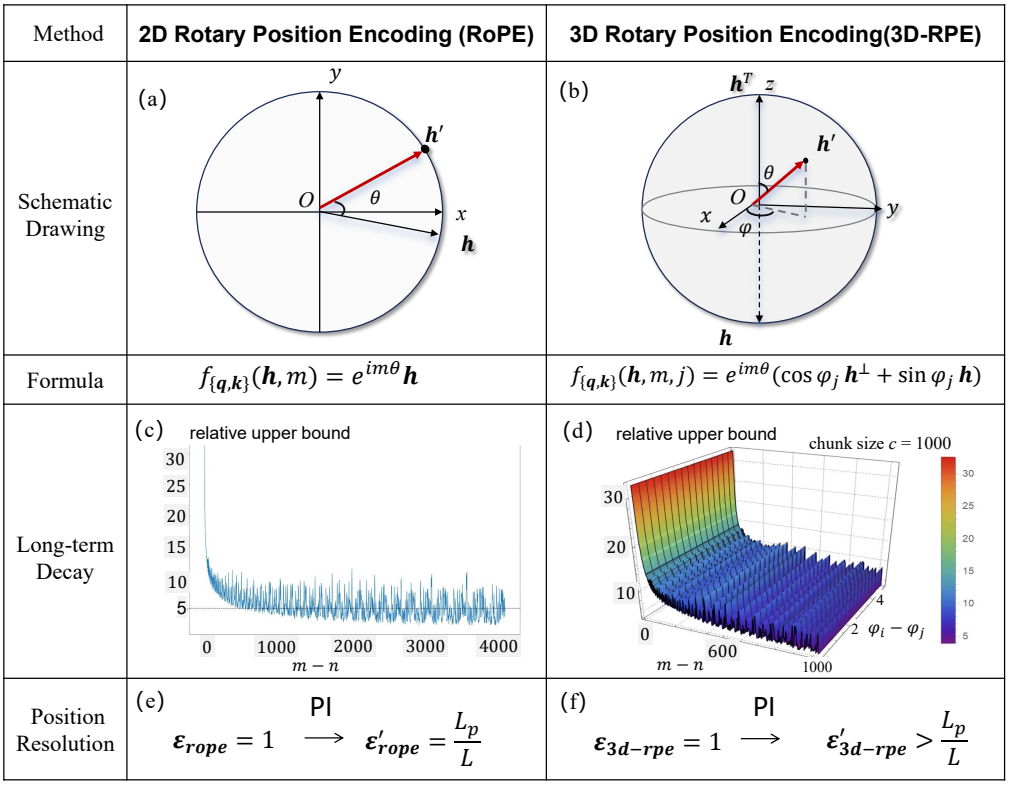

Figure 1: 2D Rotary Position Encoding (RoPE) vs. 3D Rotary Position Encoding (3D-RPE).

guage Understanding (NLU) and long-sequence Language Modeling (LM) tasks. Our experimental results highlight the promising performance of the 3D-RPE method, especially in tasks requiring long-context language understanding.

Our major contributions of this paper are as follows:

- A position encoding method on a 3D sphere, 3D-RPE, is provided, which can enhance the long-context modeling capability of LLMs by replacing RoPE.
- It is proved that 3D-RPE has two benefits, controllable long-term decay and mitigating the reduction in positional resolution caused by position interpolation.
- LLMs combine with 3D-RPE have achieved significant performance improvements in long-context NLU tasks.

Preliminaries

The analysis of 3D-RPE relies on these concepts and results from the filed of Bloch Sphere and RoPE. We offer an introduction to Bloch Sphere and RoPE (Su et al. 2024).

Bloch Sphere

Bloch Sphere (BS) offers a geometric depiction of a quantum mechanical system's pure state, limited to two levels. The state vector $|\phi\rangle$ is mathematically expressed as

$$|\phi\rangle = e^{i\theta}(\cos\frac{\varphi}{2}|0\rangle + \sin\frac{\varphi}{2}e^{i\theta_1}|1\rangle) \quad (1)$$

where $|0\rangle$ and $|1\rangle$ are Dirac's notations. θ, θ_1 and φ are rotation angles.

In our work, θ encodes the relative positions of tokens within chunks, φ encodes the relative positions of tokens across chunks, and θ_1 is equal to 0.

Rotary Position Encoding

Rotary Position Encoding (RoPE) is a commonly used relative position encoding technique in LLMs, such as LLaMA (Touvron et al. 2023), GPT-J (Wang and Komatsuzaki 2021), Vicuna (Chiang et al. 2023) and etc. RoPE is a 2-dimensional space rotary encoding, which is denoted as follows:

$$RoPE(\boldsymbol{h}_m, m) = e^{\mathrm{i}m\theta}\boldsymbol{h}_m\ ,\ RoPE(\boldsymbol{h}_n, n) = e^{\mathrm{i}n\theta}\boldsymbol{h}_n \quad (2)$$

$\boldsymbol{h}_m$ and $\boldsymbol{h}_n$ are hidden vectors from the Query and Key for a specific attention head in transformer. For ease of differentiation, $\boldsymbol{h}_m$ and $\boldsymbol{h}_n$ can be refined later as $\boldsymbol{q}_m$ and $\boldsymbol{k}_n$, i is the imaginary unit, θ is the rotary angle in RoPE. m and n are indexes about positions. Then, the inner product is employed to define the self-attention score before softmax computing:

$$\begin{aligned} s(m-n, \boldsymbol{q}_m, \boldsymbol{k}_n) &= \langle RoPE(\boldsymbol{q}_m, m), RoPE(\boldsymbol{k}_n, n)\rangle \\ &= Re[\sum_{l=0}^{d/2-1} \boldsymbol{q}_{[2l:2l+1]}\boldsymbol{k}_{[2l:2l+1]}e^{\mathrm{i}(m-n)\theta_l}] \end{aligned} \quad (3)$$

Eq (3) is unary function respect to the relative position $(m-n)$, representing the relative position between tokens and modeling the relative positional information. Here, $Re[\cdot]$ denotes the calculation of the real part of a complex number.

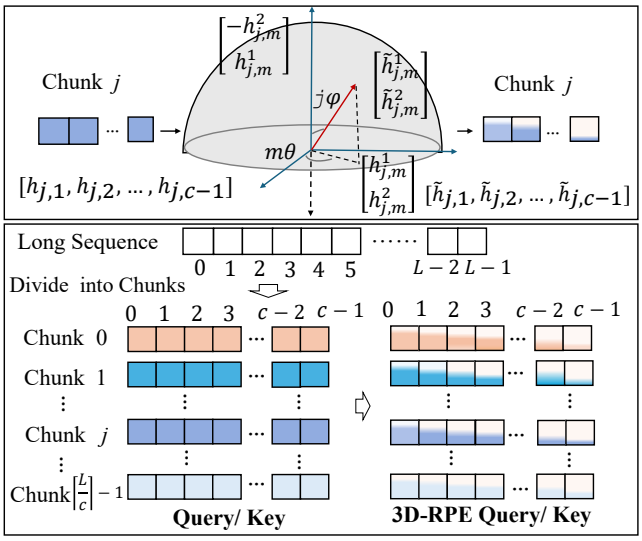

Figure 2: Visualization of the 3D Rotary Position Encoding (3D-RPE). The context size is L, and the chunk size is c. The vectors $[\boldsymbol{h}^1_{j,m}, \boldsymbol{h}^2_{j,m}]^T$ and $[-\boldsymbol{h}^2_{j,m}, \boldsymbol{h}^1_{j,m}]^T$ form an orthogonal basis, corresponding to the $|1\rangle$ and $|0\rangle$ states in Eq. (1).

In our study, the 3D-RPE self-attention score is a binary function containing the relative position $(m-n)$.

Method

In this section, we first introduce the new position encoding on a 3D sphere, 3D-RPE. Then, the benefits of 3D-RPE are described, which focuses on analyzing two benefits of 3D-RPE, namely controllable long-term decay and improved position resolution.

3D Rotary Position Encoding

For a long sequence of length L and a chunk size set to c, where c is smaller than the pre-training length of LLM, the sequence can be divided into $\lceil L/c \rceil$ chunks. Here, $\lceil . \rceil$ represents the ceiling function, rounding up to the nearest integer (see Figure 2). The state vector $\boldsymbol{h}_{j,m}$ comes from either Query or Key. Here, $j \in [0, \lceil L/c \rceil - 1]$ represents the positional index of the chunk, and $m \in [0, c-1]$ indicates the positional index of the token within the chunk. This is used to calculate the new state vector $\widetilde{\boldsymbol{h}}_{j,m}$ by rotating on the Bloch Sphere. Specifically, two rotation angles, θ and φ are defined, with θ governing the position encoding within the chunk's internal tokens, and φ governing the position encoding between the chunks. Our position encoding method is called 3D Rotary Position Encoding, or 3D-RPE.

Definition 1 (3D Rotary Position Encoding). *Let $\boldsymbol{h}_{j,m} \in \mathbb{R}^d$ be a state vector of an attention head without position encoding, where d is the dimension of the vector, which is an even number. 3D-RPE encodes $\boldsymbol{h}_{j,m}$ into the vector $\widetilde{\boldsymbol{h}}_{j,m}$, which can be formalized as:*

$$\widetilde{\boldsymbol{h}}_{j,m} = e^{\mathrm{i}m\theta}(\cos\varphi_j \boldsymbol{h}^{\perp}_{j,m} + \sin\varphi_j \boldsymbol{h}_{j,m}) \tag{4}$$

i *is the imaginary unit.* $\boldsymbol{h}^{\perp}_{j,m}$ *equals to* $[-\boldsymbol{h}^2_{j,m}, \boldsymbol{h}^1_{j,m}]^T$*, where* $\boldsymbol{h}^1_{j,m} \in \mathbb{R}^{d/2}$ *and* $\boldsymbol{h}^2_{j,m} \in \mathbb{R}^{d/2}$ *is the first and second halves of the state vector* $\boldsymbol{h}_{j,m}$*.*

In transformer-based LLMs, after applying position encoding to the state vectors from Query and Key, it is essential to compute their attention scores. For the sake of clarity and formalization, we denote the position encoding of the state vector from Query as 3d-PE$(\boldsymbol{q}, i, m)$ and from Key as 3d-PE$(\boldsymbol{k}, j, n)$, where i and j range from 0 to $\lceil L/c \rceil - 1$, and m and n range from 0 to $c-1$. The self-attention score can be obtained through the conjugate symmetric inner product of $\boldsymbol{q}_{i,m}$ and $\boldsymbol{k}_{j,n}$, which are the state vectors from Query and Key,

$$s(\boldsymbol{q}_{i,m}, \boldsymbol{k}_{j,n}, \varphi_i - \varphi_j, m-n) = Re[e^{\mathrm{i}(\varphi_i - \varphi_j)} \sum_{l=0}^{d/2-1} e^{\mathrm{i}(m-n)\theta_l}(\boldsymbol{q}_l \boldsymbol{k}_l + \boldsymbol{q}_{d/2+l}\boldsymbol{k}_{d/2+l})] \tag{5}$$

where $l \in [0, \frac{d}{2} - 1]$, $\theta_l = base^{-l}$, $\varphi_j = base^{-j}$ and $\varphi_i = base^{-i}$. Let $\{\boldsymbol{q}, \boldsymbol{k}\}_l$ denote the l-th components of $\{\boldsymbol{q}, \boldsymbol{k}\}$. In experiments using the LLaMA2 models, the $base$ is generally set to $10,000$. In LLaMA3 models, the $base$ of θ_l is $50,000$. The self-attention score computed after applying 3d-PE is a function of both the relative position between chunks $(\varphi_i - \varphi_j)$ and the relative position $(m-n)$.

Consequently, the self-attention score relying on 3d-PE is influenced by the relative positions at both the chunk and token levels. It is important to highlight that when $\boldsymbol{q}_{i,m}$ and $\boldsymbol{k}_{j,n}$ reside within the same chunk (i.e., $i = j$), Eq. (5) simplifies to the standard RoPE formulation as depicted in Eq. (3). For a detailed derivation and computation process of Eq. (5), as well as the complete formulation of Eq. (4).

Benefits of 3D-RPE

In this section, we delve into two benefits offered by 3D-RPE: the ability to control long-term decay and mitigate the reduction in positional resolution caused by position interpolation.

Controllable Long-term Decay 3D-RPE has the property of controllable long-term decay. Analogous to RoPE, by considering the absolute value s in Eq (5) and utilizing the Abel transformation, we derive the upper bound of the correlation coefficients related to term dependencies as follows:

$$|s(\boldsymbol{q}_{i,m}, \boldsymbol{k}_{j,n}, \varphi_i - \varphi_j, m-n)| \le |e^{\mathrm{i}(\varphi_i - \varphi_j)}| \cdot |\sum_{l=0}^{\frac{d}{2}-1} E_{l+1}(h_{l+1} - h_l)| \le (\max_l |h_{l+1} - h_l|) \sum_{l=0}^{d/2-1} |E_{l+1}| \tag{6}$$

where $\cdot$ denotes multiplication, $E_l = \sum_{k=0}^{l-1} e^{\mathrm{i}(m-n)\theta_k}$ and $E_0 = 0$. For RoPE (Su et al. 2024), the relative upper bound E_{rope} is given by $\frac{1}{d/2}\sum_{j=1}^{d/2}|S_j|$, where $S_j = \sum_{t=0}^{j-1} e^{i(m-n)\theta_t}$ (see the section 3.4.3 of RoPE (Su et al. 2024)). By setting $\theta_t = 10000^{\frac{-2t}{d}}$, the value decays as the relative position $(m-n)$ increases. The upper bound E_{3d-rpe} of 3D-RPE is formalized as follows:

$$E_{3d-rpe} = \frac{1}{d/2}\sum_{j=1}^{d/2}|E_l| \tag{7}$$

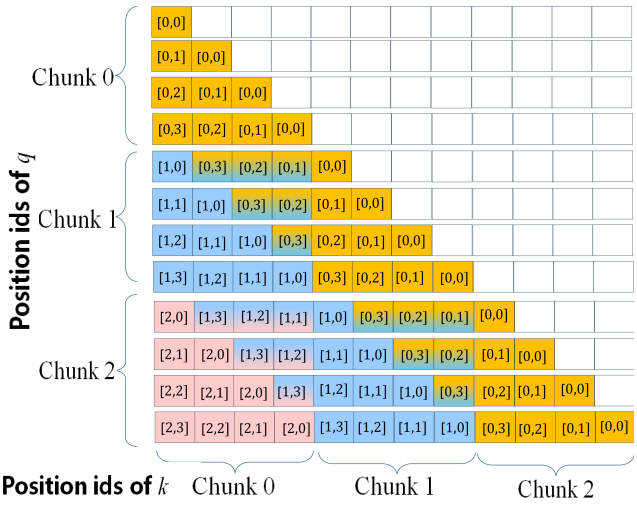

Figure 3: Visualization of the Relative Position Matrix $\boldsymbol{A}$ employing 3D-RPE, with chunk size c=4, and sequence size L=12.

The domains of the relative position $(m-n)$ differ between E_{3d-rpe} and E_{rope}. In E_{rope}, $(m-n)$ is in the range $[0, L-1]$, while in E_{3d-rpe}, it is in $[0, c-1]$. The relative positions between tokens exceeding the chunk size c are constructed collaboratively using positional encoding within and across chunks. The Relative Position Matrix $\boldsymbol{A}$ using 3D-RPE is shown in Figure 3.

To compare and illustrate the advantage of controllable long-term decay, we present the results in Figure 1(c) and Figure 1(d). As shown in Figure 1(c), when the relative position $(m-n)$ exceeds approximately 1000, E_{rope} begins to significantly decrease to below 5. This limitation of $E_{rope} \leq 5$ poses challenges for RoPE in modeling attention scores between tokens with longer relative distances (greater than 4000). In contrast, as shown in Figure 1(d), 3D-RPE employs both $(m-n)$ and $(\varphi_i - \varphi_j)$, setting $c = 1000$ to keep $(m-n)$ within 1000, thereby preventing decay over longer distances. This method ensures E_{3d-rpe} stays at or above 5 for all relative positions.

Improved Positional Resolution Position Interpolation (PI) (Chen et al. 2023a) has been introduced to scale down the position indices to align with the original window size, resulting in enhanced outcomes for context extension. However, as the extension length and interpolation increase, PI can lead to a reduction in relative positional resolution. In contrast, 3D-RPE can also be used alongside PI for long-context extensions. Compared to RoPE combined with PI, 3D-RPE has the advantage of mitigating the reduction in positional resolution caused by positional interpolation, as demonstrated in Theorem 1.

Theorem 1 (Improved Position Resolution). *For a pre-trained language model with a length of L_p and an extension length requirement of L, employing linear position interpolation extension methods $\mathcal{I}$ based on Rotary Position Encoding (RoPE) can elevate the relative positional resolution from $\mathcal{E}_{rope}$ to $\mathcal{E}'_{rope}$. Let $\mathcal{E}'_{3d-rpe}$ denote the relative positional encoding resolution achieved by the method $\mathcal{I}$ based on 3D-RPE, with chunk size $c \geq 3$, there is:*

$$\mathcal{E}'_{3d-rpe} > \mathcal{E}'_{rope} \tag{8}$$

Proof. For 3D-RPE, let the chunk size and chunk number be denoted as c and $n = \lceil L_p/c \rceil$ respectively. Prior to interpolation, the indices within a chunk range from $[0, 1, \cdots, c-1]$. Linear interpolation involves evenly distributing the excess $L - L_p$ tokens across n chunks. This results in new indices within the chunk, range from $[0, 1, 2, \cdots, c'-1]$, where $c' = \lceil L/n \rceil \leq L_p$. So the attention score of $\boldsymbol{q}_{i,m+1}$ and $\boldsymbol{k}_{i,m}$ based on 3D-RPE after interpolation is:

$$\begin{aligned} a_{3d-rpe} &= \boldsymbol{q}\boldsymbol{k}^T e^{\mathrm{i}\theta} e^{\mathrm{i}(\varphi_i - \varphi_i)} \\ &= \boldsymbol{q}\boldsymbol{k}^T e^{\mathrm{i}\theta} \end{aligned}$$

The resolution of relative position for 3D-RPE is:

$$\mathcal{E}'_{3d-rpe} = 1$$

For special cases $\boldsymbol{q}_{(i+1,0)}$ and $\boldsymbol{k}_{(i,c'-1)}$:

$$\mathcal{E}'_{3d-rpe} \geq c' - 1 + \frac{(\varphi_{i+1} - \varphi_i)}{\theta} > c' - 2 \geq 1 \tag{9}$$

where $(\varphi_{i+1} - \varphi_i)/\theta \geq -1/10000 > -1$. As long as $c' \geq 3$, there is $\mathcal{E}'_{3d-rpe} \geq 1 > \mathcal{E}'_{rope} = L_p/L$. Under normal case, the chunk size c is not set to a very small number, hence $c' \geq 3$ is certainly established; moreover, for different interpolation lengths L, we need to configure a varying number of chunks n, such that $c' = \lceil L/n \rceil \leq L_p$. □

To empirically validate the superior performance of this benefit in a training-free setting, it has been observed that methods combining RoPE with interpolation lead to a significant increase in Perplexity as the modeling length increases in language modeling tasks. Conversely, the increase in Perplexity is substantially smaller when employing 3D-RPE with linear interpolation (Refer to Table 4). This phenomenon indicates that this benefit has led to an improvement in the performance of long sequence language modeling.

Related Work

This section provides an overview of the literature related to position encoding and context extension.

Position Encoding (PE): PE is important for Transformer-based language models. Earlier studies (Shaw, Uszkoreit, and Vaswani 2018; Raffel et al. 2020; Wang et al. 2020; Su et al. 2024) have focused on enhancing the original absolute position encoding to develop better relative position encoding, thereby improving the text modeling capabilities of language models. These works (Shaw, Uszkoreit, and Vaswani 2018; Raffel et al. 2020; Wang et al. 2020) utilized trainable position vector encoding to directly incorporate positional information into context representations. Although effective, these methods typically add positional information to contextual representations, making them unsuitable for linear self-attention architectures. RoFormer (Su et al. 2024) introduced

relative position information by rotating context representations, known as RoPE. Transformers utilizing RoPE have become a prevalent backbone in various LLM designs (Touvron et al. 2023; Chowdhery et al. 2022; Wang and Komatsuzaki 2021). Our proposed 3D-RPE differs from the 2-dimensional space of RoPE by modeling the relative position of tokens through rotation on the Bloch Sphere.

Long-context LLMs based on RoPE: To enhance the contextual capabilities of Large Language Models (LLMs) using RoPE, several positional encoding interpolation techniques have been developed. These include Linear Position Interpolation (LPI) (Chen et al. 2023a), Neural Tangent Kernel (NTK) (Peng and Quesnelle 2023), and Yet Another Recurrent Network (YaRN) (Peng et al. 2023) interpolation. Position Sequence Tuning (PoSE) (Zhu et al. 2023) has notably increased sequence lengths to $128k$ by amalgamating these positional interpolation strategies. Additionally, LongLora (Chen et al. 2023b) introduced the shift-short attention mechanism, allowing for effective emulation of full attention and extending sequences up to $100k$, leveraging the LLMa-2-7B model and LoRA's fine-tuning approach (Hu et al. 2022). 3D-RPE further strengthens the positional relationships between distant tokens by capturing inter-chunk positional information and is compatible with existing fine-tuning techniques like LoRA to bolster long-context representation. The Dual Chunk Attention (DCA) (An et al. 2024) method, which enhances the use of pre-trained integer-based parameters, splits query and key sequences into chunks and uses three specialized matrices to capture the relative positions within and between these chunks. This method enhances the model's ability to process longer sequences, but it is unable to model the relative positions within distant chunks. In our work, we employ rotating positional encoding to link attention across different chunks.

Experiments

We evaluate our proposed 3D-RPE on LLaMA2 (Touvron et al. 2023) models (specifically, LLaMA-2-7B and LLaMA-2-7B-chat), which have a $4k$ pre-training context, and LLaMA-3-8B-Instruct (AI@Meta 2024), which has an $8k$ pre-training context. Our experiments aim to explore the following aspects: (1) The effect of 3D-RPE on long-context generation can be assessed using Perplexity. (2) The impact of 3D-RPE on long-context understanding and generation tasks, can be reflected by the accuracy of long sequence natural language tasks, e.g., multiply documents QA. (3) Ablation studies to confirm the advantages of 3D-RPE in position interpolation. Our code, data, and appendix are available on GitHub (https://github.com/maxindian/3D-RPE-Long-Contex-Modeling)

Experimental Settings

We elaborate on the experimental setup by introducing two types of tasks (i.e., long-context language understanding and long sequence language modeling) and detailing three aspects of the configuration (i.e., training setting, datasets, and baseline models).

Training Setting: For long-context Natural Language Understanding (NLU) tasks, we have fine-tuned LLaMA-2-7B-chat and LLaMA-3-8B-Instruct. The fine-tuning method follows the fine-tuning strategy of LongChat (Li et al. 2023a). The training step is $3,000$. For the long-sequence Language Modeling (LM) tasks, we have fine-tuned LLaMA-2-7B to support extended context length of $32k$ tokens. The training step is $1,000$. We set the per-device batch size as 1, and gradient accumulation step as 8, which means that the batch size is 8. We train the model with the next token prediction objective with LoRA (Hu et al. 2022).

We employed the AdamW optimizer (Loshchilov and Hutter 2019) with $\beta_1 = 0.9$ and $\beta_2 = 0.95$ for all fine-tuned models. Chunk size is set to $3k$. The learning rate was set to 2×10^{-5}, and a linear learning rate warmup was applied. Training was conducted on a single 4xA800 GPU machine using FlashAttention-2 (Dao 2023).

Datasets: In the context of long-context NLU tasks, we employ the LongAlpaca-12k dataset, which contains 9,000 LongQA and 3,000 short QA entries (Chen et al. 2023c), and the LongAlpace-16k-length dataset (Chen et al. 2023b). To evaluate the performance of 3D-RPE for long-context extension, we use the LongBench (Bai et al. 2023), which includes 13 English tasks, 5 Chinese tasks and 2 code tasks, with most tasks having an average context length of $5k$ to $15k$ tokens. We focus on the English and code tasks to evaluate our method, 3D-RPE. Additionally, the LEval (An et al. 2023) evaluation set, which also consists of long-context datasets, is used to verify the effectiveness of 3D-RPE. The five datasets annotated from scratch in LEval, namely Coursera, QuALiTY, CodeU, GSM,and TOEFL, are utilized.

For long-sequence LM tasks, we use the RedPajama-Data (Computer 2023) for fine-tuning training. The dataset is a large-scale pre-training dataset (the size reaches 1.2 trillion tokens) designed to provide high-quality training data for language models, and contains multiple data sources (i.e., github, arxiv, book, c4 and Wikipedia, etc.). We sample $20,000$ samples from these data sources for training. For evaluation, we utilize the PG19 book corpus dataset (Rae et al. 2020), which includes 100 documents, and the Arxiv Math Proof-pile dataset (test split). Additionally, all methods evaluate perplexity by using a sliding window following (Press, Smith, and Lewis 2022).

Baseline Models: For long-context NLU tasks, the fine-tuned models, including LongAlpace-16k (Chen et al. 2023b), LongChat-32k (Li et al. 2023b) LongLlama (Tworkowski et al. 2023) and ChatGLM (Du et al. 2022) are used as the baseline models. Models of fine-tuning free in language modeling tasks are also used in long-context NLU tasks.

In long-sequence LM tasks, the methods of LongLoRA (Chen et al. 2023b), StreamingLLM (Xiao et al. 2023), Positional Interpolation (PI) (Chen et al. 2023a), and NTK-Aware Scale RoPE (NTK) (Peng and Quesnelle 2023) are selected as the baselines, all based on the LLaMA-2-7B-base model. Among these baseline models, PI, NTK and StreamingLLM are fine-tuning-free methods. The fine-tuned models include LongLoRA and Activation Beacon (Zhang et al. 2024). In Ablation experiments, both the fine-tuned training model and the untrained model of the PI method are considered as baseline models. Our model's numerical precision is set to FP16.

METHODS	Single-Doc QA	Multi-Doc QA	Summarization	Few-shot	Code
LLaMA-2-7B-chat	24.90	22.60	24.70	60.01	48.10
LLaMA-2-7B-chat-PI	18.98	17.16	25.03	49.43	52.73
LLaMA-2-7B-chat-NTK	23.21	23.34	24.40	59.29	49.28
StreamingLLM	21.47	22.22	22.20	50.05	48.00
ChunkLLaMA-16k	24.04	22.98	21.52	46.31	49.73
LongChat-32k	31.58	23.50	26.70	64.02	54.10
LongAlpaca-16k	28.70	28.10	27.80	63.70	56.00
LongLLaMA	30.12	16.37	24.19	60.31	66.05
Vicuna-v1.5-7B-16k	28.01	18.63	26.01	66.20	47.30
ChatGLM3-6B-32k	40.30	46.60	**29.50**	68.10	56.20
3D-RPE-LLaMA2-7B-Chat	**47.40**	**60.10**	28.99	**73.16**	**76.50**

Table 1: Comparison between open-source based models on long-context NLU tasks. Our model, 3D-RPE-LlaMA2-7B-Chat is fine-tuning on LLaMA-2-7b-chat, which is extended from $4k$ to $16k$ context lengths. Baseline models can be categorized into two groups: those that necessitate fine-tuning during training (such as LongAlpaca and LongLLaMA), and those that do not require it (including PI, NTK, StreamingLLM, and ChunkLLaMA-16k).

MODELS	Coursera	QuALiTY	CodeU	GSM	TOEFL
LLaMA-2-7B-Chat	29.21	37.62	1.11	19.00	51.67
LongChat-7B-16K	29.74	33.66	3.33	10.00	47.95
LLaMA2-7B-NTK	32.71	33.16	0.00	19.00	52.78
Vicuna1.5-7B-16k	38.66	**39.60**	**5.55**	19.00	55.39
3D-RPE-LLaMA2-7B-Chat(ours)	**39.38**	38.11	2.22	**21.01**	**57.99**
LLaMA3-8B-Instruct*	51.45	**64.34**	4.44	76.00	82.89
3D-RPE-LLaMA3-8B-Instruct*	**51.89**	61.38	4.44	**80.00**	82.89

Table 2: Comparison with open-source models, LLaMA-2-7B-chat, LLaMA3-8B-Instruct, on 5 closed-ended-ended tasks with various input length from LEval (An et al. 2023). The evaluation metric "EM," which represents the exact match score, is adopted. * indicates the model is train-free.

Long-Context Natural Language Understanding

In this task, the LongBench (Bai et al. 2023) evaluation set was initially utilized. Five categories of tasks were included: single-document QA (3 tasks), multi-document QA (3 tasks), summarization (3 tasks), few-shot learning (3 tasks), and code completion (2 tasks). The average score for each type is reported in Table 1. The evaluation metrics followed those specified in LongBench (Bai et al. 2023). The results in Table 1 highlight our model's significant performance advantages over baseline models in four tasks, both for models without training and those with fine-tuning. To compare with the well-performing long sequence model at that time, we included ChatGLM3 in Table 1, even though it did not use the same base LLM. Both our method and other baseline methods use LLaMA2-7B-Chat as the base model. In summarization tasks, our model also achieved performance comparable to ChatGLM3-6B-32k. These experimental outcomes indicate that our model enhances the correlation between tokens with distant relative positions in long contexts through 3D-RPE, improving the experimental performance of the LLaMA-2-7B-Chat model in long-context understanding tasks.

Subsequently, the LEval Benchmark (An et al. 2023) was employed. Table 2 reveals that our model, 3D-RPE-LLaMA2-7B-Chat, outperformed LLaMA2-7B-NTK and LongChat-7B-16K. Although it did not surpass Vicuna1.5-7B-16K in Quality and CodeU tasks, it excelled in the Coursera, GSM, and TOEFL tasks. Additionally, we conducted experiments on LLaMA3-8B-Instruct using a $16k$ context window with 3D-RPE. The 3D-RPE-LLaMA3-8B-Instruct* showed performance improvements in the Coursera and GSM tasks. While 3D-RPE did not enhance performance in the CodeU, TOEFL, and QuALiTY tasks, there was no significant performance decline either. These experimental results demonstrate the effectiveness of the 3D-RPE method.

Long-Sequence Language Modeling

In Table 3, we present the perplexity scores for our model, 3D-RPE-LLaMA-2-7B and baseline models on the proof-pile and PG19 test datasets. 3D-RPE-LLaMA-2-7B was fine-tuned from the LLaMA2-7B-Base model using a dataset with a $32k$ context window. To evaluate performance, we set sequence lengths of $8k$, $16k$, and $32k$. We extended our model's sequence length from $32k$ to $100k$ using the position extending method from PoSE (Zhu et al. 2023). The results indicate that our method outperforms train-free sequence extending methods, namely positional interpolation (PI and NTK) and StreamingLLM. Compared to fine-tuned models, our model shows better performance at $8k$ and $16k$ sequence lengths. This suggests that the new positional encoding, 3D-RPE, improves or maintains modeling performance for larger context windows ($32k$) compared to smaller ones ($8k$ and $16k$). For the $32k$ and $100k$ tasks, although our model did not surpass LongLoRA-32k and LongLoRA-100k, it did outperform LongChat-32k and Activation Beacon.

Notably, our model can further extend from $32k$ to $100k$

METHODS	PG-19				Proof-Pile			
	$8k$	$16k$	$32k$	$100k$	$8k$	$16k$	$32k$	$100k$
LLaMA2-7B-Base	131.09	$> 10^2$	$> 10^2$	OOM	16.79	$> 10^2$	$> 10^2$	OOM
LLama2-7B-PI	11.32	19.5	$> 10^2$	OOM	3.86	5.94	33.7	OOM
LLama2-7B-NTK	10.28	11.5	37.8	OOM	3.98	5.94	33.7	OOM
StreamingLLM	9.23	9.25	9.24	9.32	3.47	3.51	3.50	3.55
LongLoRA-32k	7.33	7.16	**7.04**	–	2.78	2.61	**2.50**	–
LongLoRA-100k	7.57	7.33	7.16	**7.04**	2.78	**2.60**	2.58	**2.52**
LongChat-32k	8.92	8.85	8.81	OOM	2.98	2.70	2.65	OOM
Activation Beacon	8.52	8.54	8.56	8.68	3.45	3.42	3.39	3.35
3D-RPE-LLaMA2-7B	**7.03**	**7.10**	8.09	8.12	**2.72**	2.93	2.89	3.05

Table 3: Perplexity evaluation on different extending methods. We conduct evaluation on the Proof-pile and PG-19 test datasets, varying evaluation context window size from $8k$ to $100k$.'-' indicates that this method cannot be further extended and evaluation results can not be obtained. 'OOM' is an abbreviation for 'Out of Memory'.

MODELS	$4k$	$8k$	$16k$	$32k$
LLaMA2-7B-PI	7.94	9.19	15.11	$> 10^2$
LLaMA2-7B-NTK	7.87	11.98	26.12	58.91
LLaMA2-7B-Yarn	7.87	8.06	9.82	11.74
3D-RPE-LLaMA2-7B*	7.87	**7.90**	**7.71**	**9.34**
LLaMA2-7B-PI+	–	8.02	**8.05**	$> 10^2$
3D-RPE-LLaMA2-7B	–	**7.85**	8.15	**8.82**

Table 4: Results are evaluated in Perplexity on PG19 validation split. '*' denotes train-free. '+' indicates the fine-tuned version of the data. The context length of $8k$ is extended directly with 3D-RPE. Achieving $16k$ and $32k$ is accomplished through PI with chunks based on the $8k$ context length.

without significantly increasing perplexity values, in combination with other train-free extension methods. However, due to its specific attention mechanism, the LongLoRA models cannot be extended beyond their predefined context windows in a train-free manner. For instance, LongLoRA-$32k$ cannot be further extended to $100k$.

Ablation Study

In this section, we conduct ablation studies in this section to explore how 3D-RPE affects the linear interpolation method. We compare position interpolation methods (PI, NTK, and Yarn) with the method that combines 3D-RPE with position interpolation on the LLaMA-2-7B-Base model in a train-free manner. The experimental results can be found in Table 2. The 3D-RPE-LLaMA2-7B* model with linearly positional interpolation from $8k$ to $16k$ and $32k$, the 3D-RPE approach yields improved results by mitigating the decrease in positional resolution caused by interpolation methods. These results are consistent with the findings of Theorem 1. Additionally, our method incorporates the PI technique. To further demonstrate the effectiveness of 3D-RPE, LLaMA-2-7B-PI is fine-tuned on the same dataset as 3D-RPE-LLaMA2-7B. Our method achieved lower PPL for $8k$ and $32k$.

To analyze the impact of different chunk size settings on model performance, we applied 3D-RPE to the base model LLaMA2-Base without fine-tuning. Then, we extended the modeling context length from 4k to 8k, and set the chunk sizes to 1k, 2k, 2.5k, 3k, 3.5k, and 4k respectively. The PPL scores are $9.67, 9.08, 8.98, 7.83, 7.85$, and 8.81 respectively. These results suggest that the chunk size setting should be chosen to be close to the context length used during model pre-training, as this allows for better utilization of the positional encoding information learned during pre-training within the chunk. However, selecting a chunk size that is too close to the pre-training length is also not ideal, because LLMs typically use a fixed context length during pre-training, and the lengths of the pre-training texts are not aligned, which leads to suboptimal learning of positional information near the maximum pre-training length.

Conclusion and Future Work

In this paper, we present a novel rotary position encoding method called 3D Rotary Position Encoding (3D-RPE). Compared to RoPE, we have theoretically proved that 3D-RPE possesses two key advantages: controllable long-term decay and improved interpolation resolution. Experimentally, 3D-RPE has excelled in long-context NLU tasks. 3D-RPE doesn't require many more samples for continual training because it effectively uses the position encoding of the pre-trained LLM within the chunk. The limitation of our work is the lack of further research on the design of relative positional spacings on chunks. For example, the transition between the last token of one chunk and the first token of the next chunk is not smooth. In our experiments, in order to better adapt to the positional encoding relationships pre-trained by LLM, the relative positional spacings on Chunks are compressed as the index of the Chunk increases.

In the future, 3D-RPE holds promise as a foundational positional encoding strategy for LLMs, especially in the aspect of modeling long contexts. Moreover, given that 3D-RPE encapsulates positional encoding within a three-dimensional framework, it has the potential to integrate with visual data, thereby facilitating an in-depth exploration of its efficacy in synchronizing graphical and textual semantic information.

Acknowledgements

The present research is supported in part by the Natural Science Foundation of China (grant No. 62276188), TJU-Wenge

joint laboratory funding. We would like to thank Yizhe Li, Yuan Han and the anonymous reviewers for their insightful comments.

References

AI@Meta. 2024. Llama 3 Model Card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md. Accessed:2024-04-18.

An, C.; Gong, S.; Zhong, M.; Li, M.; Zhang, J.; Kong, L.; and Qiu, X. 2023. L-eval: Instituting standardized evaluation for long context language models. *arXiv preprint arXiv:2307.11088*.

An, C.; Huang, F.; Zhang, J.; Gong, S.; Qiu, X.; Zhou, C.; and Kong, L. 2024. Training-Free Long-Context Scaling of Large Language Models.

Bai, Y.; Lv, X.; Zhang, J.; Lyu, H.; Tang, J.; Huang, Z.; Du, Z.; Liu, X.; Zeng, A.; Hou, L.; Dong, Y.; Tang, J.; and Li, J. 2023. LongBench: A Bilingual, Multitask Benchmark for Long Context Understanding. *arXiv preprint arXiv:2308.14508*.

Chen, S.; Wong, S.; Chen, L.; and Tian, Y. 2023a. Extending context window of large language models via positional interpolation. arXiv:2306.15595.

Chen, Y.; Qian, S.; Tang, H.; Lai, X.; Liu, Z.; Han, S.; and Jia, J. 2023b. LongLoRA: Efficient Fine-tuning of Long-Context Large Language Models. *arXiv:2309.12307*.

Chen, Y.; Yu, S.; Qian, S.; Tang, H.; Lai, X.; Liu, Z.; Han, S.; and Jia, J. 2023c. Long Alpaca: Long-context Instruction-following models. https://github.com/dvlab-research/LongLoRA.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. *URL https://lmsys.org/blog/2023-03-30-vicuna*, 3(5).

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Computer, T. 2023. Redpajama: An open source recipe to reproduce llama training dataset. https://github.com/togethercomputer/RedPajama-Data.

Dao, T. 2023. FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning. *CoRR*.

Du, Z.; Qian, Y.; Liu, X.; Ding, M.; Qiu, J.; Yang, Z.; and Tang, J. 2022. GLM: General Language Model Pretraining with Autoregressive Blank Infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 320–335.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Li, D.; Shao, R.; Xie, A.; Sheng, Y.; Zheng, L.; Gonzalez, J.; Stoica, I.; Ma, X.; and Zhang, H. 2023a. How Long Can Context Length of Open-Source LLMs truly Promise? In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Li, D.; Shao, R.; Xie, A.; Sheng, Y.; Zheng, L.; Gonzalez, J. E.; Stoica, I.; Ma, X.; and Zhang, H. 2023b. How Long Can Open-Source LLMS Truly Promise on Context Length? *arXiv preprint arXiv:2306.04537*.

Loshchilov, I.; and Hutter, F. 2019. Decoupled weight decay regularization. *In ICLR, 2019*.

Peng, B.; and Quesnelle, J. 2023. Ntk-aware scaled rope allows llama models to have extended (8k+) context size without any fine-tuning and minimal perplexity degradation. https://www.reddit.com/r/LocalLLaMA/comments/14lz7j5/ntkaware_scaled_rope_allows_llama_models_to_have.

Peng, B.; Quesnelle, J.; Fan, H.; and Shippole, E. 2023. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2203.13474*.

Press, O.; Smith, N. A.; and Lewis, M. 2022. Train short, test long: Attention with linear biases enables input length extrapolation. *In ICLR, 2022*.

Rae, J. W.; Potapenko, A.; Jayakumar, S. M.; Hillier, C.; and Lillicrap, T. P. 2020. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140): 1–67.

Shaw, P.; Uszkoreit, J.; and Vaswani, A. 2018. 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*, 464–468.

Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; and Liu, Y. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568: 127063.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Tworkowski, S.; Staniszewski, K.; Pacek, M.; Wu, Y.; Michalewski, H.; and Miłos, P. 2023. Focused Transformer: Contrastive Training for Context Scaling. *arXiv preprint arXiv:2307.03170*.

Wang, B.; and Komatsuzaki, A. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. *GitHub*.

Wang, B.; Zhao, D.; Lioma, C.; Li, Q.; Zhang, P.; and Simonsen, J. G. 2020. Encoding word order in complex embeddings. In *International Conference on Learning Representations*.

Xiao, G.; Tian, Y.; Chen, B.; Han, S.; and Lewis, M. 2023. Efficient Streaming Language Models with Attention Sinks. *arXiv*.

Zhang, P.; Liu, Z.; Xiao, S.; Shao, N.; Ye, Q.; and Dou, Z. 2024. Soaring from 4K to 400K: Extending LLM's Context with Activation Beacon. arXiv:2401.03462.

Zhu, D.; Yang, N.; Wang, L.; Song, Y.; Wu, W.; Wei, F.; and Li, S. 2023. PoSE: Efficient Context Window Extension of LLMs via Positional Skip-wise Training. *arXiv preprint arXiv:2309.10400*.