# YaRN: Efficient Context Window Extension of Large Language Models

Bowen Peng[1] Jeffrey Quesnelle[1] Honglu Fan[2, 3] Enrico Shippole

[1]Nous Research [2]EleutherAI [3]University of Geneva

## Overview

One reoccurring limitation with positional encodings used in LLMs is the inability to generalize past the context window seen during pre-training.

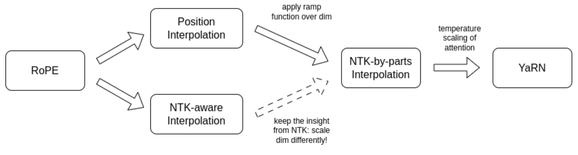
We present YaRN (Yet another RoPE extensioN method), a compute-efficient method to extend the context window of such models, requiring 10x less tokens and 2.5x less training steps than previous methods.

In addition, we demonstrate in the paper that YaRN exhibits the capability to extrapolate beyond the limited context of a fine-tuning dataset.

## RoPE Interpolation

Previous research aimed at increasing the context size of models trained with Rotary Position Embedding (RoPE) primarily focused on interpolating the RoPE embeddings.

Two of the most widely used methods for extending context size prior to this work are Positional Interpolation (PI) [Chen et al., 2023] [kaiokendev, 2023] and NTK-aware Interpolation [bloc97, 2023] (Also known as ABF [Rozière et al., 2023]).




This diagram summarizes the relationship between different methods and how they evolve into YaRN.

## This Work

**NTK-by-parts Interpolation:**

Unlike previous methods, we target and interpolate each RoPE dimensions differently. Given a wavelength $\lambda$ which represents the number of tokens required for a RoPE dimension to complete a full rotation,

- If $\lambda$ is much smaller than the pretrained context size $L$, we do not interpolate;
- If $\lambda$ is equal to or bigger than the pretrained context size $L$, we want to only interpolate and avoid any extrapolation;
- Dimensions in-between can have a bit of both, similar to the "NTK-aware" interpolation.

**YaRN:**

In addition to the previous interpolation techniques, we also observe that introducing a temperature $t$ on the logits before the attention softmax improves perplexity over the extended context size when $t$ is set to the optimal value for a given model.

$$\text{softmax}\left(\frac{q_m^T k_n}{t\sqrt{|D|}}\right). \tag{1}$$

## Experiments

We broadly followed the training and evaluation procedures as outlined in [Chen et al., 2023]. Using the LLaMA 7B model, we extended its context size to 32k using different interpolation methods. YaRN outperforms other methods given the same training budget.

A more extensive ablation experiment can be found in the paper, alongside results from extending Llama 2 models to a context size of 128k.

## Results

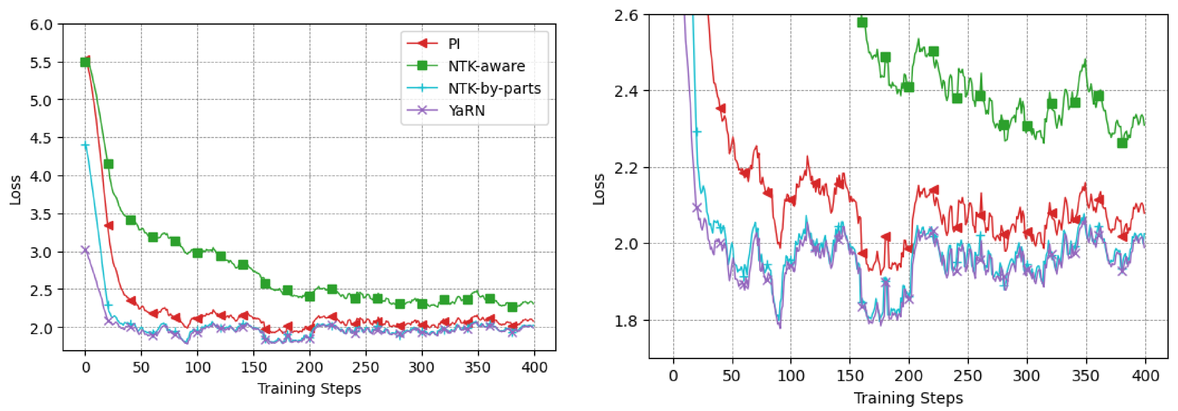



Figure 1: Training loss curves for the LLaMA 7B model extended to 32k context size using different interpolation techniques. The graph on the right is zoomed in.

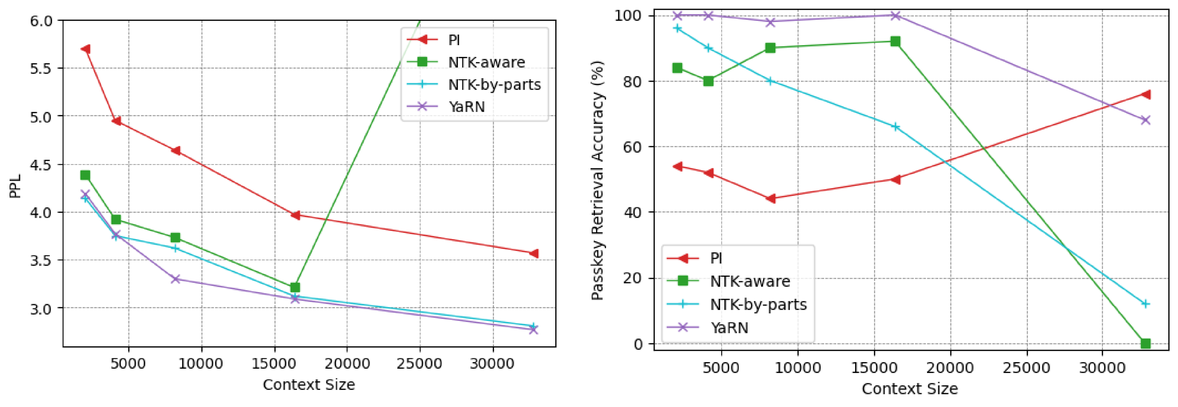



Figure 2: Sliding window perplexity and passkey retrieval accuracy at different prompt lengths after fine-tuning.

## References

[bloc97, 2023] bloc97 (2023).
NTK-Aware Scaled RoPE allows LLaMA models to have extended (8k+) context size without any fine-tuning and minimal perplexity degradation.

[Chen et al., 2023] Chen, S. et al. (2023).
Extending context window of large language models via positional interpolation.

[kaiokendev, 2023] kaiokendev (2023).
Things I'm learning while training superhot.

[Rozière et al., 2023] Rozière, B. et al. (2023).
Code Llama: Open foundation models for code.

[Su et al., 2022] Su, J. et al. (2022).
RoFormer: Enhanced transformer with rotary position embedding.

Paper (OpenReview)
wHBfxhZu1u

Code (Github)
jquesnelle/yarn