Understanding the RoPE Extensions of Long-Context LLMs: An Attention Perspective

Meizhi Zhong[1*], **Chen Zhang**[2], **Yikun Lei**[2], **Xikai Liu**[2], **Yan Gao**[2], **Yao Hu**[2], **Kehai Chen**[1†], **Min Zhang**[1]

[1]Institute of Computing and Intelligence, Harbin Institute of Technology, Shenzhen, China

[2] Xiaohongshu Inc.

meizhi.zhong.1999@gmail.com, chenzhang9702@outlook.com,

{chenkehai,zhangmin2021}@hit.edu.cn,

{zhizhu,xikai,yadun,xiahou}@xiaohongshu.com

Abstract

Enabling LLMs to handle lengthy context is currently a research hotspot. Most LLMs are built upon rotary position embedding (RoPE), a popular position encoding method. Therefore, a prominent path is to extrapolate the RoPE trained on comparably short texts to far longer texts. A heavy bunch of efforts have been dedicated to boosting the extrapolation via extending the formulations of the RoPE, however, few of them have attempted to showcase their inner workings comprehensively. In this paper, we are driven to offer a straightforward yet in-depth understanding of RoPE extensions from an attention perspective and on two benchmarking tasks. A broad array of experiments reveals several valuable findings: 1) Maintaining attention patterns to those at the pretrained length improves extrapolation; 2) Large attention uncertainty leads to retrieval errors; 3) Using longer continual pretraining lengths for RoPE extensions could reduce attention uncertainty and significantly enhance extrapolation.

1 Introduction

Large language models (LLMs) (Radford et al., 2018; Touvron et al., 2023; Zhang et al., 2023; Li et al., 2024; Zhang et al., 2024a,b) have accommodated a wide range of natural language processing applications, such as code completion (Rozière et al., 2023) and question answering (Kamalloo et al., 2023; Jiang et al., 2021; Su et al., 2019). However, a notable challenge limiting further customization is possibly the inability of LLMs to utilize context beyond the pretrained length (Minaee et al., 2024; Chen et al., 2023a) due to the inherent flaw of rotary position embedding (RoPE) being used. Fortunately, RoPE extensions emerge as key ingredients to enabling LLMs to leverage extended context that exceeds pretrained scope (Chen et al., 2023a; Peng et al., 2023; Liu et al., 2023; Han et al., 2023; Rozière et al., 2023). These RoPE extensions focus on improving performance on long texts, yet frustratingly, only a few of them (Liu et al., 2023; Han et al., 2023; Men et al., 2024) have explored the underlying mechanisms in depth.

Thus, we systematically analyze common RoPE extensions more straightforwardly, from the perspective of attention (Vaswani et al., 2017). We include three widely-used RoPE extensions, i.e., position interpolation (Chen et al., 2023a), YaRN (Peng et al., 2023), and NTK-Aware interpolation (Rozière et al., 2023). To our best knowledge, there is simply no research in *understanding RoPE extensions for long-context models thoroughly from an attention perspective*.

As a start, we strive to primarily study these methods on a long-context perplexity test (PPL) and empirically compare their corresponding attention patterns. We found that finetuning LLMs with these RoPE-extension methods which match the original pretraining length improves extrapolation performance. Particularly with the NTK-Aware interpolation method, one can extrapolate up to 32$\times$ beyond the pretrained length. To unleash the reasons behind the successes of these methods, we collect the attention scores respectively distributed in 2K and 8K lengths during inference. The results demonstrate that these methods maintain attention patterns consistent with those observed at the pretrained length. In contrast, the attention patterns of the RoPE are substantially deviated.

Afterward, following literature (Fu et al., 2024), we examine these RoPE extensions on a more challenging long-context test called Needle-in-a-Haystack (Needle) (Kamradt, 2023). We find that the RoPE extensions could pass more tests than the RoPE does. Nonetheless, as the context

[*]Work during Xiaohongshu internship.

[†]Corresponding authors

Proceedings of the 31st International Conference on Computational Linguistics, pages 8955–8962
January 19–24, 2025.

length increased, the RoPE extensions could hardly locate the needles. We associate the observation with attention uncertainty. We uncover that large uncertainty leads to retrieval errors: the positions that incur large attention uncertainty are exactly where the incorrect answers are borrowed from.

We further hypothesize that this large attention uncertainty stems from a mismatch between the context lengths in training and inference. Inspired by the conjecture, a natural way to ease the mismatch is to directly train on longer texts. Experimental results exhibit that, with the same amount of training tokens consumed, using examples with longer contexts largely alleviates uncertainty. Thereby, the ability to digest long texts is promoted.

Our key contributions can be summarized as follows:

- We study various RoPE extensions for length extrapolation in perplexity testing and find that the effectiveness could be yielded from maintaining the original attention patterns.
- We analyze these methods using advanced Needle testing and observe that they may fail to extrapolate to regions where large attention uncertainty persists.
- We hypothesize that large attention uncertainty stems from a context length mismatch between training and inference. It is possible to reduce this large uncertainty by minimizing the mismatch through continual training with lengths closer to those in inference.

2 Backgroud

2.1 Target LLMs

We consider LLaMa series at different sizes to conduct experiments, including MiniMA-2-3B (Zhang et al., 2023), LLaMa-2-7B, and LLaMa-2-13B (Touvron et al., 2023). All these mentioned LLMs consistently use rotary position embeddings to take position information into consideration. Owing to space limitation, we only present the experimental results for LLaMa-2-7B, and the results for MiniMA-2-3B and LLaMa-2-13B, share similar trends with those for LLaMa-2-7B, as shown in Appendix A and B.

2.2 RoPE and Its Extensions

Rotary Position Embedding (RoPE).

Before diving into RoPE extensions, we first briefly describe RoPE itself. The use of RoPE (Su et al., 2021) has become pervasive in contemporary LLMs (Touvron et al., 2023; Bai et al., 2023; Bi et al., 2024). RoPE encodes the position information of tokens with a rotation tensor that naturally incorporates explicit relative position dependency. To illustrate, given a hidden vector $\boldsymbol{h} = [h_0, h_1, ..., h_{d-1}]$, where d is the hidden dimension, and a position index m, RoPE operates as follows:

$$f(\boldsymbol{h}, m) = \begin{pmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \\ \vdots \\ h_{d-2} \\ h_{d-1} \end{pmatrix} \otimes \begin{pmatrix} \cos m\theta_0 \\ \cos m\theta_0 \\ \cos m\theta_1 \\ \cos m\theta_1 \\ \vdots \\ \cos m\theta_{d/2-1} \\ \cos m\theta_{d/2-1} \end{pmatrix} + \begin{pmatrix} -h_1 \\ h_0 \\ -h_3 \\ h_2 \\ \vdots \\ -h_{d-1} \\ h_{d-2} \end{pmatrix} \otimes \begin{pmatrix} \sin m\theta_0 \\ \sin m\theta_0 \\ \sin m\theta_1 \\ \sin m\theta_1 \\ \vdots \\ \sin m\theta_{d/2-1} \\ \sin m\theta_{d/2-1} \end{pmatrix} \quad (1)$$

where $\theta_j = b^{-2j/d}, j \in \{0, 1, ..., d/2-1\}$, and b represents the base frequency for RoPE.

Position Interpolation (PI). As described in Chen et al. (2023b) and Kaiokendev (2023), PI involves proportionally downscaling the position index m to m/α in Equation 1.

NTK-Aware Interpolation (NTK). NTK (Rozière et al., 2023) assumes that interpolating all dimensions equally, as done by PI, may result in the loss of high-frequency information. Therefore, NTK introduces a nonlinear interpolation strategy by adjusting the base frequency b.

Yet another RoPE extensioN (YaRN). Unlike PI and NTK, which treat each dimension of RoPE uniformly, YaRN (Peng et al., 2023) employs a ramp function to combine PI and NTK at varying proportions across different dimensions. Additionally, it introduces a temperature factor to mitigate the distribution shift of the attention caused by long inputs.

Following the default settings of the original papers (Chen et al., 2023a; Peng et al., 2023; Liu et al., 2023), we adjust α from 1 to 16 in m/α for **PI** and **YaRN**, while adjusting b from 10,000 to 1,000,000 for **NTK** in our experiments.

2.3 Long-Context Evaluations

Following existing works (Chen et al., 2023a; Peng et al., 2023; Fu et al., 2024), we use the perplexity test (dubbed PPL) as the primary evaluation and the Needle-in-a-Haystack test as a more challenging evaluation. The perplexity is a primary measure that reflects a model's ability to handle long texts. The Needle-in-a-Haystack test (dubbed Needle) (Kamradt, 2023) requires LLMs to accurately recall

	PI	YaRN	NTK	RoPE
LLaMa-2	1.29	0.05	0.06	0.00
LLaMa-3	2.29	1.72	1.68	2.57

Table 1: Jensen–Shannon (JS) divergence of mean attention distributions between different models at lengths of 2048 (top row) and 8192 (bottom row). A lower JS divergence indicates that the two attention distributions are similar.

a specific sentence (the Needle) embedded at an arbitrary location within a long document (the haystack). We obtain the perplexity on the Proof-pile (Azerbayev et al., 2022) dataset. We follow the standard described in Fu et al. (2024) for the Needle-in-a-Haystack accuracy.

3 RoPE Extensions on PPL

We study RoPE extensions by comparing performance on long-context perplexity testing. From the test, as illustrated in Figure 1, we identify that NTK can extrapolate from 4K to 128K, whereas PI and YaRN can extrapolate to 62K. We observe similar results in both the smaller model MiniMA-2-3B and the larger model LLaMa-2-13B, as illustrated in Figures 1(b) and 1(c). To recognize why these RoPE extensions enable train-short-and-test-long properties in PPL, we collect the attention scores on 10 sequences in 2K and 8K and visualize their attention distributions. The followings are a few key takeaways from the attention perspective:

RoPE extensions maintain the original attention patterns. As shown in Figure 2, similar to the findings from Chen et al. (2023a), we observe that the attention patterns fluctuate when the RoPE is tested on 8K sequences (exceeding the training length). However, with RoPE extensions, the attention distributions, as illustrated in Figures 2(c-e), revert to the original pattern seen in Figure 2(a) when tested on 8K sequences. Similar observations are seen in both LLaMa-2-13B and MiniMA-2-3B, as illustrated in Figures 3 and 6.

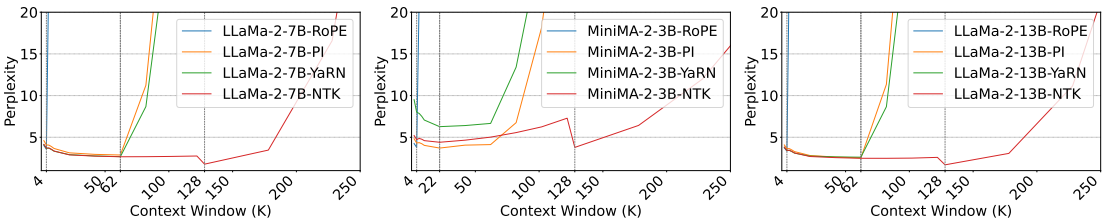
RoPE extensions closely resemble the attention patterns of models trained on longer context. To further verify whether RoPE extensions maintain the original attention patterns, we aim to directly quantify the Jensen–Shannon (JS) divergence between different attention distributions. Using LLaMa-2 and LLaMa-3 as baselines, we collected 10,240 samples of attention distributions to calculate the JS divergence. As illustrated in the bottom row of Table 1, the JS divergence between the RoPE extensions and LLaMa-3 is more minor than between the RoPE and LLaMa-3. This indicates that the attention patterns of RoPE extensions resemble those of models directly trained on a longer context.

NTK and YaRN do not affect the attention patterns within the pretrained length. Some RoPE extensions can degrade performance within the original pretrained length (Peng et al., 2023; Zhang et al., 2024c). To verify whether RoPE extensions alter the attention patterns within the pretrained length, we also calculate the JS divergence among these models' attention distributions at a 2K length. As illustrated on the top row of Table 1, the JS divergence for the NTK and YaRN is very low, almost zero, indicating minimal impact on attention distribution. On the contrary, the JS divergence for the PI is significantly higher. Therefore, we conclude that the NTK and YaRN methods do not affect attention patterns within the pretrained length.

4 RoPE Extensions on Needle

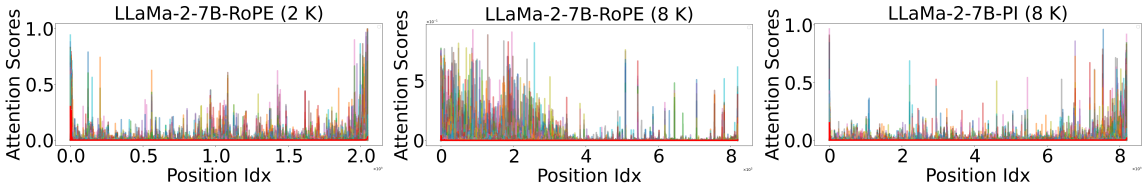
To understand the performance and behavior of the RoPE extensions on more challenging long-context tasks, we conduct Needle testing (Fu et al., 2024). As shown in Figure 4(a-d), LLaMa-2-7B with RoPE extensions can pass more needle tests than the RoPE. However, as the context length increases, some tests fail, resulting in needle retrieval errors. Eventually, almost all fail in extremely long contexts. We also conduct Needle testing on the MiniMA-2-3B and LLaMa-2-13B models with RoPE and PI. Unlike the LLaMa-2-7B, the PI method shows a more significant improvement in the LLaMa-2-13B, as depicted in Figure 7. In contrast, on the MiniMA-2-3B, PI passes only a few needle tests at longer lengths, as illustrated in Figure 5. We attribute these observations to the impact of model size. Below are key takeaways from the attention perspective:

Attention uncertainty leads to more needle retrieval errors. To find the reason behind the needle retrieval errors, we calculate the entropy of attention for each length and depth, as illustrated in Figures 4. For details on the calculation of attention entropy, please refer to Appendix C. Our findings demonstrate that the locations of needle retrieval errors often coincide with high

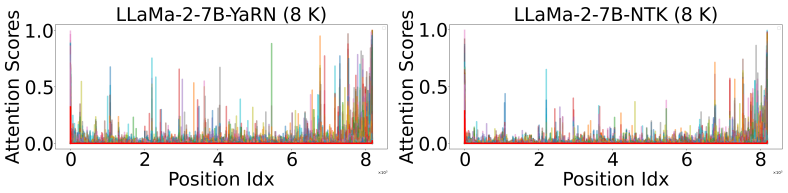


(a) LLaMa-2-7B (b) MiniMA-2-3B (c) LLaMa-2-13B

Figure 1: Perplexity on Proof-pile(Lower is better).

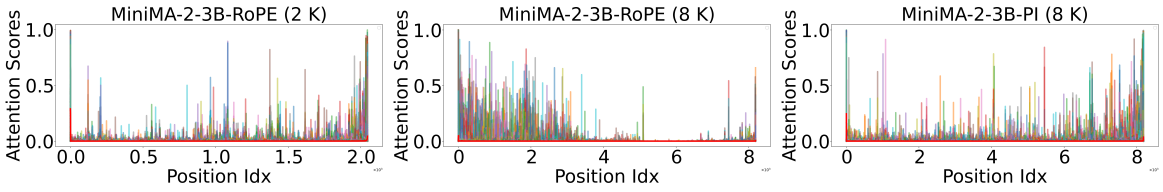


(a) RoPE on 2K sequences. (b) RoPE on 8K sequences. (c) PI on 8K sequences.

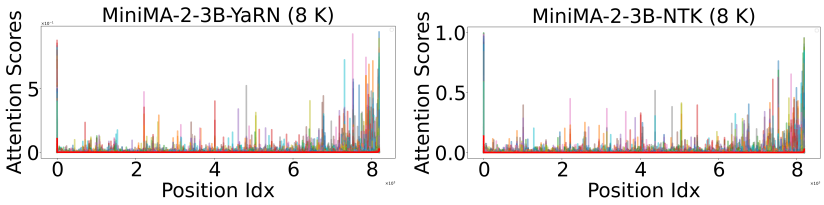


(d) YaRN on 8K sequences. (e) NTK on 8K sequences.

Figure 2: Attention distributions of RoPE, PI, YaRN, and NTK methods on 2K and 8K sequences. The red line represents the mean attention scores across all heads, layers, and examples. The other lines indicate the attention scores for each head in each layer.



(a) RoPE on 2K sequences. (b) RoPE on 8K sequences. (c) PI on 8K sequences.

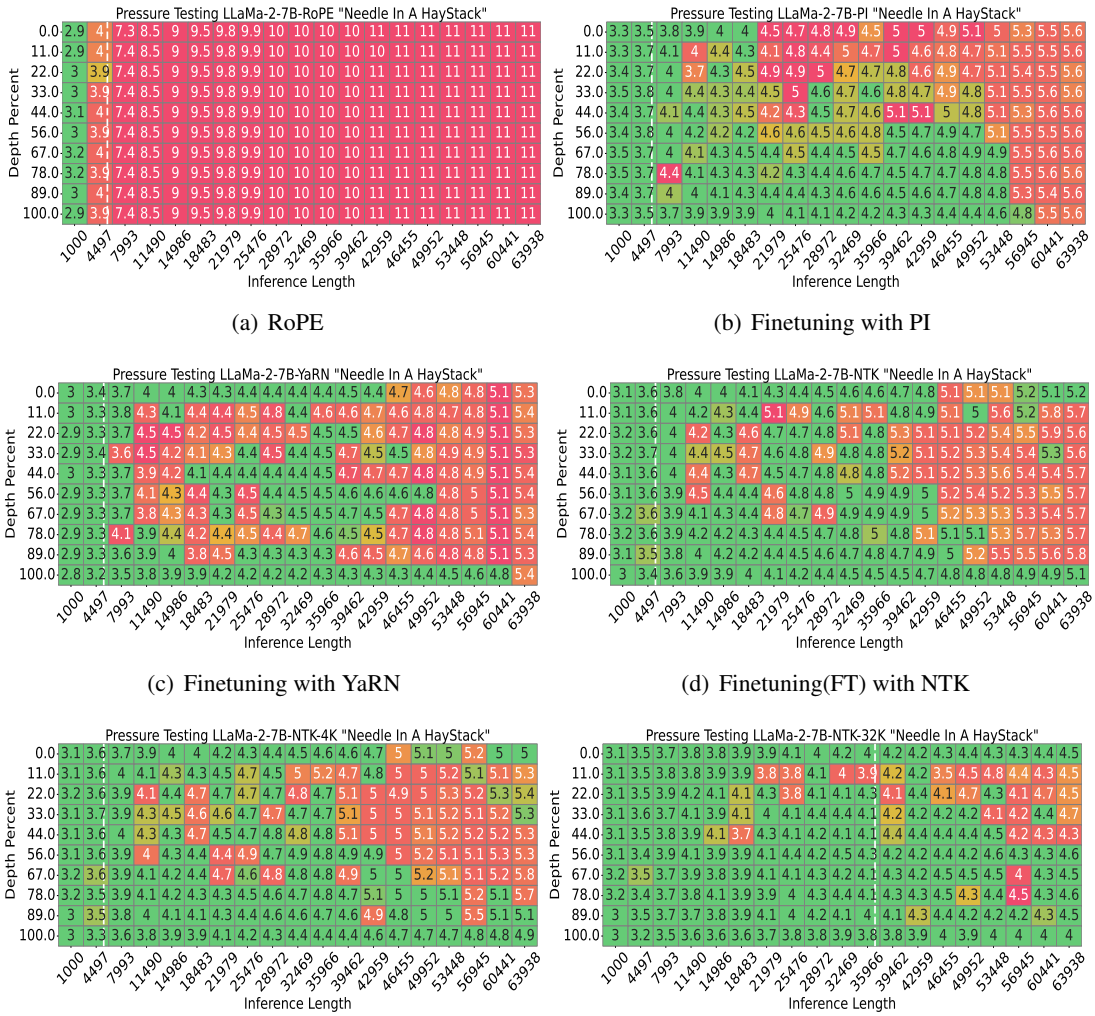


(d) YaRN on 8K sequences. (e) NTK on 8K sequences.

Figure 3: Attention distributions of RoPE, PI, YaRN, and NTK methods on 2K and 8K sequences on MiniMA-2-3B.

attention entropy. For example, at the same depth, the positions with errors are among the top-k in entropy; similarly, at the same length, the error positions also have high entropy. We hypothesize that the increase in attention entropy with longer test lengths is due to the train-short-and-test-long setting. During inference, the number of tokens handled by the self-attention mechanism far exceeds that during training. More tokens lead to more dispersed attention, i.e., higher uncertainty, causing a mismatch between training and inference.

A natural approach to lower attention uncertainty for enhancing extrapolation. A direct solution is to train on longer contexts,



(a) RoPE

(b) Finetuning with PI

(c) Finetuning with YaRN

(d) Finetuning(FT) with NTK

(e) FT on 4K with NTK from (d)

(f) FT on 32K with NTK from (d)

Figure 4: Performance comparison for the Needle-in-a-Haystack Test. The x-axis represents the length of the document, while y-axis indicates the depth percentage, showing the needle's position within the document. For instance, a position of 50% signifies that the needle is placed in the middle of the document. A red cell indicates that the model fails to recall the information in the needle, whereas a green cell indicates success. A white dashed line denotes the model's continual pretrain length. Each value in the cells signifies the mean attention entropy, with higher values reflecting more dispersed attention.

thereby increasing the number of attention tokens during training and reducing attention uncertainty. To validate our hypothesis, we finetune models on 4K and 32K training lengths with the same tokens on NTK. As shown in Figures 4(e) and 4(f), compared to models trained in short contexts, models trained in more extended contexts exhibited significantly lower attention uncertainty. For example, at length 63938, the attention entropy is generally below 5. The Needle test pass rates improved significantly, especially in longer testing contexts. Conversely, models trained with the same number of tokens but shorter context sizes showed little to no change in attention entropy, remaining similar to the original one (4(d)).

5 Conclusions

This paper provides the first thorough understanding of RoPE extensions for long-context LLMs from an attention perspective, evaluated on two widely-used benchmarks: Perplexity and Needle-in-a-Haystack. Extensive experiments demonstrate some valuable findings: 1) Compared to direct extrapolation, RoPE extensions can maintain the original training length attention patterns. 2) Large attention uncertainty leads to retrieval errors in needle testing in RoPE extensions. 3) Using longer continual pretraining lengths for RoPE extensions can reduce attention uncertainty and significantly enhance extrapolation in target LLMs.

Limitations

This paper primarily analyzes the widely-used decoder-only LM, LLaMa (Touvron et al., 2023). It does not include a validation study of encoder-decoder and encoder-only architectures.

Acknowledgements

We would like to thank the anonymous reviewers and meta-reviewer for their insightful suggestions. This work was supported by the National Natural Science Foundation of China under Grant U23B2055 and 62276077, and Shenzhen Science and Technology Program under Grant ZDSYS20230626091203008.

References

Zhangir Azerbayev, Edward Ayers, , and Bartosz Piotrowski. 2022. Proof-pile.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *ArXiv preprint*, abs/2309.16609.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. 2024. Deepseek llm: Scaling open-source language models with longtermism. *ArXiv preprint*, abs/2401.02954.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023a. Extending context window of large language models via positional interpolation. *ArXiv preprint*, abs/2306.15595.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023b. Extending context window of large language models via positional interpolation. *ArXiv preprint*, abs/2306.15595.

Yao Fu, Rameswar Panda, Xinyao Niu, Xiang Yue, Hannaneh Hajishirzi, Yoon Kim, and Hao Peng. 2024. Data engineering for scaling language models to 128k context. *ArXiv preprint*, abs/2402.10171.

Chi Han, Qifan Wang, Wenhan Xiong, Yu Chen, Heng Ji, and Sinong Wang. 2023. Lm-infinite: Simple on-the-fly length generalization for large language models. *ArXiv preprint*, abs/2308.16137.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Kaiokendev. 2023. Things i'm learning while training superhot. https://kaiokendev.github.io/til#extending-context-to-8k.

Ehsan Kamalloo, Nouha Dziri, Charles LA Clarke, and Davood Rafiei. 2023. Evaluating open-domain question answering in the era of large language models. *ArXiv preprint*, abs/2305.06984.

Greg Kamradt. 2023. Needle in a haystack - pressure testing llms. https://github.com/gkamradt/LLMTest_NeedleInAHaystack.

Zelin Li, Kehai Chen, Lemao Liu, Xuefeng Bai, Mingming Yang, Yang Xiang, and Min Zhang. 2024. Tf-attack: Transferable and fast adversarial attacks on large language models. *arXiv preprint arXiv:2408.13985*.

Xiaoran Liu, Hang Yan, Shuo Zhang, Chenxin An, Xipeng Qiu, and Dahua Lin. 2023. Scaling laws of rope-based extrapolation. *ArXiv preprint*, abs/2310.05209.

Xin Men, Mingyu Xu, Bingning Wang, Qingyu Zhang, Hongyu Lin, Xianpei Han, and Weipeng Chen. 2024. Base of rope bounds context length. *ArXiv preprint*, abs/2405.14591.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *ArXiv preprint*, abs/2402.06196.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2023. Yarn: Efficient context window extension of large language models. *ArXiv preprint*, abs/2309.00071.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2023. Code Llama: Open foundation models for code. *Preprint*, arXiv:2308.12950. ArXiv: 2308.12950.

Dan Su, Yan Xu, Genta Indra Winata, Peng Xu, Hyeondey Kim, Zihan Liu, and Pascale Fung. 2019. Generalizing question answering system with pre-trained language model fine-tuning. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 203–211, Hong Kong, China. Association for Computational Linguistics.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. 2021. Roformer: Enhanced transformer with rotary position embedding. *ArXiv preprint*, abs/2104.09864.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv preprint*, abs/2307.09288.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

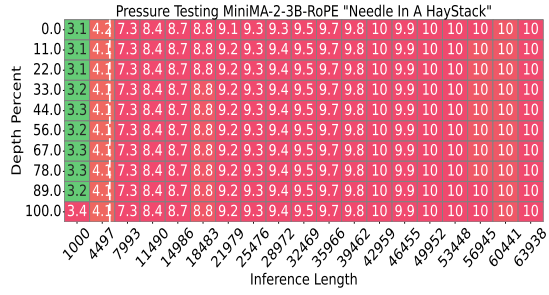
Chen Zhang, Dawei Song, Zheyu Ye, and Yan Gao. 2023. Towards the law of capacity gap in distilling language models. *ArXiv preprint*, abs/2311.07052.

Chen Zhang, Meizhi Zhong, Qimeng Wang, Xuantao Lu, Zheyu Ye, Chengqiang Lu, Yan Gao, Yao Hu, Kehai Chen, Min Zhang, et al. 2024a. Modification: Mixture of depths made easy. *arXiv preprint arXiv:2410.14268*.

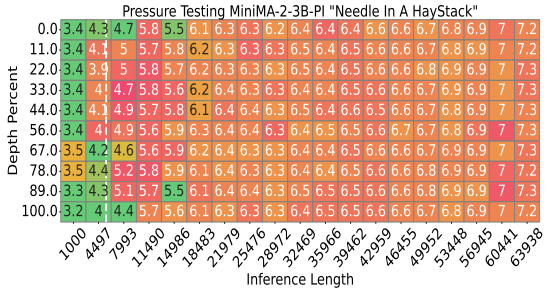
Shaoqing Zhang, Zhuosheng Zhang, Kehai Chen, Xinbei Ma, Muyun Yang, Tiejun Zhao, and Min Zhang. 2024b. Dynamic planning for llm-based graphical user interface automation. *arXiv preprint arXiv:2410.00467*.

Yikai Zhang, Junlong Li, and Pengfei Liu. 2024c. Extending llms' context window with 100 samples. *ArXiv preprint*, abs/2401.07004.

A Experimental Results on MiniMA-2-3B



(a) RoPE



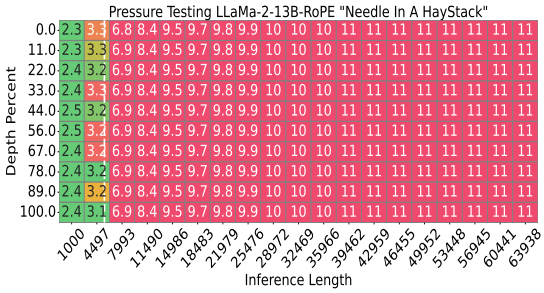
(b) Finetuning with PI

Figure 5: Performance comparison for the Needle-in-a-Haystack Test of MiniMa-2-3B.

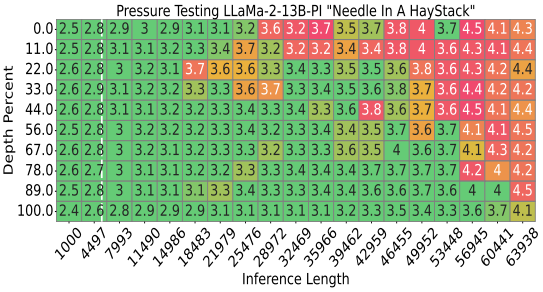
Similar to the analysis in § 4, the Needle-in-a-Haystack Test for MiniMa-2-3B also indicates that the locations of needle retrieval errors frequently align with areas of high attention entropy.

B Experimental Results on LLaMa-2-13B

Consistent with the analysis in § 3, we observe that the attention patterns fluctuate when RoPE is applied to 8K sequences, which exceed the training length. However, when using RoPE extensions, the attention distributions return to their original patterns for 8K sequences, as demonstrated in Figures 6.



(a) RoPE

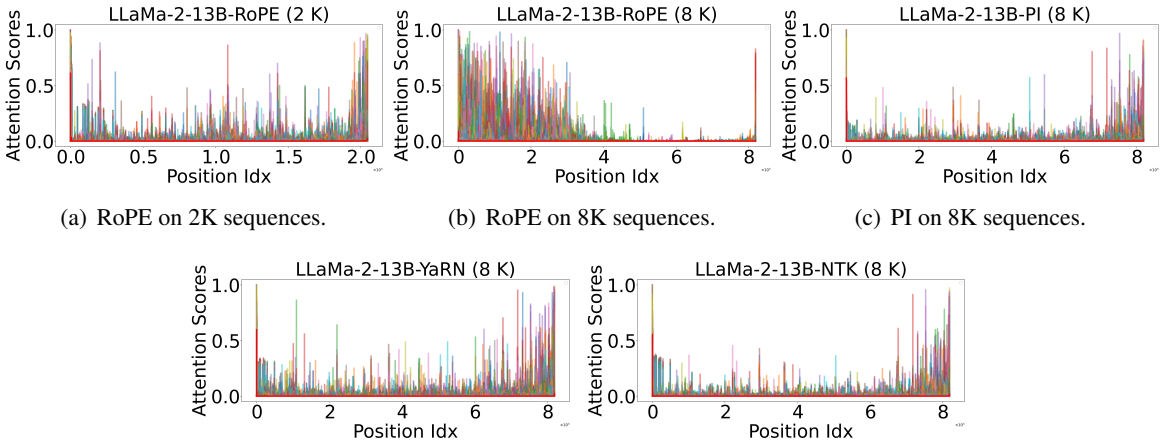


(b) Finetuning with PI

Figure 7: Performance comparison for the Needle-in-a-Haystack Test of LLaMa-2-13B.

Similar to the analysis in § 4, the Needle-in-a-Haystack Test for LLaMa-2-13B also indicates that the locations of needle retrieval errors frequently align with areas of high attention entropy.

C Detailed Calculation of Attention Entropy



(a) RoPE on 2K sequences. (b) RoPE on 8K sequences. (c) PI on 8K sequences.

(d) YaRN on 8K sequences. (e) NTK on 8K sequences.

Figure 6: Attention distributions of RoPE, PI, YaRN, and NTK methods on 2K and 8K sequences on LLaMa-2-13B.

Algorithm 1 Calculation of Attention Entropy

Input: model, prompt
Output: average attention entropy score
procedure ATTENTIONENTROPY($model, prompt$)
    Initialize $entropy\_list \leftarrow []$
    $output\_tokens \leftarrow [\,]$
    while not end of generation **do**
        $token, attention\_distribution \leftarrow$ GenerateTokenAndGetAttention($model, prompt + output\_tokens$)
        $output\_tokens$.append($token$)
        $entropy \leftarrow$ CalculateEntropy($attention\_distribution$)
        $entropy\_list$.append($entropy$)
    end while
    $average\_entropy \leftarrow$ Average($entropy\_list$)
    return $average\_entropy$
end procedure
function CALCULATEENTROPY($distribution$)
    $entropy \leftarrow 0$
    for all p in $distribution$ **do**
        if $p > 0$ **then**
            $entropy \leftarrow entropy - p\log(p)$
        end if
    end for
    return $entropy$
end function