Context Length Extension via Generalized Extrapolation Scale

Linhan Li and **Huaping Zhang**[*]
School of Computer Science
Beijing Institute of Technology
Beijing, China
kevinzhang@bit.edu.cn

Abstract

Context length expansion of transformer models is considered a key challenge, especially when handling context beyond the training length during inference stage. In this paper, we propose **Ge**neralized extrapolatio**N** scal**E** (GeNE), a straightforward and effective method applied to the interpolate function of positional embeddings to achieve training short, test long. Experimental results show that GeNE notably improves long context language modeling. By randomly scaling the extrapolation ratio during the finetuning, GeNE achieves stable extrapolation on 64k contexts by training on 16k length. Further, the instruction following Llama2 model based on GeNE achieved competitive results compared with other open-source models of the same parameter scale. Our code is available at https://github.com/LhLi-QED/GeNE.

1 Introduction

Large language models (LLMs) based on transformer architectures (Vaswani et al., 2017) have been shown to have excellent performance in practice on a variety of natural language processing tasks. However, sequences are usually truncated to be less than a predefined value during pre-training, forming a fixed context window and becoming one of the limitations of LLMs in the application of long context tasks. One of the challenges in extending the context window of LLMs is the extrapolation of positional embedding (PE), so it is necessary to explore efficient and balanced extrapolation methods.

Rotary Position Embedding (RoPE) (Su et al., 2024) encodes position information by implementing rotation transformations at varying frequencies across different dimensions, and its superior performance has been demonstrated in models such as GPT-NeoX (Black et al., 2022), LLaMA (Touvron et al., 2023a), Llama2 (Touvron et al., 2023b) and Mistral (Jiang et al., 2023). Previous work on the extension of RoPE mainly focuses on artificially designing extrapolation functions, such as position interpolation (PI) (Chen et al., 2023b), NTK-aware (bloc97, 2023) and YaRN (Peng et al., 2023). Artificially designed extrapolation functions exhibit good generality, but expressing the extrapolation in greater detail can prove challenging. CLEX (Chen et al., 2023a) parameterizes the extrapolation function by a neural ordinary differential equation, thus establishing a continuous extrapolation for each frequency of the RoPE with remarkable improve-

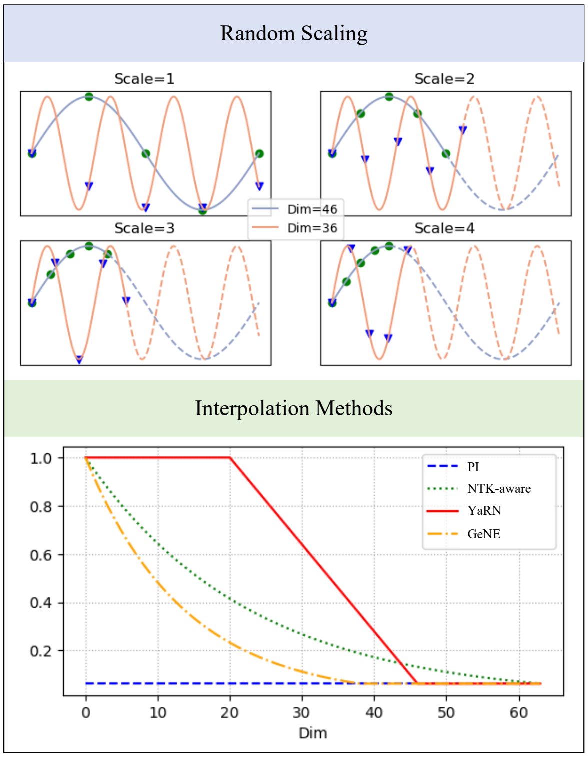

Figure 1: Overview of our approach. In each training batch, GeNE randomly increases the extrapolated scale to simulate the patterns of longer contexts. And GeNE uses a smaller critical dimension compared to previous extrapolation functions.

[*]Corresponding author.

Findings of the Association for Computational Linguistics ACL 2024, pages 4211–4218
August 11-16, 2024 ©2024 Association for Computational Linguistics

ment.

In this paper, we present GeNE, an advanced and simple method for extrapolating context window sizes. We argue that for RoPE-based LLMs, there are two sufficient conditions for context length extrapolation: (1) the model needs to be trained to adapt to denser position embeddings, (2) the position index will not lead to out-of-distribution PE when context length extends. For the first condition, in order to achieve "train short, test long", we propose Batch-wise Random Scaling (BRS), a simple but effective method that can adapt the model to longer sequence PE patterns even finetuning with short sequence. Additionally, we utilize the "NTK-by-parts" style (Peng et al., 2023) extrapolation function to satisfy condition (2) to avoid any out-of-distribution positional embedding.

We evaluated the effectiveness of GeNE in three aspects: (1) test splits of proof-pile (Azerbayev et al., 2022) and PG19 (Rae et al., 2020) for language modeling, and (2) LongBench (Bai et al., 2023) for downstream tasks (3) passkey retrieval (Mohtashami and Jaggi, 2024) for retrieval capability. GeNE efficiently extrapolates the context length and achieves competitive results in language modeling. Specifically, models based on GeNE trained on 16k lengths can extrapolate the length to 64k in the inference phase. The ablation experiment further proves the effectiveness of our method. Additionally, compared with open-source models of the same parameter scale, GeNE's instruction finetuning model has also achieved comparable results. Finally, additional experiments reveal that setting an appropriate critical dimension enables the model to perform better in the maximum achievable context length during the inference phase. In summary, our contributions are as follows:

1. We propose GeNE, an advanced method for extending the context window of RoPE-based LLMs, and verify its effectiveness through extensive experiments.

2. The results of our experiments indicate that random scaling has a more pronounced effect on the expansion of context length for GeNE.

2 Related Work

Relative Positional Embeddings for Long Context. Since relative positional embeddings usually have better extrapolation performance than absolute positional embeddings, it is widely used in large language models. T5 (Raffel et al., 2020), ALiBi (Press et al., 2022), KERPLE (Chi et al., 2022) and Sandwich (Chi et al., 2023) achieved relative position encoding by adding a bias term to attention score. Unlike position embeddings based on biases of attention scores, RoPE (Su et al., 2024) is applied to the vectors of queries and keys and achieves relative position embeddings through inner products.

Interpolation and Extrapolation of RoPE. To further achieve expansion of the context window of LLMs with a small amount of fine-tuning, extrapolation or interpolation of RoPE is an effective method. Interpolating the position index (Chen et al., 2023b) or modifying the base value of RoPE (emozilla, 2023; Liu et al., 2023; Roziere et al., 2023) can facilitate a degree of context length extension. In addition, (Peng et al., 2023; Pal et al., 2023) implement interpolation or extrapolation at various scales for different feature dimensions. Finetuning with a larger or smaller rope base can also significantly extend the context length (Roziere et al., 2023; Liu et al., 2023). CLEX (Chen et al., 2023a) continuously scales the extrapolated factor through neural ordinary differential equations. Especially, (Su, 2023; Jin et al., 2024) found that ensuring the consistency of local relative position embeddings with pre-trained embedding also facilitates extrapolation and achieves longer text and lower perplexity. Different from the above methods, RandPos (Ruoss et al., 2023) and PoSE (Zhu et al., 2024) mainly focus on the position index, and they are also compatible with those methods above.

3 Preliminaries

3.1 RoPE

The basic idea of RoPE (Su et al., 2024) is to fuse the position information into the query and key vectors, and encode the relative position through the dot product operation of self-attention. Denote an input sequence of length N as $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{N-1} \in \mathbb{R}^d$, where d is the embedding dimension. We denote the query and key vectors of $\mathbf{x}_n$ as $\mathbf{q}_n = \mathbf{W}_q\mathbf{x}_n, \mathbf{k}_n = \mathbf{W}_k\mathbf{x}_n$, respectively. RoPE groups the embedding dimensions pairwise and implements a 2D rotation transformation in each group:

$$f(\mathbf{u}_n, n) = [\mathbf{R}(n\theta_i)\mathbf{u}_{n,i:i+1}]_{i=0}^{\frac{d}{2}-1}, \theta_i = b^{-\frac{2i}{d}} \quad (1)$$

where $\mathbf{u}_n$ represents the nth query or key vector. $\mathbf{R}(n\theta_i)$ is a 2D rotation matrix with rotation angle $n\theta_i$. b is a hyperparameter named the base of RoPE, which is set to 10^4 in Llama2 (Touvron et al., 2023b).

3.2 Extrapolation of RoPE

Formally, let N_0, N be the pre-trained context length and the finetuning context length, respectively, and the ratio of extrapolation be $r = \frac{N}{N_0}$. Previous approaches (Chen et al., 2023b; bloc97, 2023) implement length extrapolation by manually setting a extrapolation function $f(r, i)$ in $\theta_i(r) = f(r, i)(b^{\frac{2i}{d}})^{-1}$ to achieve extrapolation. For example, we have $f^{\text{PI}}(r, i) = r^{-1}, f^{\text{NTK}}(r, i) = r^{-\frac{2i}{d-2}}$. Moreover, further study shows that different wavelengths $\lambda_i = 2\pi/\theta_i$ require different extrapolation functions. For those dimensions that satisfy $\lambda_i \le N_0$, they can be extrapolated directly, while for those dimensions whose wavelength satisfies $\lambda_i > N_0$, interpolation is required because extrapolation will lead to out-of-distribution encodings. In particular, the dimension with the smallest value satisfy $\lambda_i > N_0$ is referred as critical dimension (Liu et al., 2023):

$$\beta_{\text{critic}} = 2 \left\lceil \frac{d}{2} \log_b \frac{N_0}{2\pi} \right\rceil \tag{2}$$

YaRN extends the context window beyond 128k by interpolating for dimensions larger than critical dimension and using extrapolation at different scales for other parts (Peng et al., 2023).

4 GeNE

4.1 Batch-wise Random Scaling

For interpolation-based extrapolation methods, when the test context length gets larger, the position embedding will get denser, especially for those parts larger than the critical dimension, which could lead to higher perplexity. To address this problem we simply randomly scale the extrapolation ratio during finetuning to make the model adapt to a denser position embedding pattern.

Concretely, to adapt $\theta^{\text{GeNE}}(r)$ to a wider range of extrapolation ratios, we randomly increase r in each batch so that it exceeds $\frac{N_{\text{Train}}}{N_0}$. Formally, we denote the predefined random scaling factor as s, and the extrapolation ratio r of each batch is reset to $\widetilde{s}r$, where $\widetilde{s}$ is sampled from $\{1, \ldots, s\}$ with a uniform distribution. And the scaling factor of RoPE is defined as $S = r \cdot s$. The extrapolation ratio is set to $r \cdot s$ to enable the model to adapt to a denser position embedding pattern, while an extrapolation ratio of r ensures that the model does not forget the maximum range of position embedding. When the extrapolation ratio lies between these two values, it serves as a gradual transition.

4.2 Interpolating Method of GeNE

We adopt "NTK-by-parts" (Peng et al., 2023) style interpolation to initialize $r^{\xi(i)}$ considering that using artificially designed for initialization can speed up convergence (Chen et al., 2023a) and ensure that no out-of-distribution position embedding occurs:

$$\xi(i) = \begin{cases} \frac{2i}{\beta_{\text{critic}}}, & 0 \le i \le \frac{\beta_{\text{critic}}}{2} \\ 1, & \frac{\beta_{\text{critic}}}{2} < i \le \frac{d}{2} - 1 \end{cases} . \tag{3}$$

And we denote that $f^{\text{GeNE}}(r, i) = r^{-\xi(i)}$. Furthermore, we conducted more extensive experiments on the selection of the critical dimension. Specifically, we defined the critical dimension as

$$\beta_{\text{critic}} = 2 \left\lceil \frac{d}{2} \log_b \frac{N_0}{2m\pi} \right\rceil \tag{4}$$

where m is a hyperparameter. For more details, please refer to Section 5.3.

We also consider trainable parameters for the extrapolation function so that each attention head can flexibly and independently adjust the extrapolation scale. Concretely, for a transformer model with L layers and H attention heads, we define the vector of rotational angular velocity parameterized by $\{\psi_{l,h}(i), \phi_{l,h}(i) \in \mathbb{R}\}$ as

$$\theta^{\text{GeNE-Param}}(r) = (e^{\psi(i)} \cdot r^{\xi(i)} b^{\frac{2i}{d}} + \phi(i) \cdot r)^{-1} \tag{5}$$

where $l \in \{0, \ldots, L-1\}$, $h \in \{0, \ldots, H-1\}$, $i \in \{0, \ldots, d/2-1\}$, and the parameters $\{\phi_{l,h}(i), \psi_{l,h}(i)\}$ are initially set to zero.

5 Experiments

In this section, we evaluate GeNE on long context language modeling and long context benchmarks. We compared GeNE with other methods include PI (Chen et al., 2023b), Dynamic-NTK (emozilla, 2023), YaRN (Peng et al., 2023) and CLEX (Chen et al., 2023a). by fine-tuning on Llama2-7B model (Touvron et al., 2023b). In addition, we conduct ablation experiments on batch-wise random scaling (BRS) and the interpolating method of GeNE. For practical tasks, we evaluate GeNE on the LongBench, comparing it with other open-source models in long context tasks.

Table 1: Perplexity of different methods on the test splits of proof-pile (Azerbayev et al., 2022) and PG19 (Rae et al., 2020). The better performance are in **bold**. We conduct the experiment three times with different random seeds. On long context language modeling, our GeNE achieves competitive results compared with CLEX which is the previous state-of-the-art method.

Method	**Proof-pile**				
	4k	**8k**	**16k**	**32k**	**64k**
PI	2.94±0.11	2.72±0.10	2.57±0.15	2.59±0.13	3.48±0.10
Dy-NTK	3.10±0.22	2.71±0.19	2.56±0.17	2.47±0.16	2.67±0.16
YaRN	**2.84±0.09**	2.63±0.08	2.54±0.11	2.39±0.13	2.44±0.13
CLEX	3.26±0.13	**2.62±0.10**	**2.534±0.12**	**2.37±0.08**	2.43±0.14
GeNE(ours)	2.91±0.14	2.64±0.10	2.535±0.13	2.38±0.09	**2.40±0.11**

Method	**PG19**				
	4k	**8k**	**16k**	**32k**	**64k**
PI	7.82±0.13	7.57±0.15	7.45±0.14	7.98±0.12	12.30±0.11
Dy-NTK	8.27±0.18	7.70±0.15	7.65±0.16	7.63±0.17	9.43±0.21
YaRN	7.76±0.09	7.63±0.10	7.55±0.09	7.57±0.12	8.42±0.12
CLEX	8.23±0.11	7.60±0.09	7.46±0.10	7.56±0.14	8.12±0.15
GeNE(ours)	**7.73±0.10**	**7.54±0.08**	**7.43±0.12**	**7.48±0.10**	**7.53±0.14**

Table 2: We performed ablation experiments with the same random seed. NTK† denotes using f^{NTK} as mentioned in Section 3.2 instead of f^{GeNE} as initialization while keeping other methods unchanged.

Method	**Proof-pile**				
	4k	**8k**	**16k**	**32k**	**64k**
GeNE-Param	2.812	2.602	2.453	2.343	2.296
GeNE	2.8132	2.603	2.455	2.345	2.298
w/o BRS	2.824	2.613	2.456	2.350	2.755
NTK†	2.814	2.604	2.459	2.349	2.309
vanilla NTK	2.828	2.612	2.464	2.395	2.803

Method	**PG19**				
	4k	**8k**	**16k**	**32k**	**64k**
GeNE-Param	7.802	7.560	7.433	7.420	7.670
GeNE	7.801	7.568	7.438	7.421	7.667
w/o BRS	7.846	7.614	7.461	7.581	10.861
NTK†	7.826	7.581	7.454	7.446	7.685
vanilla NTK	7.964	7.650	7.459	7.596	10.122

Dataset For long text language modeling, our finetune dataset is collected from RedPajama-arxiv and RedPajama-book (Computer, 2023), and truncated to the length of 16k for every sample. We use the test splits of proof-pile (Azerbayev et al., 2022) and PG19 (Rae et al., 2020) for evaluation. Specifically, we sample 20 items from each of the two datasets and ensure each sample has at least 64k tokens. For instruction fine-tuning, we conduct instruction fine-tuning based on the filtered UltraChat (Ding et al., 2023; Tunstall et al., 2023) dataset.

Finetune Settings For long context modeling finetuning, we use the AdamW (Loshchilov and Hutter, 2019) with a learning rate of 2×10^{-5} for all the above models. We set a global batch size of 128, and finetune on 8 × A100 GPUs for 300 steps with a linear warmup of 100 steps. Additionally, we use Deepspeed Zero (Rajbhandari et al., 2020) and FlashAttention-2 (Dao, 2023) for acceleration.

5.1 Evaluation on Language Modeling

We fine-tune all of the models based on a 16k context length, and we evaluate sliding window perplexity (Peng et al., 2023) ($S_w = 1024$) for language modeling with the sequence length from 4k to 64k. The results are shown in Table 1. We adopt the scaling factor of RoPE $S = 16$ for YaRN (Peng et al., 2023), CLEX (Chen et al., 2023a) and GeNE, as for Dynamic NKT (emozilla, 2023), we set $\alpha = 2$. Our GeNE achieved competitive results in both long-text and short-text language modeling, and in particular, our GeNE outperformed all baseline models on PG19. And it can maintain the perplexity reduction in 32k text language modeling and has better stability in 64k.

We further performed the ablation experiment on the language modeling, and the results are shown in Table 2. For ablation of BRS, we fix the extrapolation ratio as $r \cdot s$. Ablation results showed that both BRS and GeNE-Param could improve the performance of language modeling. The improvement of

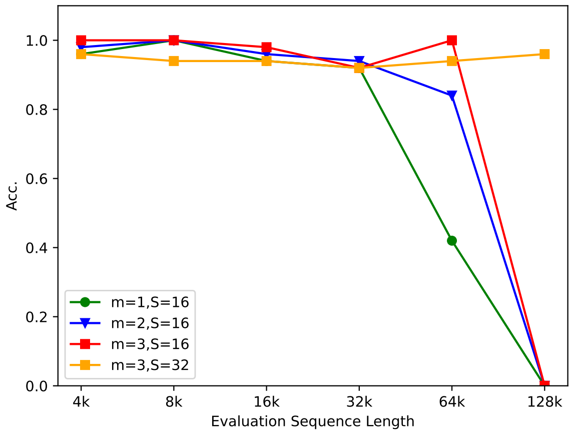

Figure 2: Retrieval accuracy of passkey. Here, $S = 16$ indicates a target extrapolation length of 64k, while $S = 32$ corresponds to 128k.

BRS implementation becomes more notable with the increasing extrapolation length. And previous method such as NTK can also benefit from our approach. Considering that the pre-trained position embedding is consistent for each layer and attention head for Llama2, we believe that this makes the model fully adapted to this pattern so that the model benefits less from diverse position embeddings.

5.2 Evaluation on Benchmark

We further conduct finetune with the UltraChat (Ding et al., 2023; Tunstall et al., 2023) dataset for 400 steps based on the checkpoint of GeNE finetuned on 4k length in language modeling to make it capable of instruction following tasks. We evaluate Llama2-7b-chat (Touvron et al., 2023b), CodeLlama-7b-Instruct (Roziere et al., 2023), longchat-7b-v1.5-32k (Li et al., 2023), vicuna-7b-v1.5-16k (Zheng et al., 2023) and our GeNE on LongBench (Bai et al., 2023) and report the average scores on each type of task in Table 3.

5.3 Evaluation on Synthetic Retrieval Tasks

Synthetic retrieval tasks such as passkey retrieval (Mohtashami and Jaggi, 2024) can measure the model's maximum context length. In this section, we compare the maximum context length achievable under different hyperparameter settings of GeNE finetuned on a context length of 16k.

For passkey retrieval, we compare the retrieval accuracy under different values of the hyperparameter m in Equation 4. Our testing context lengths range from 4k to 128k. For each length, we conducted 50 tests. The results are shown in Figure 2. The results indicate that the selection of the critical dimension affects the model's actual maximum length. When we set $m = 3$, the model achieves an accuracy rate of over 90% within the target length. As for $m < 3$, the accuracy rate declines at a length of 64k.

6 Conclusion

In this paper, we introduce GeNE, a simple and effective context window expansion method. GeNE stably extrapolates the context window from 4k to 64k and is competitive in long-text language modeling compared with the current state-of-the-art method CLEX. Experiments demonstrate that compared with trainable extrapolation functions, random scaling has a more notable impact on extrapolation.

7 Limitations

Although our experiments based on GeNE found that random scaling has a notable impact, more research is needed to determine whether this is a general conclusion. For example, (Chen et al., 2023a) found that trainable parameters are still necessary for continuous dynamic extrapolation. We did not perform significance tests for cases with small gaps because collecting statistics requires finetuning the model multiple times, causing higher computational costs.

In addition, (Su, 2023; Jin et al., 2024) show that another sufficient condition for modeling long-text languages is to keep the code distance of local position encoding unchanged. This method can also achieve long text language modeling without finetuning, while our approach still requires around 0.6B of tokens for finetuning.

8 Ethics Statement

This research is mainly about how to effectively extend the context window length of language models, which in itself does not have the possibility of directly posing any social risks. Context length extrapolation is a key issue in large language models, aiming to create more powerful language models. However, these language models can be abused by humans, which is a common problem currently faced in the field of NLP.

References

Z. Azerbayev, E. Ayers, and B. Piotrowski. 2022. Proof-pile.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2023. Longbench: A bilingual, multitask benchmark for long context understanding. *arXiv preprint arXiv:2308.14508*.

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. 2022. Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*.

bloc97. 2023. Ntk-aware scaled rope allows llama models to have extended (8k+) context size without any fine-tuning and minimal perplexity degradation.

Guanzheng Chen, Xin Li, Zaiqiao Meng, Shangsong Liang, and Lidong Bing. 2023a. Clex: Continuous length extrapolation for large language models. *arXiv preprint arXiv:2310.16450*.

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

Ta-Chung Chi, Ting-Han Fan, Peter J Ramadge, and Alexander Rudnicky. 2022. Kerple: Kernelized relative positional embedding for length extrapolation. In *Advances in Neural Information Processing Systems*, volume 35, pages 8386–8399. Curran Associates, Inc.

Ta-Chung Chi, Ting-Han Fan, Alexander Rudnicky, and Peter Ramadge. 2023. Dissecting transformer length extrapolation via the lens of receptive field analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13522–13537, Toronto, Canada. Association for Computational Linguistics.

Together Computer. 2023. Redpajama: An open source recipe to reproduce llama training dataset.

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, Singapore. Association for Computational Linguistics.

emozilla. 2023. Dynamically scaled rope further increases performance of long context llama with zero fine-tuning.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Hongye Jin, Xiaotian Han, Jingfeng Yang, Zhimeng Jiang, Zirui Liu, Chia-Yuan Chang, Huiyuan Chen, and Xia Hu. 2024. Llm maybe longlm: Self-extend llm context window without tuning. *arXiv preprint arXiv:2401.01325*.

Dacheng Li, Rulin Shao, Anze Xie, Ying Sheng, Lianmin Zheng, Joseph Gonzalez, Ion Stoica, Xuezhe Ma, and Hao Zhang. 2023. How long can context length of open-source LLMs truly promise? In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Xiaoran Liu, Hang Yan, Shuo Zhang, Chen An, Xipeng Qiu, and Dahua Lin. 2023. Scaling laws of rope-based extrapolation. *arXiv preprint arXiv:2310.05209*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Amirkeivan Mohtashami and Martin Jaggi. 2024. Random-access infinite context length for transformers. *Advances in Neural Information Processing Systems*, 36.

Arka Pal, Deep Karkhanis, Manley Roberts, Samuel Dooley, Arvind Sundararajan, and Siddartha Naidu. 2023. Giraffe: Adventures in expanding context lengths in llms. *arXiv preprint arXiv:2308.10882*.

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

Ofir Press, Noah Smith, and Mike Lewis. 2022. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. 2020. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. 2023. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Anian Ruoss, Grégoire Delétang, Tim Genewein, Jordi Grau-Moya, Róbert Csordás, Mehdi Bennani, Shane Legg, and Joel Veness. 2023. Randomized positional encodings boost length generalization of transformers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1889–1903, Toronto, Canada. Association for Computational Linguistics.

Jianlin Su. 2023. Rectified rotary position embeddings. `https://github.com/bojone/rerope`.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

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

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Dawei Zhu, Nan Yang, Liang Wang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. 2024. PoSE: Efficient context window extension of LLMs via positional skip-wise training. In *The Twelfth International Conference on Learning Representations*.

A Additional Results

Table 3: Evaluation average scores on LongBench. The best performance and suboptimal performance are **bolded** and underlined respectively.

Model	Single-Doc QA	Multi-Doc QA	Sum.	Few-shot	Synthetic	Code
Llama2-7b-chat-4k	22.28	18.35	18.45	51.45	6.56	55.15
CodeLlama-7b-Instruct-16k	**33.79**	13.94	21.66	57.27	7.00	**60.37**
vicuna-7b-v1.5-16k	32.32	18.87	**23.57**	56.37	5.00	45.14
longchat-v1.5-7b-32k	28.53	**19.95**	22.02	50	**11.77**	52.73
GeNE-llama2-7b-4k	27.98	16.81	23.33	**58.05**	2.82	56.43

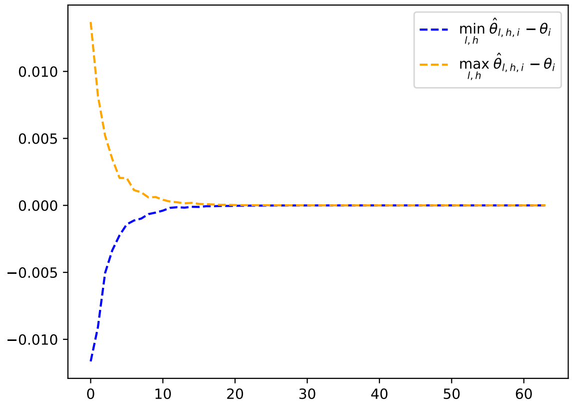

Figure 3: Visualization of the difference in trainable extrapolation functions and initialization in GENE-Param.

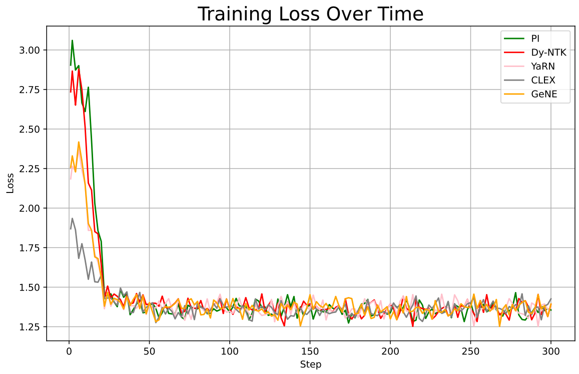

Figure 4: Training loss curves of various baselines.