Stable Random Sampling (SRS): A New Method to Refine Causal Masking in Decoder-Only Transformer

Shuhao Zhang[1,*], Jiayi Yu[2], Jiarui Li[3]

{U202142800@xs.ustb.edu.cn[1], yu_jia_yi@sjtu.edu.cn[2], 22009200659@stu.xidian.edu.cn[3]}

School of Economics and Management, University of Science and Technology Beijing, Beijing, 100083, China[1]
UM-SJTU Joint Institute, Shanghai Jiaotong University, Shanghai, 200240, China[2]
School of Cyber Engineering, Xidian University, Xian, 710126, China[3]
*corresponding author

Abstract. In current language modelling, the decoder-only Transformer architecture with causal masking has become a cornerstone, demonstrating exceptional performance across various tasks. However, we have identified two significant limitations: First, causal masking presents a substantial obstacle to further optimizing overall model efficiency, particularly in handling long contexts. Second, traditional optimization of causal masking struggles with uneven attention distribution and the inability to encode absolute positional information, limiting their effectiveness in position-sensitive tasks. In this work, we propose the Stable Random Sampling (SRS) algorithm, a novel method to address both limitations by refining the causal masking process. SRS introduces a pseudo-attention mask to balance attention distributions for performance refinement and incorporates random sampling and Locality-Sensitive Hashing (LSH) in causal masking part for efficient processing, reducing time complexity of this part to $O(n)$. The effectiveness of SRS is validated both theoretically and empirically. Our pre-training ablation experiments demonstrate that SRS module virtually enhances the performance of causal masking while each functional part of it relatively improves efficiency and effectiveness towards different sizes of tasks, on average showing a 30% reduction in training time and a 50% decrease in loss rate compared to traditional methods. Then, we further show in empirical experiments that SRS makes the time 2x faster on a single attention layer than FlashAttention and exhibits 33% lower perplexity compared to HyperAttention on average in which requires highly positional sensitive scenarios. Moreover, SRS naturally supports efficient processing of long sequences and can be easily integrated with existing attention optimization techniques.

Keywords: Decoder-only Transformer, Causal Masking, Random Sampling, Positional Information.

1 Introduction

Today, large language models (LLMs) [1-6] have rapidly developed and achieved impressive success in many areas, especially in the field of NLP [7-10]. The underlying architecture of LLMs is the decoder-only Transformer, with causal masking [11, 12] being an essential component. This structure enables LLMs to perform well in generative tasks[3,5], such as text generation and dialogue systems. Consequently, models like GPT-3 and GPT-4 have been the most rapid and prosperous. Optimization efforts for decoder-only Transformers have never

CONF-MLA 2024, November 21, Adana, Turkey

DOI 10.4108/eai.21-11-2024.2354592

ceased. A novel self-attention layer called 'HyperAttention'[13] has been proposed, which reduces the time complexity of self-attention layer calculations in previous decoder-only Transformers from $O(n^2)$ to $O(nlogn)$. Additionally, Stable Mask adjusts causal masking through its unique pseudo-attention score matrix, leading to more balanced attention allocation within the context and addressing the shortcomings of relative positional encoding. Despite this impressive success, we identify two significant issues that need to be solved within that improvement

The first issue is the algorithm efficiency problem of causal masking. Causal masking [11, 12] is the essential component of a decoder-only transformer. It enables Transformer models to better handle tasks that require consideration of temporal order, generating more natural, coherent, and reasonable text.[14] However, causal masking poses a significant obstacle to the optimization of self-attention algorithms [15]. For instance, HyperAttention [13] achieves near-linear complexity for context when the sequence length n=131k through a series of algorithmic optimizations, resulting in more than a 50-fold increase in both forward and backward propagation speed. However, the recursive algorithmic aspect of causal masking still has a significant resistance to optimizing the overall model efficiency: when causal masking is used, the optimization speed of the algorithm plummets from the original 50x to 5x [13]. To be more specific, the causal masking part increased time expenses and reduced the performance of the original algorithm. Therefore, through our research on this point, we wanted to find a method for optimizing the time complexity during the process of causal-masking.

The second restriction is that we found that many other traditional causal masking optimization attempts cannot solve the following two problems while maintaining calculation efficiency [16]: (1) Uneven attention distribution: This is a drawback of the softmax function [17,18], as the output of the softmax function sums to 1. These characteristic forces some attention to be allocated to unimportant content, such as punctuation [19], leading to decreased efficiency and accuracy when the model processes long contexts. As a result, the model has certain limitations. (2) Inability to encode absolute positional information [20, 21]: Relative positional encoding does not support the application of the model in position-sensitive tasks. For example, Stable Mask [16], which can address these two issues to help improve accuracy performance in a long context, still faces the dilemma when trying to alleviate accuracy performance while reducing computational demand. Consequently, we aim to solve the accuracy problem while addressing efficiency issues

In order to meet these challenges, through careful study of the optimization theory of Hyper attention algorithm, and the core ideas of the stable mask algorithm [16], we propose an algorithm named Stable Random Sampling Algorithm (SRS) to replace the recursive algorithm used in the previous causal masking in this paper. Our algorithm aims to optimize the time complexity and to improve the allocation mechanism of the attention mechanism, and finally achieve an algorithm with $O(n)$ time complexity and can well improve the uneven allocation of attention and the inability to recognize the absolute positional encoding. Finally, we realize an algorithm with $O(n)$ time complexity that can improve the uneven allocation of attention and the inability to recognize absolute positional encoding. The goal is to make HyperAttention significantly more efficient and productive in dealing with long contexts. The algorithm consists of three parts: (1) firstly, the Pseudo Attention mask matrix [16] is introduced to participate in the computation of the attention matrix, (2) on the causal masking processed matrix, LSH is applied to each row to recognize the heavy entries and the Scale Sampling algorithm for the overall estimation of the remaining light entries. (3) Finally, the remaining pseudo attention score is added to the overall computation of the D matrix.

The effectiveness of SRS has been confirmed by us through ablation experiment and empirical experiments on LLM, indicating it can be conveniently and explicably integrated into all kinds of decoder-only transformers in replacement of traditional causal masking.

Our core contributions can be summarized as follows:

1. We identified two issues in the optimization of the causal masking part in decoder-only transformers: the efficiency in optimizing transformers with causal masking and the inability to reduce the calculational demand while accurately capturing positional information.

2. We propose SRS, an effective and integrable solution to settle both issues by delicately modifying the causal mask in an efficient way.

3. We validate the effectiveness of SRS with pre-training ablation experiments and empirical experiments across various tasks and sizes of datasets.

4. We modify SRS with a hardware-efficient method for further practical application.

2 Methodology

To optimize the causal masking process in HyperAttention models, we propose the Stable Random Sampling (SRS) method. As shown in Fig.1, the algorithm first applies StableMask to the attention matrix A to ensure accurate causal masking, then leverages Kernel Locality-Sensitive Hashing (Kernel-LSH) and selective sampling techniques to obtain the diagonal matrix D_C. Specifically, the random sampling method replaces recursive call in the original HyperAttention algorithm, which reduces the time complexity of from $O(nlgn)$ to $O(n)$.

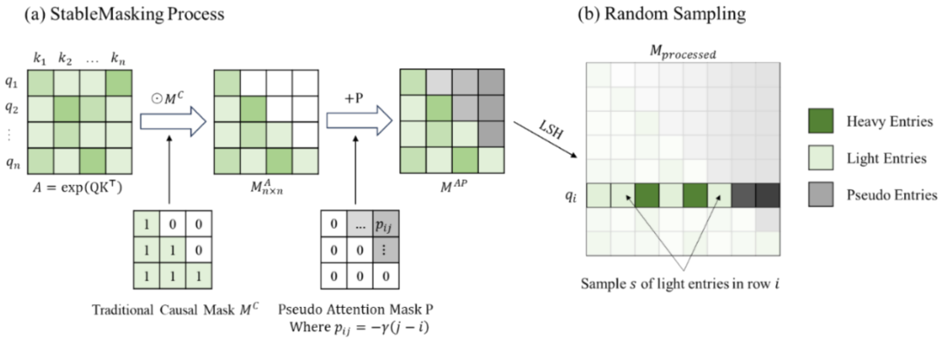


Fig. 1. (a) The StableMasking process, where attention matrix A is first element-wise multiplied by traditional causal mask M^C, then added with pseudo attention mask P to get M^{AP}. (b) The Random Sampling process, where M^{AP} is first processed by Kernel LSH to get $M_{processed}$, then for each row q_i in $M_{processed}$, the sum is calculated by adding all the heavy entries, s samples of light entries, and all the pseudo entries.

2.1 StableMasking

Algorithm 1: StableMask, StableMasking Process

```
**input**: Matrices $Q, K \in \mathbb{R}^{n\times d}$, Pseudo Attention Matrix $P$, StableMask $M^C$
Initialize Attention Matrix: $A = exp(QK^T)$
Apply StableMask: $M^{AP} = A \odot M^C + P$
**return** $M^{AP}$, $\boldsymbol{A^{SM}} = \boldsymbol{M^{AP}} \odot M^C$
```

From the StableMask method presented by Yin et al. [16], a lower binary triangular matrix M^C is generated to serve as the causal mask and an upper triangular P is then generated as the pseudo attention matrix, where each element p_{ij} is computed as

$$p_{ij} = -\gamma(j - i) \tag{1}$$

Here, γ is a constant scaling factor that helps maintain stability in the attention scores by introducing a temporal bias.

The two masks are represented below.

$$M^C = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}, P = \begin{pmatrix} 0 & p_{11} & p_{12} & \cdots & p_{1n} \\ 0 & 0 & p_{22} & \cdots & p_{2n} \\ 0 & 0 & 0 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & p_{nn} \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix}$$

The intermediate masked matrix M^{AP} is obtained by applying the two masks.

$$M^{AP} = (A \odot M^C) + P \tag{2}$$

Another layer of causal mask is applied to obtain the masked attention matrix A^{SM}:

$$A^{SM} = (A \odot M^C + P) \odot M^C \tag{3}$$

Effectiveness. From Eq. (3), the final output of the masked attention matrix A^{SM} is still valid in causal decoding, as it is element-wise multiplied by the causal mask again.

According to Theorem 4.1 in [16], pseudo attention also allows for unique positional encoding for identical sequences, which refines the attention distribution of the original model.

2.2 Random Sampling

Algorithm 2: RandS, Random Sampling Process

```
**input**: Intermediate Matrix $M^{AP}$, Kernel-LSH Mask Matrix $M^H$, Pseudo Attention Matrix $P$, sampling size S, threshold s
Initialize: generate upper triangular matrix $P$ with $p_{ij,j\geq i} = -\gamma(j - i)$
Apply Kernel-LSH to $M^{AP}$: $M_{processed} = M^H \odot M^{AP}$
```

```
for row $q_i$ in $M_{porcessed}$ do
    Sampling $S$ columns of light entries when $j \leq i$
    if $length(column) \geq s$ then
        Sample $S$ columns
    else
        Sample all columns
    end if
    Compute the sum $d_i$ of heavy entries, sampled light entries, and $p_{ij,j \geq i}$
end for
return $D_S = \boldsymbol{diag}(\{\boldsymbol{d}_i\}_{i=1}^{n})$
```

We apply the same sampling method with Algorithm 2 in [13]. According to its proof to Lemma 2, we can guarantee the effectiveness of this approximation D_S of the diagonal matrix D.

Together, we propose the entire algorithm.

Algorithm 3: SRS, Stable Random Sampling

```
input: Matrices $Q, K, V \in \mathbb{R}^{n \times d}$, Pseudo Attention Matrix $P$, StableMask $M^C$, Kernel-LSH Mask Matrix $M^H$, sampling size S, threshold s
Run Algorithm 1 and let $\{M^{AP}, A^{SM}\} = StableMask(Q, K, P, M^C)$
Run Algorithm 2 and let $D_S = RandS(M^{AP}, M^H, P, S, s)$
return $\boldsymbol{Att} = D_S^{-1} A^{SM} V$
```

2.3 Time Complexity Analysis

In this section, we focus on our substitution for recursion in the original hyperattention model, the Random Sampling process, which involves the following two main operations:

Sampling Operation. During the sampling process, each element in the set S is considered exactly once. The time complexity for processing each element is $O(1)$, which results in $O(n)$ time for the entire process.

Aggregation Operation. Aggregating the results involves a linear pass through the data, contributing $O(n)$ time.
In all, the total time complexity of the random sampling method is: $O(n)$

3 3. Experiment

3.1 Pre-training

In this session, we provide a comprehensive overview of the Ablation Experiments we have conducted for SRS. As shown in the Fig.2 and Fig.3, we begin by outlining the experimental methodology. Through the ablation experiments, we want to determine two points: first, if there is an improvement in the efficiency of DS over the recursive algorithm in HyperAttention, and second, if there is an improvement in the accuracy of the StableMask for the transformer's performance in long contexts.

Our Pre-training environment is Google Colab with an NVIDIA A100 GPU and 40 GB of memory, and the datasets are LongBench [22] from HuggingFace [23].

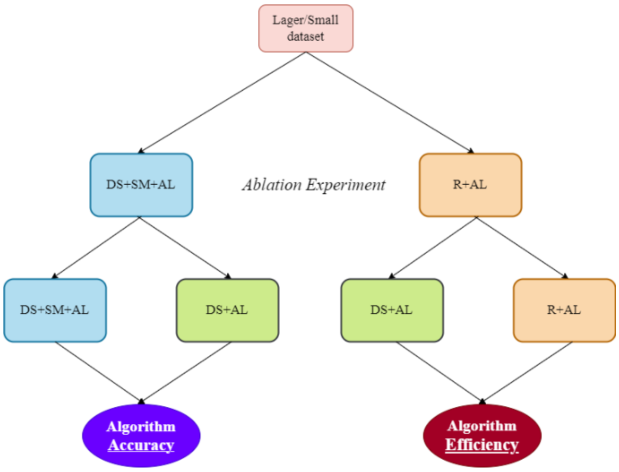


Fig. 2. Ablation experiment schema, where DS stands for random (direct) sampling, SM stands for StableMask, and AL stands for angular LSH.

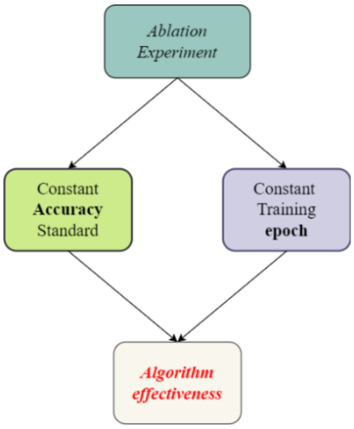


Fig. 3. (1) A constant accuracy requirement is applied to all models, and the training is stopped once the accuracy is reached or exceeded. (2) All algorithms have the same training epochs.

Constant Accuracy. According to the results shown in the experimental images, it can be concluded that compared to the original HyperAttention, which uses a recursive algorithm to deal with causal masking, both our proposed algorithms DS and DS+SM have a larger reduction in training time. The result indicates that direct sampling has a larger efficiency improvement for dealing with causal masking. Focusing on DS and DS+SM, it can be seen that StableMask has some reduction in the loss of the model during training. This indicates that SRS not only improves the efficiency of the transformer but also reduces its loss during training.

From Fig.4, we can find out that the loss of SRS decrease about 30% in large and small datasets. From Fig.5, we observed that the training time of SRS is about 33% lower than the original algorithm.

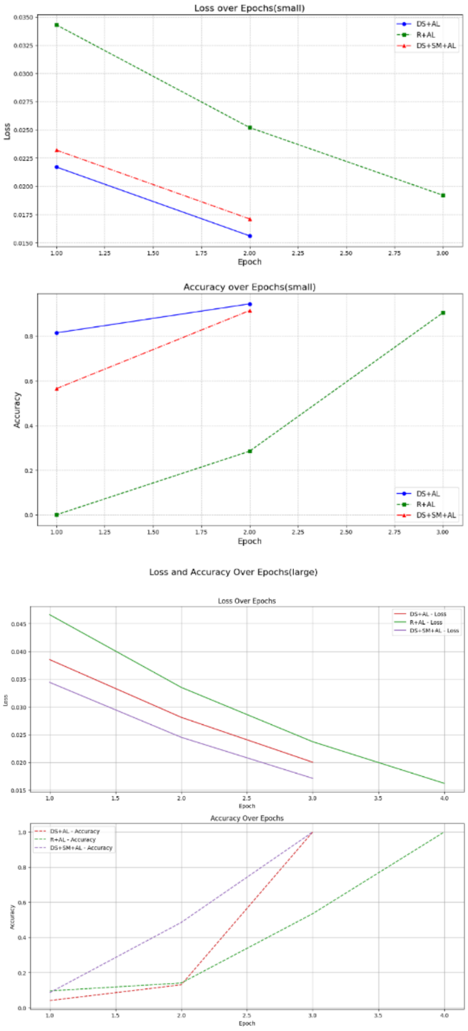


Fig. 4. Loss and accuracy results of the constant accuracy experiment in small and large datasets.

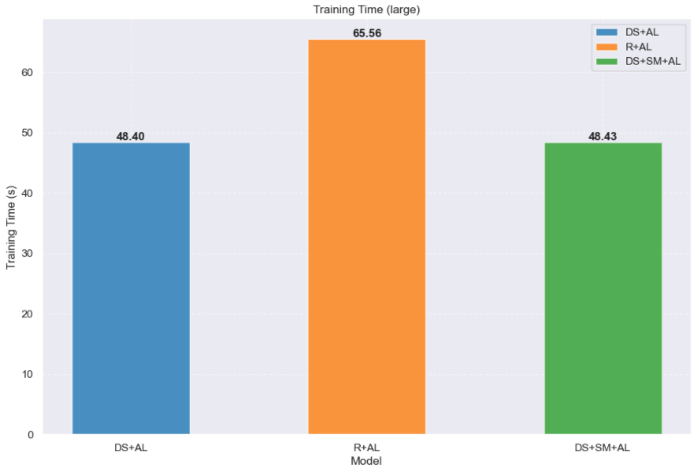


Fig. 5. Training time results of the constant accuracy experiment.

Constant Training Epoch. Since the number of training epochs is the same, there is not much difference in their training time, so we can better focus on improving the model performance.

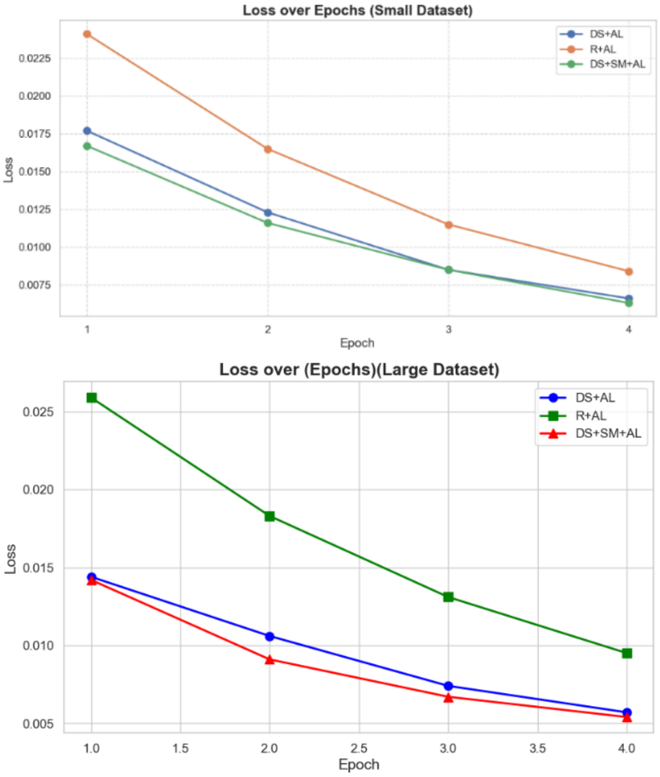


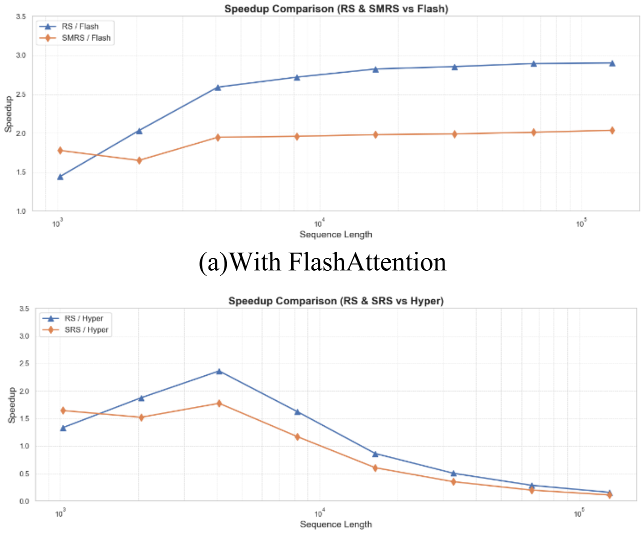
Fig. 6. Loss results of the constant training epoch experiment.

3.2 Empirical Experiments

We use the experimental setup of [13] as a foundation and conduct a series of experiments to validate the performance of SRS. Our experimental environment is Google Colab with an NVIDIA A100 GPU and 40 GB of memory, and the datasets are still LongBench[22]. We selected FlashAttention as the control group to evaluate the performance of SRS.

Single Self-Attention Layer Replacement. In this experiment, the performance of SRS across different sequence lengths is tested. We replace the original self-attention layer in the decoder-only Transformer with SRS and conduct experiments with sequence lengths ranging from 4,096 to 131,072. We then calculate and compare the wall-clock times for both forward and forward+backward operations when accelerated by SRS, and when computed by FlashAttention and HyperAttention. We only measure time with causal masking. All input matrices Q, K, V have the same length with dimension d=64, and the head size for all types of attention is 24. Additionally, SRS includes an extra gamma parameter used to generate the pseudo-attention score matrix. All other parameters for SRS and RS remain consistent with FlashAttention and HyperAttention.

From Fig.7(a), it can be observed that both SRS and RS exhibit are two to three times faster than FlashAttention, regardless of the sequence length. As the sequence length increases, the processing speeds of SRS and RS continue to rise relative to FlashAttention before eventually plateauing. We observe SRS run to up 2×faster than FlashAttention. Fig.7(b) shows that SRS is faster than HyperAttention when the sequence length is within 10,000 and exhibits an upward trend. Specifically, SRS run to up 1.5×faster than HyperAttention. However, as the sequence length exceeds 10,000, the advantage of HyperAttention gradually becomes more pronounced, which calls for closer scrutinization.



(a)With FlashAttention

(b)With HyperAttention

Fig. 7. (a) Comparison of precise computation times at different sequence lengths using SRS, RS, and FlashAttention. (b) Comparison of precise computation times at different sequence lengths using RS, SRS, and HyperAttention.

Monkey Patching Self-attention. To test our SRS performance in long context situations, we choose chatglm2-6b-32k [24], which is widely used in practical applications. In these LLMs, we use the SRS to replace the original final attention layers, which can vary from 0 to the number of all attention layers in each LLM. Then we experiment in LongBench dataset to evaluate the performance of monkey patched models in respect to perplexity and speed up.

The overall performance of perplexity and speed in **chatglm2-6b-32k** (ChatGLM2) with SRS is shown in Fig.8. With the number of replaced layers increasing, the perplexity and speed increase as well, making the inference time of ChatGLM2 30% faster on 32k context length while perplexity increases from 5.1 to 6.5. The result is under our theoretical expectation.

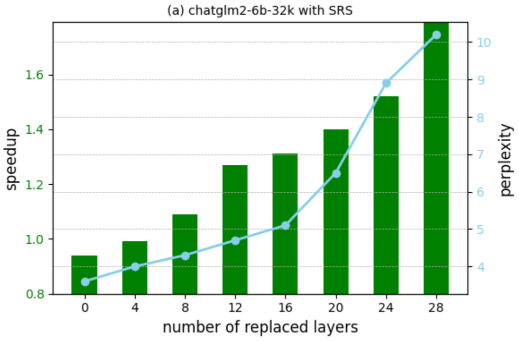
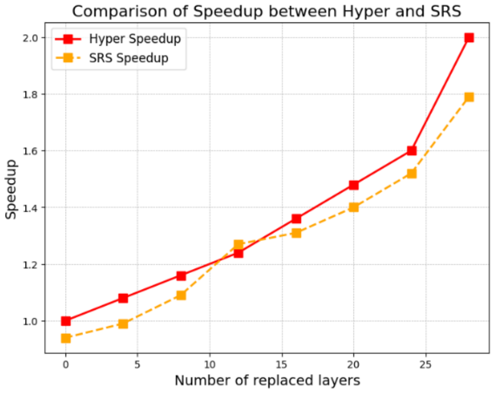


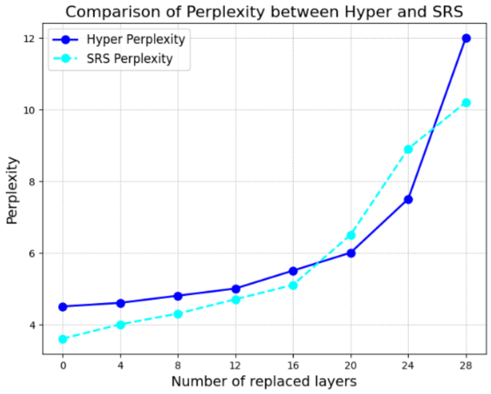
Fig. 8. The variations of perplexity and speed up of chatglm2-6b-32k monkey patched with SRS. The number of replaced layers vary from 0 to 28.

For a clearer comparison, we use two separate charts to highlight the differences between SRS and HyperAttention. In Fig.9(a), the speed of SRS is slightly slower than HyperAttention. However, Fig.8(b) shows that SRS generally has lower perplexity in different numbers of replaced layers. Specifically, the average SRS perplexity was 33% lower compared to HyperAttention.

From a comprehensive point of view, we hold the firm belief that SRS is fully effective in improving the LLMs' stability in long context settings even though it has a little loss of speed, which is acceptable.



(a) Speed up



(b) Perplexity

Fig. 9. (a)Comparison of speed up between SRS and HyperAttention in chatglm2-6k-32k (b) Comparison of perplexity between SRS and HyperAttention in chatglm2-6k-32k.

4 Conclusion

This paper presents the Stable Random Sampling Algorithm, which enhances the performance of large language models by tackling challenges in causal masking and attention distribution in decoder-only Transformers. Our method reduces the time complexity to $O(n)$, significantly improving efficiency over traditional recursive algorithms. By using a pseudo attention mask and Locality-Sensitive Hashing, our approach optimizes attention mechanisms, especially for handling long texts.

Overall, our algorithm performed well. We confirmed that SRS significantly outperforms FlashAttention in terms of speed performance, with SRS running at twice the speed of FlashAttention at any sequence length. The monkey patch experiment demonstrated that LLMs with SRS replaced exhibit lower perplexity compared to those with HyperAttention, indicating that SRS provides greater stability in improving LLM performance on long texts. Nonetheless, SRS still maintains significant potential in speed performance, especially when compared to HyperAttention for long sequences, which we need to address in future work. Conclusively, our work has presented a method for more effective and efficient causal masking optimization for decoder-only transformers in comparison to currently prevalent ones and will be further improved in the future.

Acknowledgement

Shuhao Zhang, Jiarui Li, and Jiayi Yu contributed equally to this work and should be considered co-first authors.

References

[1] Qingru Zhang, Dhananjay Ram, Cole Hawkins, Sheng Zha, Tuo Zhao Efficient long-range transformers*: You need to attend more, but not necessarily at every layer (EMNLP)2023*

[2] Lester, B., Al-Rfou, R., and Constant, N. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059, 2021.

[3] Yuhong Mo; Hao Qin; Yushan Dong; Ziyi Zhu; Zhenglin *Li Large Language Model (LLM) AI text generation detection based on transformer deep learning algorithm arXiv preprint arXiv:* 2405.06652v1

[4] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, François Fleuret *Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention arXiv preprint arXiv:*2012.12556

[5] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A Survey on Evaluation of Large Language Models. ACM Trans. Intell. Syst. Technol. 15, 3, Article 39 (June 2024)

[6] Naveed, H., Khan, A.U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Barnes, N., & Mian, A.S. (2023). A Comprehensive Overview of Large Language Models. *ArXiv, abs/2307.06435*.

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al*. Language models are few-shot learners. Neural Information Processing Systems (NeurIPS), 2020*

[8] Anthony Gillioz; Jacky Casas; Elena Mugellini*; Omar Abou Khaled Overview of the Transformer-based Models for NLP Tasks IEEE 2020*

[9] Narendra Patwardhan; Narendra Patwardhan; Carlo Sansone *Transformers in the Real World: A Survey on NLP Applications* MDPL 2023

[10] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, Zhaohui Yang, Yiman Zhang, Dacheng *Tao A Survey on Visual Transformer arXiv preprint arXiv:*2012.12556

[11] Ziyang Luo, Yadong Xi, Jing Ma, Zhiwei Yang, Xiaoxi Mao, Changjie Fan, and Rongsheng Zhang. 2022. *DecBERT: Enhancing the Language Understanding of BERT with Causal Attention Masks. In Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1185–1197, Seattle, United States. Association for Computational Linguistics.

[12] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. *Reformer: The Efficient Transformer*. In International Conference on Learning Representations (ICLR), 2020.

[13] Han, Insu, Rajesh Jayaram, Amin Karbasi, Vahab Mirrokni, David P. Woodruff, and Amir Zandieh. *"Hyperattention: Long-context attention in near-linear time." arXiv preprint arXiv:2310.05869* (2023).

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin *Attention is All you Need (NIPS)2017*

[15] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, Adrian Weller *Rethinking Attention with Performers arXiv preprint arXiv:*2009.14794v4

[16] Yin, Qingyu, Xuzheng He, Xiang Zhuang, Yu Zhao, Jianhua Yao, Xiaoyu Shen, and Qiang Zhang. *"StableMask: Refining Causal Masking in Decoder-only Transformer." arXiv preprint arXiv:2402.04779* (2024).

[17] Amir Zandieh, Insu Han, Majid Daliri, and Amin Karbasi. *Kdeformer: Accelerating transformers via kernel density estimation*. In International Conference on Machine Learning (ICML),2023.

[18] Pang, T., Xu, K., Dong, Y., Du, C., Chen, N., and Zhu, J. Rethinking softmax cross-entropy loss for adversarial robustness. In International Conference on Learning Representations, 2019

[19] Xiao, G., Tian, Y., Chen, B., Han, S., and Lewis, M. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453, 2023*

[20] Luo, S., Li, S., Zheng, S., Liu, T.-Y., Wang, L., and He, D. *Your transformer may not be as powerful as you expect.* Advances in Neural Information Processing Systems, 35: 4301–4315, 2022

[21] Philipp Dufter, Martin Schmitt, Hinrich Schütze; Position Information in Transformers: An Overview. *Computational Linguistics* 2022

[22] 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. *LongBench: A Bilingual, Multitask Benchmark for Long Context Understanding*, 2023

[23] Wolf, Thomas et al. “HuggingFace's Transformers: State-of-the-art Natural Language Processing.” *ArXiv* abs/1910.03771 (2019): n. pag.

[24] Du, Zhengxiao, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang and Jie Tang. “GLM: General Language Model Pretraining with Autoregressive Blank Infilling.” *Annual Meeting of the Association for Computational Linguistics* (2021).