Stable Language Model Pre-training by Reducing Embedding Variability

Woojin Chung **Jiwoo Hong** **Na Min An** **James Thorne** * **Se-Young Yun** *

KAIST AI
{gartland, jiwoo_hong, naminan, thorne, yunseyoung}@kaist.ac.kr

Abstract

Stable pre-training is essential for achieving better-performing language models. However, tracking pre-training stability by calculating gradient variance at every step is impractical due to the significant computational costs. We explore Token Embedding Variability (TEV) as a simple and efficient proxy for assessing pre-training stability in language models with pre-layer normalization, given that shallower layers are more prone to gradient explosion (section 2.2). Moreover, we propose Multi-head Low-Rank Attention (MLRA) as an architecture to alleviate such instability by limiting the exponential growth of output embedding variance, thereby preventing the gradient explosion (section 3.2). Empirical results on GPT-2 with MLRA demonstrate increased stability and lower perplexity, particularly in deeper models.

1 Introduction

Improving large language models (LLMs) typically involves increasing model size, especially through greater depth (Brown et al., 2020; Kaplan et al., 2020; Rae et al., 2022; Xue et al., 2023). However, this approach often causes instability during pre-training, indicated by sudden spikes in loss (Chowdhery et al., 2022; Zhai et al., 2023), while stable pre-training typically leads to stronger performance under controlled training configurations (Touvron et al., 2023a; Takase et al., 2024). Such instability can lead to catastrophic divergence or degradation, underscoring the importance of assessing pre-training stability (Chowdhery et al., 2022; Zhai et al., 2023; Takase et al., 2024).

The conventional methods for monitoring the pre-training stability are computationally expensive (Kaplan et al., 2020), such as observing the gradient variance which needs additional $O(nd)$ for gradient matrix $g_t \in \mathbb{R}^{n \times d}$ (Zhao et al., 2024) or analyzing the singular values of the second-order derivative of the loss with respect to model parameters (Yao

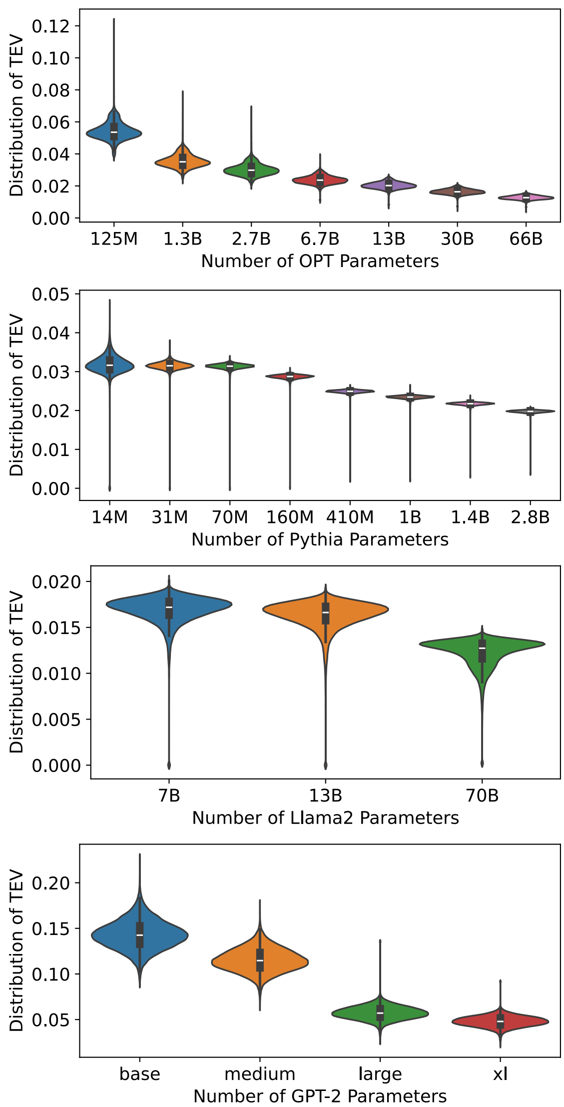

Figure 1: TEV distribution for OPT, Pythia, Llama-2, and GPT-2 reveals that as model size grows, both μ_{TEV} and σ_{TEV} decrease. This trend correlates with better model performance, as reduced noisy gradients lead to higher pre-training stability and improved performance. For a fair comparison, Pythia 6.9B and 12B were excluded due to their different vocabulary sizes.

*Corresponding authors

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 10852–10863
November 12-16, 2024

et al., 2020; Gilmer et al., 2021; Cohen et al., 2024). Further details will be addressed in Appendix A.

We address both issues by dissecting the token embedding layer: we theoretically and empirically substantiate that the standard deviation of token embedding in the embedding layer, denoted token embedding variability (TEV), can be a simple and efficient proxy for estimating pre-training stability in models with pre-layer normalization (Radford et al., 2019; Zhang et al., 2022; Touvron et al., 2023b) as it best reflects the level of gradient noise (*i.e.,* gradient variance). We demonstrate a correlation between TEV and language model performance by evaluating OPT (Zhang et al., 2022), Pythia (Biderman et al., 2023), Llama-2 (Touvron et al., 2023b), and GPT-2 (Radford et al., 2019) (Figure 1). Furthermore, we introduce factorized multi-head attention projection matrices (*i.e.,*, Multi-head Low-Rank attention; MLRA) as a fundamental method to mitigate pre-training instability. we empirically show that pre-training GPT-2 (Radford et al., 2018) with MLRA effectively *lowers* TEV and achieves higher downstream performance with better pre-training stability, aligning with the theoretical analysis of TEV.

2 Pre-training Stability Proxy

2.1 Preliminaries

The token embedding layer $\mathbf{E} \in \mathbb{R}^{|V| \times d_{\text{model}}}$ of the transformer (Vaswani et al., 2017) maps an input sequence $\mathbf{x} = [x_1, x_2, \cdots, x_n]$ with n tokens into the vector-wise representations $X_0 \in \mathbb{R}^{n \times d_{\text{model}}}$,

$$\mathbf{E} = \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \cdots & \mathbf{e}_{|V|} \end{bmatrix}^{\mathrm{T}},$$

where $|V|$ and d_{model} refer to the size of vocabulary and the hidden dimension, and $e_i \in \mathbb{R}^{d_{\text{model}}}$ denotes the embedding weight vector corresponding to each token. Thus, $\mathbf{e}_i$ can be written as:

$$\mathbf{e}_i = \begin{pmatrix} e_{i,1} & e_{i,2} & \cdots & e_{i,d_{\text{model}}} \end{pmatrix}.$$

The initial embedding vectors, $X_0 \in \mathbb{R}^{n \times d_{\text{model}}}$, pass through $2N$ different sub-layers $\mathcal{F}$,

$$X_t = \mathcal{F}_t(X_{t-1}) + X_{t-1}$$

where $t \in \{1, 2, \ldots, N\}$ denotes the layer index, X_t denotes the hidden representation returned from t-th layer. Finally, the logit $L \in \mathbb{R}^{n \times |V|}$ for predicting the next token is calculated by mapping X_N into $|V|$-dimensional space with the language model head. Language models such as BERT (Devlin et al., 2019) GPT-2 (Radford et al., 2019) Mistral (Jiang et al., 2023) and Llama-2 (Touvron et al., 2023a) typically tie language modeling head with the embedding matrix $\mathbf{E}$ to reduce the number of trainable parameters and induce input and output embedding behaves similarly to similar words (Mnih and Teh, 2012; Press and Wolf, 2017; Inan et al., 2017).

$$L = X_N \cdot \mathbf{E}^T.$$

2.2 Stability and Token Embedding Layer

We show that the token embedding layer $\mathbf{E}$ plays a crucial role in understanding the pre-training stability in two perspectives: 1) gradient explosion and 2) skewness in token frequency.

Gradient explosion Recently proposed LLMs typically apply pre-layer norm (Xiong et al., 2020, pre-LN) to mitigate pre-training instability in the early stage of pre-training due to high gradient variance (*i.e.,* noisy gradient) (Liu et al., 2021) [1]. Contrary to post-layer norm (Ba et al., 2016; Vaswani et al., 2017, post-LN), the gradient norms are usually larger in shallower layers compared to deeper layers (Xie et al., 2023), leading the gradient of token embedding layer ∇X_0 to have the greatest magnitude:

$$\nabla X_0 = \nabla X_N \cdot \prod_{t=1}^{N-1} \left(\frac{\partial \mathcal{F}_{t-1}(X_{t-1})}{\partial X_{t-1}} + \mathbf{I} \right),$$

as the gradient exponentially grows over the layers due to the residual connection (He et al., 2016). Such property, which causes spikes in pre-training loss, is amplified in the token embedding layer (*i.e.,* gradient explosion). Thus, the token embedding layer $\mathbf{E}$ effectively reflects the training instability. We empirically confirm this in Section 4.2. For simplicity, we assume a negligible or zero correlation between the gradient and weight matrix ($\mathrm{Cov}(X_0, \nabla X_0) \approx 0$), and our experiment supports that this assumption is valid in real scenarios.

$$\begin{aligned} \mathrm{Var}(X_0 - \nabla X_0) = \mathrm{Var}(X_0) + \mathrm{Var}(\nabla X_0) \\ - 2\mathrm{Cov}(X_0, \nabla X_0) \end{aligned}$$

Skewness in token frequency Since the true distribution of natural language is inherently non-uniform (Zipf, 1935), mini-batch gradient descent

[1] Gradient mean close to 0 in the early stage of pre-training as weights are initialized from normal distributions with mean 0 (Balduzzi et al., 2018). Exponential moving average amplifies variance of gradient estimation (Liu et al., 2021)

leads to imbalanced updates of token embeddings. The gradient of the mini-batch is normalized by its total number of tokens (Laurent et al., 2024; Dettmers et al., 2022). The tokens in each mini-batch B can be written as:

$$B = \{\mathbf{x}_i\}_{i=1}^{M} = \left\{[x_{i,j}]_{j=1}^{C} \mid i = 1, 2, \ldots, M\right\},$$

where M is the batch size and C is the sequence length of each token. This could be understood as sampling a total of $M \times C$ independent random samples from the population V *with replacement*. Therefore, the skewed token distribution and mini-batch updates lead to the selective update of certain token's embedding weights of $\mathbf{E}$.

2.3 Token Embedding Variability (TEV)

When pre-training is stable, the norm of each token's embedding weight vector $||e_i||$ should be close to uniform. $||e_i||$ can be written as:

$$||e_i|| = \sqrt{d_{\text{model}} \cdot \left(\mu_i^2 + \sigma_i^2\right)}$$

where μ_i^2 and σ_i^2 are element-wise mean and variance of e_i. d_{model} is a fixed value with a positive integer, and μ_i stays close to zero throughout pre-training[2]. We confirmed that μ_i is close to zero in multiple pre-trained LLMs in Appendix B. Hence, σ_i^2 is the dominant term determining $||e_i||$.

However, the standard deviation is typically significantly less than one, and the token embedding norm falls short as a reliable proxy of pre-training stability. Given that the model dimension (d_{model}) is a positive integer and generally much larger than the standard deviation (σ), the token embedding norm largely overlooks the standard deviation. This oversight is critical, as the standard deviation is key to capturing gradient variance during pre-training, which the norm fails to account for accurately.

Therefore, we propose the distribution of token-level standard deviation (σ) as the pre-training stability proxy: *i.e.,* token embedding variability (TEV) distribution. TEV of i-th token (x_i) is defined as:

$$\text{TEV}_i = \sqrt{\frac{1}{d}\sum_{j=1}^{d}\left(e_{ij} - \bar{e}_i\right)^2},$$

where $\bar{e}_i$ is the element mean of the ith token's weight vector. Eventually, the mean μ_{TEV} and standard deviation σ_{TEV} of TEV over the entire vocabulary is:

$$\mu_{\text{TEV}} = \frac{1}{|V|}\sum_{i=1}^{|V|}\text{TEV}_i$$

$$\sigma_{\text{TEV}} = \sqrt{\frac{1}{|V|}\sum_{i=1}^{|V|}(\text{TEV}_i - \mu_{\text{TEV}})^2}$$

Our experiments in Section 4.2 verify that stable pre-training with less suffer from noisy gradient results in a TEV distribution with a lower μ_{TEV} and σ_{TEV}.

3 Mitigating TEV with Factorization

We propose low-rank factorized attention projection matrices (*i.e.,* Multi-head Low-Rank attention; MLRA) as a simple way of lowering TEV mean and variances, improving pre-training stability and performance.

3.1 Multi-head Low Rank Attention (MLRA)

Query, key, and value projection matrices W_q, W_k and W_v can be factorized as: $WX_t = W^U W^D X_t$, where $W \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$, $W^U \in \mathbb{R}^{d_{\text{model}} \times r}$ and $W^D \in \mathbb{R}^{r \times d_{\text{model}}}$ ($r < d_{\text{model}}$), and r refers to the rank. MLRA introduces minimal overhead since MLRA is only applied to the weights within the multi-head attention mechanism. As W can be reconstructed from the two low-rank matrices, there is no additional cost at inference.

3.2 Theoretical Analysis

The factorization property of MLRA mitigates the exponential growth of variance in the output representations across layers. The variance with MLRA with the hidden representation in tth layer can be simply written as:

$$\sigma^2(W^U W^D X_t) = r \cdot d_{\text{model}} \cdot \sigma^2(W^U) \cdot \sigma^2(W^D),$$

assuming X_t, W^U and W^D are independent each other, and X_t has zero mean. This is due to the independent initialization of weights from the identical distribution and the application of layer normalization to the input, guaranteeing a zero mean. Similarly, we can assume $\sigma^2(X_t) = 1$.

For $\sigma^2(W^U)$ and $\sigma^2(W^D)$, we use the actual values from `torch.nn.Linear`, where the weights w are initialized using Kaiming uniform initialization (He et al., 2015), $w \sim \mathcal{U}\left(-\sqrt{\frac{1}{n}}, \sqrt{\frac{1}{n}}\right)$ with

[2]Token embedding layer $\mathbf{E}$ is initialized using a normal distribution with a mean of zero.

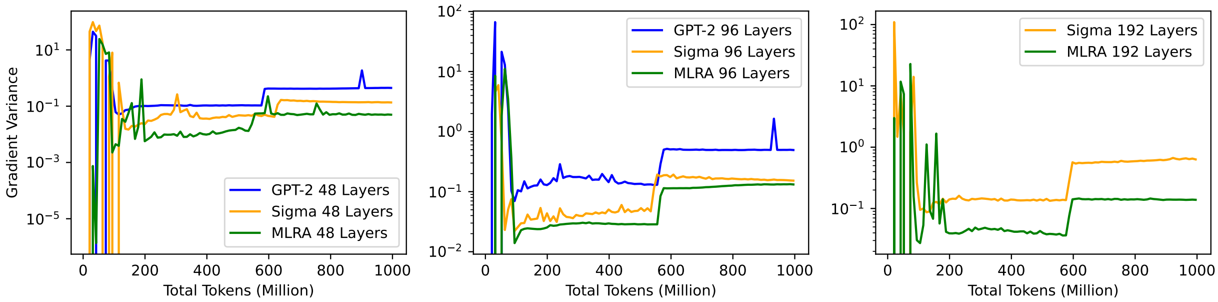

Figure 2: Gradient variance (↓) comparison across tested models with different layers. MLRA shows the lowest gradient variance than GPT-2 and σReparam. GPT-2 with 192 layers was excluded as the training failed 5 times (*i.e.*, The gradient variance is infinite at the earlier steps and becomes infinitesimal in the later steps).

$\sigma^2(W) = \frac{1}{3d_{\text{model}}}$, $W \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ (*See* Appendices C for details). First, the initial variances of a square matrix attention weight and MLRA are [3]:

$$\sigma^2(WX_t) = d_{\text{model}} \cdot \frac{1}{3d_{\text{model}}} = \frac{1}{3}$$
$$\sigma^2(W^U W^D X_t) = d_{\text{r}} \cdot d_{\text{model}} \cdot \frac{1}{3d_{\text{model}}} \cdot \frac{1}{3d_{\text{r}}} = \frac{1}{9}.$$

Therefore, passing through two linear layers further diminishes the variance of the token embeddings. We further the extended calculation of variance growth of each attention head and self-attention in Appendix D.

MLRA addresses gradient explosion in a similar manner to scaled initialization by reducing the magnitude of weights in both the feed-forward network and self-attention module during the weight initialization (Shoeybi et al., 2020; Scao et al., 2022; Biderman et al., 2023; Takase et al., 2024). Unlike scaled initialization, MLRA uses standard initialization, leading to larger gradient updates.

On the other hand, simply applying low-rank reparameterization to all weights from the beginning of pre-training degrades performance due to the high intrinsic rank of weight matrices (Aghajanyan et al., 2020; Lialin et al., 2023; Zhao et al., 2023, 2024). To address this, we concentrate on the multi-head architecture, which divides output representation across hidden dimensions, to mitigate low-rank bottlenecks. One example is as follows: Let matrix $\mathbf{A}$ be a 3×6 matrix with

$$\mathbf{A} = [\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_1 + \mathbf{e}_2, \mathbf{e}_1 + \mathbf{e}_3, \mathbf{e}_2 + \mathbf{e}_3]$$

where $\mathbf{e}_1, \mathbf{e}_2$ and $\mathbf{e}_3$ are the standard basis vectors in $\mathbb{R}^3$. The submatrices $\mathbf{A}_1 = [\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3]$ and $\mathbf{A}_2 = [\mathbf{e}_1 + \mathbf{e}_2, \mathbf{e}_1 + \mathbf{e}_3, \mathbf{e}_2 + \mathbf{e}_3]$ both possess a rank of 3, illustrating that a rank-3 matrix $\mathbf{A}$ can still have full-rank submatrices, even when the matrix is divided along hidden dimensions. Thus we hypothesize that matrix factorization within a multi-head architecture could reduce gradient variance and avoid low-rank bottlenecks during pre-training.

4 Experiments

We demonstrate the significance of TEV in Section 4.1 and the effectiveness of MLRA on pre-training stability and performance in Section 4.2.

4.1 Experimental Design

Baseline We pre-train GPT-2 (Radford et al., 2019) from scratch with three different methods: 1) conventional architecture (GPT-2), 2) σReparam (Zhai et al., 2023), and 3) MLRA. All the pre-training configurations, including learning rate and number of parameters, are fixed over methods. Further details can be found in Appendix E.

Datasets We pre-train each model using WebText (Radford et al., 2019) and evaluate the downstream performances on Lambada (Paperno et al., 2016), Wikitext-2 (Merity et al., 2016), Wikitext-103 (Merity et al., 2022), Penn Tree Bank (PTB) (Marcus et al., 1993), and 1th Billion Word Benchmark (1BW) (Chelba et al., 2014) datasets.

4.2 Results

Pre-training stability In Figure 2, MLRA has the lowest gradient variance in all configurations when pre-trained on the first one billion tokens. As models deepen, the gradient variance gap between baselines and MLRA is increasingly pronounced. A significant spike in gradient variance around 600M tokens across all configurations suggests high optimization difficulty (Faghri et al.,

[3] For $X_t \sim \mathcal{N}(0,1)$, and $W^U, W^D \sim \mathcal{N}(0, \sigma^2)$, $\sigma^2(WX_t) = d_{\text{model}} \cdot \sigma^2$ and $\sigma^2(W^U W^D X_t) = r \cdot d_{\text{model}} \cdot \sigma^4$. Weights are initialized with $\sigma = 0.02$ in huggingface library

Models	Layers	$\mu_{\text{TEV}}\downarrow$	$\sigma_{\text{TEV}}\downarrow$	LAMBADA ↓	WIKI2 ↓	WIKI103 ↓	PTB ↓	1BW ↓
GPT-2	48	0.0892	0.0125	79.60	44.74	54.53	53.05	59.28
σReparam		0.0879	0.0115	76.02	45.06	54.73	50.91	57.67
MLRA		**0.0875**	**0.0114**	**70.61**	**42.86**	**50.92**	**50.27**	**55.46**
GPT-2	96	0.0872	0.0120	71.52	42.84	51.61	49.80	56.92
σReparam		0.0849	0.0113	70.39	42.34	50.23	49.53	55.72
MLRA		**0.0843**	**0.0110**	**62.31**	**39.44**	**46.22**	**44.17**	**51.56**
GPT-2	192	0.0875	0.0117	64.62	41.31	47.75	47.73	51.97
σReparam		0.0870	0.0112	59.86	39.06	44.13	43.79	48.51
MLRA		**0.0864**	**0.0104**	**53.69**	**35.39**	**44.17**	**41.14**	**45.03**

Table 1: Zero-shot perplexity and token embedding variability (TEV) comparison between GPT-2, σReparam, and MLRA with varying number of layers. The bolded texts indicate the lowest μ_{TEV}, σ_{TEV} and perplexity across the model configurations with the same number of layers. The model dimension d_{model} for GPT-2 and σReparam is set to 384, while the intermediate dimension d_{rank} of MLRA is configured to 192. MLRA demonstrates both the lowest μ_{TEV} and σ_{TEV} and perplexity, implying MLRA leads to the best pre-training stability and performance.

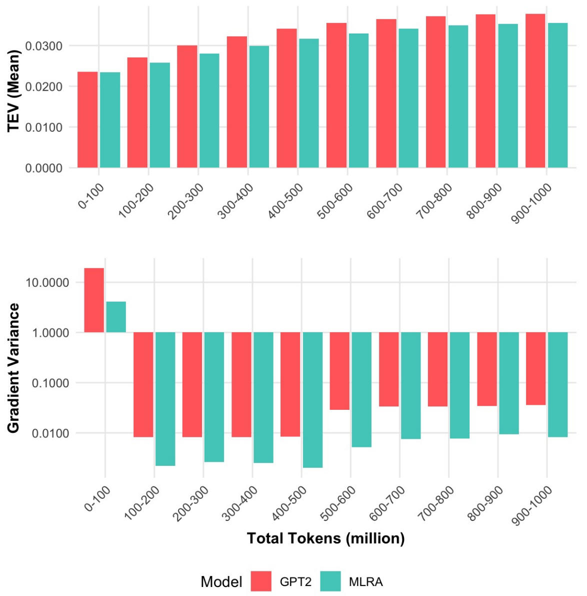

Figure 3: μ_{TEV} (top) and gradient variance (bottom) during the pre-training of both GPT-2 and MLRA, each with 48 layers, over the course of 1 billion tokens. For both settings, μ_{TEV} and gradient variance imply identical trends over the pre-training procedure.

2020). We excluded the result of GPT-2 with 192 layers as it failed five times during pre-training, showing the pre-training instability of GPT-2 as it gets deeper Deeper model pre-training is unstable (Wang et al., 2022a,b) due to shattered gradients resembling white noise (Balduzzi et al., 2018).

Token Embedding Variability Aligned with the results in Figure 2, MLRA *consistently* exhibits lower μ_{TEV} and σ_{TEV} compared to GPT-2 and σReparam in Table 1. Moreover, μ_{TEV} and σ_{TEV} for 192 layers are higher than those for 96 layers, indicating increased gradient variance at this deeper layer, as shown in Figure 2. We further study the correlation between gradient variance and μ_{TEV} over 1 billion tokens. Figure 3 shows that higher gradient variance corresponds to a higher μ_{TEV}, with the rate of increase in μ_{TEV} depending on the magnitude of gradient variance.

Perplexity performance As can be observed in Table 1, the zero-shot performances of MLRA are significantly improved compared to baselines across different numbers of layers and datasets on which these models are not fine-tuned. We also achieve better zero-shot perplexity results than σReparam (Zhai et al., 2023), the current state-of-the-art model that alleviates the attention entropy collapse problem. Furthermore, the perplexity gap of MLRA becomes much larger than the vanilla counterparts as the number of layers increases. These findings empirically prove the efficacy and depth-scalability of the proposed method.

5 Conclusion

This paper shows that Token Embedding Variability (TEV) can be used as a simple and efficient proxy for pre-training stability, avoiding the high cost of monitoring gradient variance. Theoretical analysis reveals that factorized multi-head attention projection matrices (*i.e.*, MLRA) reduce gradient explosion. Empirically, MLRA lowers TEV mean and variance, improves stability, and outperforms GPT-2 and σReparam in reducing zero-shot perplexity, particularly in deeper models.

Limitations

While we conducted a controlled study of the pre-training stability and token embedding variability (TEV) as a proxy by pre-training GPT-2 from scratch, the scale of the base model was limited to a maximum of 1.5B parameters. We also compare the performance and stability with a single pre-training corpus, the WebText. Therefore, the scalability of MLRA and TEV as a pre-training stability proxy will be further studied across a larger range of scales, 7B, for instance.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government(MSIT) (No. RS-2024-00457882, AI Research Hub Project) and (No.RS-2019-II190075 Artificial Intelligence Graduate School Program(KAIST)).

References

Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. 2020. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *Preprint*, arXiv:2012.13255.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *Preprint*, arXiv:1607.06450.

David Balduzzi, Marcus Frean, Lennox Leary, JP Lewis, Kurt Wan-Duo Ma, and Brian McWilliams. 2018. The shattered gradients problem: If resnets are the answer, then what is the question? *Preprint*, arXiv:1702.08591.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. *Preprint*, arXiv:2304.01373.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Gino Brunner, Yang Liu, Damian Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. 2019. On identifiability in transformers. In *International Conference on Learning Representations*.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2014. One billion word benchmark for measuring progress in statistical language modeling. *Preprint*, arXiv:1312.3005.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways. *Preprint*, arXiv:2204.02311.

Jeremy M. Cohen, Behrooz Ghorbani, Shankar Krishnan, Naman Agarwal, Sourabh Medapati, Michal Badura, Daniel Suo, David Cardoze, Zachary Nado, George E. Dahl, and Justin Gilmer. 2024. Adaptive gradient methods at the edge of stability. *Preprint*, arXiv:2207.14484.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 2022. 8-bit optimizers via block-wise quantization. *Preprint*, arXiv:2110.02861.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Fartash Faghri, David Duvenaud, David J. Fleet, and Jimmy Ba. 2020. A study of gradient variance in deep learning. *Preprint*, arXiv:2007.04532.

Justin Gilmer, Behrooz Ghorbani, Ankush Garg, Sneha Kudugunta, Behnam Neyshabur, David Cardoze, George Dahl, Zachary Nado, and Orhan Firat. 2021. A loss curvature perspective on training instability in deep learning. *Preprint*, arXiv:2110.04369.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *Preprint*, arXiv:1502.01852.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Yerlan Idelbayev and Miguel Á. Carreira-Perpiñán. 2020. Low-rank compression of neural nets: Learning the rank of each layer. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8046–8056.

Hakan Inan, Khashayar Khosravi, and Richard Socher. 2017. Tying word vectors and word classifiers: A loss framework for language modeling. *Preprint*, arXiv:1611.01462.

Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. 2014. Speeding up convolutional neural networks with low rank expansions. *Preprint*, arXiv:1405.3866.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Thomas Laurent, James von Brecht, and Xavier Bresson. 2024. Feature collapse. In *The Twelfth International Conference on Learning Representations*.

Vladislav Lialin, Namrata Shivagunde, Sherin Muckatira, and Anna Rumshisky. 2023. Relora: High-rank training through low-rank updates. *Preprint*, arXiv:2307.05695.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2021. On the variance of the adaptive learning rate and beyond. *Preprint*, arXiv:1908.03265.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of english: the penn treebank. *Comput. Linguist.*, 19(2):313–330.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *Preprint*, arXiv:1609.07843.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2022. Pointer sentinel mixture models. In *International Conference on Learning Representations*.

Andriy Mnih and Yee Whye Teh. 2012. A fast and simple algorithm for training neural probabilistic language models. *Preprint*, arXiv:1206.6426.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2023. Scaling data-constrained language models. *Preprint*, arXiv:2305.16264.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The lambada dataset: Word prediction requiring a broad discourse context. *Preprint*, arXiv:1606.06031.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. *Preprint*, arXiv:1608.05859.

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

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. 2022. Scaling language models: Methods, analysis & insights from training gopher. *Preprint*, arXiv:2112.11446.

Teven Le Scao, Thomas Wang, Daniel Hesslow, Lucile Saulnier, Stas Bekman, M Saiful Bari, Stella Biderman, Hady Elsahar, Niklas Muennighoff, Jason Phang, Ofir Press, Colin Raffel, Victor Sanh, Sheng Shen, Lintang Sutawika, Jaesung Tae, Zheng Xin Yong, Julien Launay, and Iz Beltagy. 2022. What language model to train if you have one million gpu hours? *Preprint*, arXiv:2210.15424.

Steffen Schotthöfer, Emanuele Zangrando, Jonas Kusch, Gianluca Ceruti, and Francesco Tudisco. 2022. Low-rank lottery tickets: finding efficient low-rank neural networks via matrix differential equations. In *Advances in Neural Information Processing Systems*, volume 35, pages 20051–20063. Curran Associates, Inc.

Sam Shleifer, Jason Weston, and Myle Ott. 2021. Normformer: Improved transformer pretraining with extra normalization. *Preprint*, arXiv:2110.09456.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2020. Megatron-lm: Training multi-billion parameter language models using model parallelism. *Preprint*, arXiv:1909.08053.

Y Sui, M Yin, W Yang, Y Gong, J Xiao, H Phan, D Ding, X Xu, S Liu, Z Chen, et al. 2023. Elrt: Towards efficient low-rank training for compact neural networks, 2023. In *URL https://openreview. net/forum*.

Sho Takase, Shun Kiyono, Sosuke Kobayashi, and Jun Suzuki. 2024. Spike no more: Stabilizing the pre-training of large language models. *Preprint*, arXiv:2312.16903.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023a. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv: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. *Preprint*, arXiv:1706.03762.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, and Furu Wei. 2022a. Deepnet: Scaling transformers to 1,000 layers. *Preprint*, arXiv:2203.00555.

Hongyu Wang, Shuming Ma, Shaohan Huang, Li Dong, Wenhui Wang, Zhiliang Peng, Yu Wu, Payal Bajaj, Saksham Singhal, Alon Benhaim, Barun Patra, Zhun Liu, Vishrav Chaudhary, Xia Song, and Furu Wei. 2022b. Foundation transformers. *Preprint*, arXiv:2210.06423.

Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. 2022. Mitigating neural network overconfidence with logit normalization. *Preprint*, arXiv:2205.09310.

Genta Indra Winata, Samuel Cahyawijaya, Zhaojiang Lin, Zihan Liu, and Pascale Fung. 2020. Lightweight and efficient end-to-end speech recognition using low-rank transformer. *Preprint*, arXiv:1910.13923.

Shufang Xie, Huishuai Zhang, Junliang Guo, Xu Tan, Jiang Bian, Hany Hassan Awadalla, Arul Menezes, Tao Qin, and Rui Yan. 2023. Residual: Transformer with dual residual connections. *Preprint*, arXiv:2304.14802.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. 2020. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pages 10524–10533. PMLR.

Fuzhao Xue, Jianghai Chen, Aixin Sun, Xiaozhe Ren, Zangwei Zheng, Xiaoxin He, Yongming Chen, Xin Jiang, and Yang You. 2023. A study on transformer configuration and training objective. *Preprint*, arXiv:2205.10505.

Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael Mahoney. 2020. Pyhessian: Neural networks through the lens of the hessian. *Preprint*, arXiv:1912.07145.

Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatao Gu, and Joshua M Susskind. 2023. Stabilizing transformer training by preventing attention entropy collapse. In *International Conference on Machine Learning*, pages 40770–40803. PMLR.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models. *Preprint*, arXiv:2205.01068.

Jiawei Zhao, Yifei Zhang, Beidi Chen, Florian Schäfer, and Anima Anandkumar. 2023. Inrank: Incremental low-rank learning. *Preprint*, arXiv:2306.11250.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. 2024. Galore: Memory-efficient llm training by gradient low-rank projection. *Preprint*, arXiv:2403.03507.

George K. Zipf. 1935. *The Psycho-Biology of Language*. Houghton Mifflin, Boston, MA.

A Related Works

Training instability in LLMs Modern LLMs, such as the GPT series (Radford et al., 2018, 2019; Brown et al., 2020) and llama series (Touvron et al., 2023a,b) frequently use pre-layer normalization (pre-LN), which normalizes inputs instead of outputs (Zhai et al., 2023). Pre-LN increases the standard deviation of hidden representation in upper layers, preserving unique data features and preventing token embeddings from becoming too similar (Brunner et al., 2019). However, it can cause gradient explosion in shallow layers, where gradients from shallower layers grow disproportionately larger than those from deeper layers, affecting training stability (Shleifer et al., 2021; Takase et al., 2024). Takase et al., 2024 shows that in pre-LN settings, sub-component norms grow exponentially when standard deviations exceed 1, which is a common issue with typical initialization. To address this, methods like sub-LayerNorm (Shleifer et al., 2021; Wang et al., 2022b) and sigma reparameterization (σReparam) (Zhai et al., 2023), which scales weights by their spectral norms, have been developed to enhance stability. Scaled initialization which scales down the initial weight values (Shoeybi et al., 2020; Scao et al., 2022) also helps mitigate gradient spikes during pre-training.

Low-rank pre-training A plethora of literature regarding low-rank training has been conducted in the domains of convolution neural network (CNN) compression, regularization, and the pursuit of efficient training and inference (Idelbayev and Carreira-Perpiñán, 2020; Jaderberg et al., 2014; Sui et al., 2023; Schotthöfer et al., 2022; Winata et al., 2020). Nevertheless, most of these methods are tailored exclusively for CNNs and have yet to undergo assessment on large-scale Transformers (Vaswani et al., 2017), which could significantly benefit from efficient training due to the large scale of language models.

Recently, methods like ReLoRA (Lialin et al., 2023) and InRank (Zhao et al., 2023) have adopted an approach that starts training with full-rank matrices and then transitions to low-rank training. These studies suggest that the intrinsic rank of Transformers decreases as training progresses (Aghajanyan et al., 2020; Hu et al., 2021). In earlier phases, full-rank matrices are used to stabilize training before switching to low-rank matrices after a few initial steps.

B Mean of Token Embedding in pre-trained LLM

Figure 4 illustrates that the row-wise average of the absolute mean value of $|V|$ token embeddings in the token embedding layer $\mathbf{E} \in \mathbb{R}^{|V| \times d_{\text{model}}}$ across OPT (Zhang et al., 2022), Pythia (Biderman et al., 2023), Llama-2 (Touvron et al., 2023b) and GPT-2 (Radford et al., 2019) remains centered around zero after pre-training. One possible conjecture on this phenomenon is that pre-LN (Xiong et al., 2020) introduces layer normalization before the logits, resulting in a similar effect as logit normalization (Wei et al., 2022). This process enforces a constant vector norm on the logits during training, helping to alleviate the issue of overconfidence (*i.e.* unusually high softmax confidences, even when the inputs are significantly different from the training data). Additionally, a slight negative correlation between model size and the embedding mean is observed, warranting further investigation.

C Variance of Kaiming Uniform Initialization

The Kaiming uniform initialization is defined by the following distribution:

$$w \sim \mathcal{U}\left(-\sqrt{\frac{6}{n \cdot (1+a^2)}}, \sqrt{\frac{6}{n \cdot (1+a^2)}}\right)$$

where $\mathcal{U}(a, b)$ denotes the uniform distribution between a and b. n is the number of input units in the weight tensor. a is a scaling parameter, given as $\sqrt{5}$ in this case.

Given $a = \sqrt{5}$, we have $a^2 = 5$. Therefore, the range of the uniform distribution becomes:

$$w \sim \mathcal{U}\left(-\sqrt{\frac{6}{n \cdot 6}}, \sqrt{\frac{6}{n \cdot 6}}\right) = \mathcal{U}\left(-\sqrt{\frac{1}{n}}, \sqrt{\frac{1}{n}}\right)$$

The variance of a uniform distribution $\mathcal{U}(a, b)$ is given by:

$$\sigma^2(\mathcal{U}(a,b)) = \frac{(b-a)^2}{12}$$

For our distribution:

$$a = -\sqrt{\frac{1}{n}}, \quad b = \sqrt{\frac{1}{n}}$$

The range width $b - a$ is:

$$b - a = \sqrt{\frac{1}{n}} - \left(-\sqrt{\frac{1}{n}}\right) = 2\sqrt{\frac{1}{n}}$$

Thus, the variance is:

$$\sigma^2(w) = \frac{\left(2\sqrt{\frac{1}{n}}\right)^2}{12} = \frac{4 \cdot \frac{1}{n}}{12} = \frac{1}{3n}$$

D Further Extension of 3.2

In this section, we calculate output representation variance after a single attention head and self-attention. To simplify the equation, let softmax $\left(\frac{X_t W_{Q_i} (X_t W_{K_i})^T}{\sqrt{d_{\text{head}}}}\right)$ be A. For the simplicity of calculation, we assume $d_{\text{model}} = d_{\text{head}}$, which is a single-head attention. Because X is layer-normalized input, $\sigma^2(\mathrm{A}X_t)$ reaches the maximum value of 1 when the result of A is a one hot vector. Thus, the upper-bound variance of each head i and attention in the initialization stage of MLRA are as follows:

$$\begin{aligned}
\sigma^2(\text{head}(X_t)) &= \sigma^2(\mathrm{A}X_t) \cdot d_{\text{r}} \cdot d_{\text{model}} \cdot \sigma^2(W^U) \cdot \sigma^2(W^D) \\
&= \sigma^2(\mathrm{A}X_t) \cdot d_{\text{r}} \cdot d_{\text{model}} \cdot \frac{1}{3d_{\text{model}}} \cdot \frac{1}{3d_{\text{r}}} \\
&< \frac{1}{9}
\end{aligned} \tag{1}$$

$$\begin{aligned}
\sigma^2(\text{Attention}(X_t)) &= \sigma^2(\text{head}_i(X_t)) \cdot d_{\text{model}} \cdot \sigma^2(W_O) \\
&= \sigma^2(\text{head}_i(X_t)) \cdot d_{\text{model}} \cdot \frac{1}{3d_{\text{model}}} \\
&< \frac{1}{27}
\end{aligned} \tag{2}$$

where $\sigma^2(W_O) \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$. The calculation shows that attention weights $W \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ have a variance upper bound of $\frac{1}{9}$ per head and $\frac{1}{27}$ for the entire module. In contrast, MLRA's variance upper bound is one-third lower under Kaiming uniform initialization.

E Implementation Details

Configuration We measure TEV and apply MLRA to the widely adopted GPT-2 language model configuration (Radford et al., 2019) Specifically, we pre-train GPT-2 (Radford et al., 2019), σReparam (Zhai et al., 2023), and MLRA with hidden dimensions 384 and depth layers of 48, 96, 192 using the WebText dataset (Radford et al., 2019), where total number of token is 5.5B. Each model is trained with 4 epochs with the casual language modeling objective, as a recent study experimentally shows that repeating data more than 4 times in a decoder-only model with a data-constrained regime is computationally inefficient (Muennighoff et al., 2023). We set a batch size of 512 and a learning rate of 1e-3 for the base model. All model types in this paper follow the same training configuration for consistency.

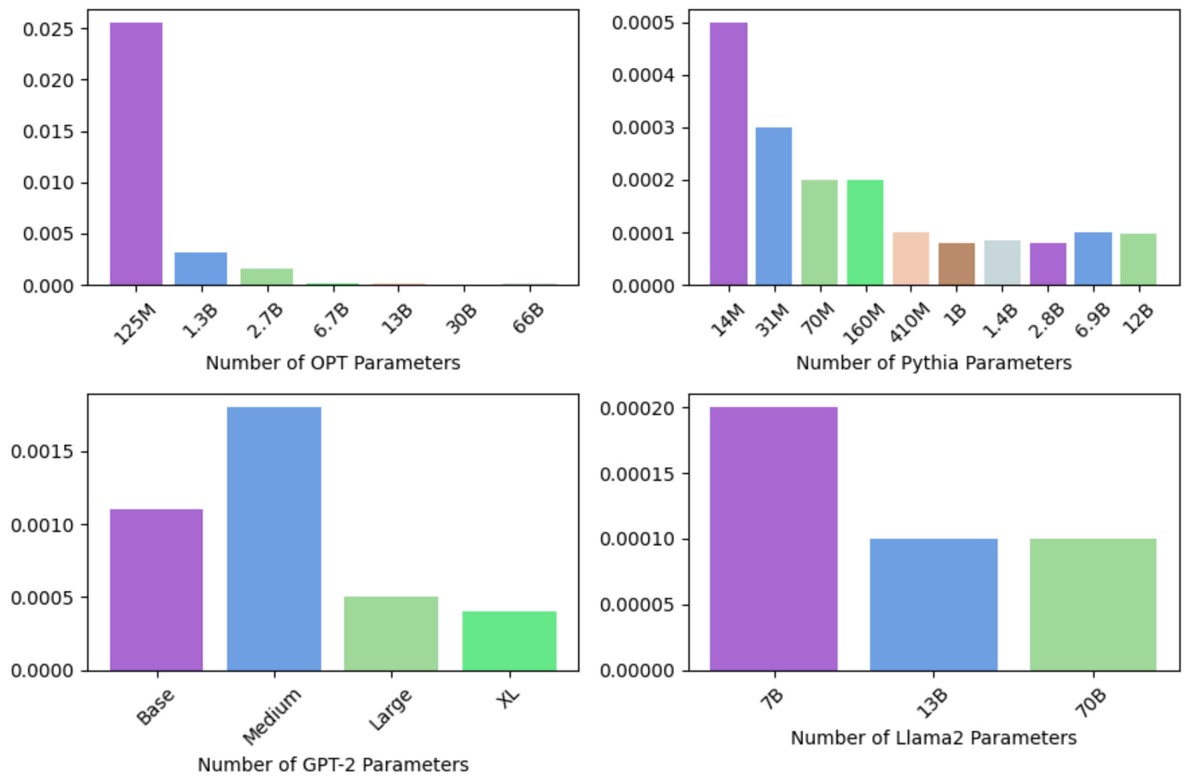

Figure 4: Row-wise average of the absolute mean value of $|V|$ token embeddings in the token embedding layer $\mathbf{E} \in \mathbb{R}^{|V| \times d_{\text{model}}}$ across OPT (Zhang et al., 2022), Pythia (Biderman et al., 2023), Llama-2 (Touvron et al., 2023b) and GPT-2 (Radford et al., 2019). $\mathbf{E}$ in pre-trained checkpoint remains centered around zero.

Model assessment For evaluation of GPT-2 models on the upstream language modeling tasks, we follow conventions in language modeling and report the perplexity, which measures average log probabilities of each sentence token predictions in an autoregressive way (Radford et al., 2019):

$$\mathrm{PPL}(W) = \exp\left(-\frac{1}{N}\sum_{t=1}^{N} \log P(x_t | x_{<t}; \Theta)\right) \tag{3}$$

where $\mathbf{x} = \{x_1, x_2...., x_N\}$ are the set of N tokens.