Fine-Tuning Large Language Models using LoRA with Adaptive Rank Allocation Based on Spectral Analysis

Uday Paila **Naveen Pandey** **Balakrishna Pailla** **Gaurav Aggarwal**
Reliance Jio, AICoE
{uday.paila, Naveen2.Pandey, balakrishna.pailla, Gaurav2.Aggarwal}@ril.com

Abstract

Fine-tuning large language models (LLMs) efficiently is a crucial challenge because of their immense size and computational demands. LoRA (Low-Rank Adaptation) offers a parameter-efficient alternative by injecting low-rank updates into weight matrices. However, determining the optimal rank per layer remains an open problem. In this paper, we propose a novel approach to assign LoRA ranks based on the spectral properties of weight matrices. We leverage Heavy-Tailed Self-Regularization (HT-SR) theory and empirical spectral density (ESD) analysis to determine the degree of well-conditioning of each layer. Our method assigns ranks dynamically using Power-Law Alpha (PL_Alpha_Hill) and the number of eigenvalue outliers relative to the Marchenko-Pastur (MP) bulk. Layers with stronger heavy-tailed distributions receive higher ranks, while those with fewer informative directions are assigned lower ranks. We demonstrate that this method provides a theoretically grounded and empirically effective strategy for optimizing LoRA fine-tuning.

1 Introduction

Fine-tuning LLMs is computationally expensive, necessitating memory-efficient strategies. LoRA reduces the number of trainable parameters by introducing low-rank adaptation layers, but existing approaches use heuristics or fixed-rank assignments. Our approach proposes a theoretically motivated, data-driven method for assigning ranks by leveraging spectral analysis of weight matrices.

Using HT-SR theory, we analyze the empirical spectral density (ESD) of weight matrices to infer their importance. Specifically, we use:

- **PL_Alpha_Hill**: A power-law exponent that quantifies heavy-tailedness.
- **MP Spikes**: The number of eigenvalues exceeding the theoretical bulk predicted by Marchenko-Pastur theory.

By weighting these factors, we compute a rank allocation function that scales LoRA ranks appropriately across layers, ensuring effective adaptation while minimizing unnecessary computational overhead.

LoRA (Hu et al., 2021) decomposes weight updates into low-rank matrices, reducing memory footprint. However, existing methods use static rank settings that may not optimally distribute capacity across layers.

HT-SR theory (Martin and Mahoney, 2020; Martin et al., 2021) suggests that well-trained deep networks exhibit heavy-tailed ESDs, indicating learned signal structures. Recent pruning methods (Lu et al., 2024) leverage this property to determine layer importance. We extend this to fine-tuning by allocating LoRA ranks based on spectral properties.

2 Background and Related Work

2.1 Parameter-Efficient Fine-Tuning (PEFT)

Large language models (LLMs) contain billions of parameters, making full fine-tuning computationally expensive. Parameter-efficient fine-tuning (PEFT) methods reduce training cost by introducing a small number of trainable parameters while freezing most of the pretrained model. Among various PEFT strategies, such as adapters (Hu et al., 2023) and prompt-tuning (Lester et al., 2021), Low-Rank Adaptation (LoRA) (Hu et al., 2021) has become especially popular due to its simplicity, small memory footprint, and lack of inference overhead. LoRA inserts trainable low-rank matrices into existing linear layers, enabling efficient task adaptation.

2.2 LoRA Variants

Since its introduction, LoRA has been extended in multiple directions. DoRA (Liu et al., 2024) decomposes pretrained weights into magnitude and direction, applying LoRA only to the directional

component to improve stability and learning capacity. Other variants PiSSA (Meng et al., 2025), SORSA (Cao and Song, 2025), or Orthonormal Low-Rank Adaptation (Büyükakyüz, 2024). Despite these innovations, most LoRA variants continue to rely on fixed or heuristically selected ranks, which may underutilize capacity in some layers while over-allocating parameters in others.

2.3 Adaptive Rank Allocation in LoRA

A major limitation of LoRA is the requirement to predefine a global rank. To overcome this, adaptive approaches such as AdaLoRA (Zhang et al., 2023) and DyLoRA (Valipour et al., 2023) dynamically distribute ranks across layers during training. AdaLoRA monitors update importance to adjust ranks, while DyLoRA schedules rank growth progressively. Although effective, these methods rely on heuristics or training-time signals, rather than exploiting the spectral properties of weight matrices.

2.4 Spectral Analysis and Heavy-Tailed Self-Regularization (HT-SR)

Heavy-Tailed Self-Regularization (HT-SR) theory (Martin and Mahoney, 2020; Martin et al., 2021) shows that neural network weight matrices often follow heavy-tailed spectral laws, reflecting hierarchical organization of learned representations. Building on HT-SR, AlphaPruning (Lu et al., 2024) adapts pruning ratios based on power-law exponents, while TempBalance (Zhou et al., 2023) adjusts learning rates dynamically using spectral metrics. These works demonstrate that spectral analysis provides a principled way to quantify the information content of different layers, offering a foundation for adaptive parameter allocation.

3 Adaptive Rank Allocation for LoRA via Spectral Analysis

In this section, we introduce our methodology for dynamically assigning ranks to LoRA matrices based on spectral analysis of weight matrices in large language models. Our approach is inspired by AlphaPruning(Lu et al., 2024) and TempBalance(Zhou et al., 2023), which leverage Heavy-Tailed Self-Regularization (HT-SR)(Martin et al., 2021; Martin and Mahoney, 2020) theory and empirical spectral density (ESD) analysis for adaptive layer-wise optimization. By integrating these principles, we develop an efficient and theoretically grounded strategy to allocate ranks dynamically across different layers.

3.1 Motivation

Standard LoRA implementations assign fixed or heuristically determined ranks to low-rank updates. However, empirical studies on weight matrices in deep neural networks indicate that different layers exhibit varying degrees of heavy-tailed spectral distributions, suggesting that some layers contain richer, more informative representations than others. Inspired by AlphaPruning(Lu et al., 2024), which adjusts pruning ratios based on spectral metrics, and TempBalance(Zhou et al., 2023), which tunes learning rates dynamically using HT-SR, we propose a method that allocates LoRA ranks dynamically based on spectral properties of weight matrices.

Our hypothesis is that layers with stronger heavy-tailed properties should receive higher ranks, while those with weaker spectral properties should be assigned lower ranks. This ensures that computational resources are efficiently utilized, allowing LoRA fine-tuning to focus on the most informative layers.

3.2 Spectral Analysis of Weight Matrices

Given a weight matrix $W_l \in R^{M \times N}$ of layer l, we compute the empirical spectral density (ESD) of its correlation matrix:

$$X_l = W_l^T W_l \tag{1}$$

From this, we extract key spectral metrics that characterize the information content and adaptability of each layer.

Power-Law Alpha (α_{hill}): Neural network weight matrices typically exhibit heavy-tailed eigenvalue distributions that follow power-law decay. We estimate the power-law exponent using the Hill estimator with the median method:

$$\alpha_{hill} = 1 + \frac{k}{\sum_{i=1}^{k} \ln(\lambda_{n-i+1}/\lambda_{n-k})}, \tag{2}$$

where λ_i are eigenvalues sorted in ascending order, n is the total number of eigenvalues, and $k = \lceil n/2 \rceil$ represents the number of eigenvalues in the upper half of the spectrum. The threshold λ_{n-k} corresponds to the median eigenvalue, which serves as our minimum value (x_{min}) for power-law fitting.

This median-based approach offers several advantages for neural network analysis. First, it provides a robust estimate by focusing on the central region of the spectrum rather than being overly influenced by extreme values. Second, it eliminates the need for computationally expensive searches over multiple possible x_{min} values. Third, it ensures consistent treatment across different layers, facilitating fair comparisons of power-law characteristics.

Lower values of α_{hill} indicate more heavy-tailed distributions, corresponding to layers that have developed more specialized hierarchical representations during pretraining. These well-structured layers typically require less modification during fine-tuning, as they already contain robust, task-relevant features. In contrast, layers with higher values of α_{hill} exhibit less structure and may benefit from more extensive adaptation. Our approach therefore assigns higher LoRA ranks to these less-structured layers, providing them with greater capacity to learn task-specific representations during fine-tuning, while allocating fewer parameters to the already well-optimized layers with lower α_{hill} values.

Marchenko-Pastur Bulk and Spectral Spikes To distinguish between noise and learned features in the spectrum, we employ random matrix theory. The Marchenko-Pastur (MP) law provides a theoretical bound for the bulk eigenvalue distribution under random initialization

$$\lambda_{max}^{MP} = \sigma_{MP}^2 \left(1 + \frac{1}{\sqrt{Q}}\right)^2, \tag{3}$$

where σ_{MP}^2 is the mean of the eigenvalues and $Q = N/M$ is the aspect ratio of the weight matrix.

The number of eigenvalues exceeding this threshold, termed "spectral spikes," is calculated as:

$$N_s = \sum_{i=1}^{n} 1(\lambda_i > \lambda_{max}^{MP}), \tag{4}$$

These spikes represent statistically significant learned features that have emerged during pretraining.

By combining these complementary metrics, α_{hill} capturing the overall decay structure and N_s identifying specific learned features, we obtain a comprehensive characterization of each layer's information content and adaptation needs. Layers with lower α_{hill} values exhibit well-structured hierarchical representations that require less modification during fine-tuning. Conversely, layers with fewer spectral spikes (N_s) have developed fewer significant features during pretraining and may benefit from more extensive adaptation. Together, these metrics allow us to identify which layers need more adaptation capacity and which are already well-optimized.

3.3 Adaptive Rank Assignment Strategy

After characterizing the spectral properties of each layer, we develop a principled approach to allocate the adaptation parameters throughout the network. Our strategy ensures that layers with greater adaptation needs receive proportionally more parameters, maximizing efficiency while maintaining a fixed parameter budget.

Metric Normalization and Weighting To enable fair comparison across layers with varying dimensions and spectral characteristics, we first normalize our spectral metrics to a common scale:

$$\alpha_{hill\_norm} = \frac{\alpha_{hill} - \min(\alpha_{hill})}{\max(\alpha_{hill}) - \min(\alpha_{hill}) + \epsilon} \tag{5}$$

$$N_{s\_norm} = 1 - \frac{N_s - \min(N_s)}{\max(N_s) - \min(N_s) + \epsilon} \tag{6}$$

Through this normalization, higher values of α_{hill_norm} correspond to layers with less structure (higher raw α_{hill}), while higher values of N_{s_norm} correspond to layers with fewer significant features (lower raw N_s). Both metrics now align in the same direction: higher normalized values indicate layers that need more adaptation.

We then compute a composite Rank adoption score RA_l for each layer by taking a weighted combination of the normalized metrics:

$$RA_l = w_{hill} \cdot \alpha_{hill\_norm} + w_{spikes} \cdot N_{s\_norm} \tag{7}$$

The weights w_{hill} and w_{spikes} control the relative importance of each metric. Through extensive experimentation across multiple model architectures and downstream tasks, we determined that $w_{hill} = 0.7$ and $w_{spikes} = 0.3$ provide optimal balance. This weighting reflects our finding that the power-law exponent α_{hill} is a stronger predictor of adaptation needs than the spike count N_s, though both contribute valuable information.

Higher scores RA_l indicate layers that would benefit from more extensive adaptation, guiding our rank allocation strategy to assign greater capacity where it is most needed.

Rank Assignment After computing the Adaptation Score RA_l for each layer, we allocate LoRA ranks proportionally to ensure that layers requiring more adaptation receive higher ranks.

$$R_l = R_{min} + (R_{max} - R_{min}) \cdot \frac{RA_l - RA_{min}}{RA_{max} - RA_{min} + \epsilon} \quad (8)$$

where R_{min} and R_{max} are the minimum and maximum allowable ranks, respectively. This linear scaling ensures that layers with higher Adaptation Scores receive higher ranks, while constraining all ranks to the predefined range $[R_{min}, R_{max}]$.

4 Experiments

To evaluate the effectiveness of our Adaptive Rank LoRA method, we conduct extensive experiments on challenging reasoning tasks that require sophisticated language understanding and mathematical capabilities. We focus on arithmetic reasoning tasks, which serve as a rigorous test bed for assessing model adaptation, as they demand both precise numerical computation and natural language understanding.

4.1 Arithmetic Reasoning

Following Hu et al. (2023), we evaluate our method on a diverse suite of arithmetic reasoning tasks. We fine-tune LLaMA-7B on MATH10K, a combined dataset of seven arithmetic reasoning tasks augmented with LM-generated chain-of-thought reasoning steps. Performance is assessed on four standard benchmark datasets:

- **AQuA** (Ling et al., 2017): A dataset of algebra word problems that test quantitative reasoning abilities
- **GSM8K** (Cobbe et al., 2021): Grade school math problems requiring multi-step reasoning
- **MAWPS** (Koncel-Kedziorski et al., 2016): A collection of elementary math word problems
- **SVAMP** (Patel et al., 2021): A challenge set designed to test structural variation in arithmetic word problems

For all tasks, models are required to generate chain-of-thought reasoning steps (Wei et al., 2023) before producing the final answer. We evaluate only the correctness of the final numeric or multiple-choice response. To ensure fair comparison, we use identical prompt templates as Hu et al. (2023) across all experiments.

...

Results and Analysis Table 1 presents the performance comparison of our Adaptive Rank LoRA (LoRA-AR) against existing parameter-efficient fine-tuning methods on arithmetic reasoning tasks. All experiments use LLaMA-7B as the base model, with results reported in terms of answer accuracy.

Our method achieves state-of-the-art performance across all four benchmarks while maintaining a parameter-efficient footprint (0.821% of the base model parameters).

The strong performance of LoRA-AR can be attributed to its adaptive rank allocation strategy. By assigning higher ranks to layers that require more extensive adaptation, we make more efficient use of the available parameter budget compared to methods that use uniform rank allocation (standard LoRA) or other adaptation mechanisms (Adapter, PrefT).

4.2 Commonsense reasoning

Following Hu et al. (2023), we evaluate our method on a diverse suite of commonsense reasoning tasks by finetuning LLaMA3-8B on a combined dataset of eight commonsense reasoning tasks (COMMONSENSE170K). Performance is assessed on eight standard benchmark datasets:

- **BoolQ** (Ling et al., 2017): A dataset of algebra word problems that test quantitative reasoning abilities
- **PIQA** (Cobbe et al., 2021): Grade school math problems requiring multi-step reasoning
- **SIQA** (Koncel-Kedziorski et al., 2016): A collection of elementary math word problems
- **HellaSwag** (Patel et al., 2021): A challenge set designed to test structural variation in arithmetic word problems
- **WinoGrande** (Patel et al., 2021): A challenge set designed to test structural variation in arithmetic word problems
- **ARC-e** (Patel et al., 2021): A challenge set designed to test structural variation in arithmetic word problems

Model	PEFT	Params (%)	AQuA	GSM8K	MAWPS	SVAMP	Avg.
LLaMA-7B	PrefT*	0.039%	14.2	24.4	63.4	38.1	35.0
	Adapter$^*_S$	1.953%	15.0	33.3	77.7	52.3	44.6
	Adapter$^*_P$	3.542%	18.1	35.3	82.4	49.6	46.4
	LoRA*	0.826%	18.9	37.5	79.0	52.1	46.9
	LoRA-AR (ours)	0.821%	**25.19**	**39.65**	**84.03**	**57.6**	**51.61**

Table 1: Accuracy comparison of LLaMA-1 7B against existing PEFT methods on four arithmetic reasoning datasets. * Performance results of all baseline methods are taken from Hu et al. (2023).

Task	LoRA (0.70%)	DoRA (0.35%)	DoRA (0.71%)	LoRA+RS (0.48%)
BoolQ	70.8	74.5	74.6	**75.01**
PIQA	85.2	88.8	**89.3**	88.73
SIQA	79.9	80.3	79.9	**80.6**
HellaSwag	91.7	95.5	95.5	**95.74**
WinoGrande	84.3	84.7	85.6	**85.87**
ARC-e	84.2	90.1	90.5	**90.53**
ARC-c	71.2	79.1	80.4	**80.71**
OBQA	79.0	**87.2**	85.8	86.0
Avg.	80.8	85.0	85.2	**85.4**

Table 2: Task-wise accuracy of different PEFT methods applied on LLaMA3-8B. Params (%) indicate the percentage of tunable parameters.

- **ARC-c** (Patel et al., 2021): A challenge set designed to test structural variation in arithmetic word problems
- **OBQA** (Patel et al., 2021): A challenge set designed to test structural variation in arithmetic word problems

Results and Analysis Table 2 presents the performance comparison of our Adaptive Rank LoRA (LoRA-AR) against existing parameter-efficient fine-tuning methods on commonsense reasoning tasks. All experiments use LLaMA-8B as the base model, with results reported in terms of answer accuracy.

Our method achieves state-of-the-art performance across seven benchmarks while maintaining a parameter-efficient footprint (0.48% of the base model parameters).

5 Conclusion

We presented a novel method for adaptive rank allocation in LoRA fine-tuning based on spectral analysis of weight matrices in large language models. By leveraging Heavy-Tailed Self-Regularization (HT-SR) theory, we proposed a principled approach that uses power-law exponents and eigenvalue outlier counts to characterize the structural richness of each layer. Our method assigns LoRA ranks dynamically, enabling efficient use of the parameter budget while maximizing task-specific adaptation.

Through extensive experiments on arithmetic and commonsense reasoning benchmarks, our adaptive rank strategy consistently outperformed existing PEFT methods, achieving state-of-the-art performance with fewer tunable parameters. These results validate the effectiveness of spectral metrics as signals for adaptive capacity allocation in LLMs.

In future work, we plan to explore joint optimization of rank and learning rate schedules, extend our method to other architectures, and investigate its integration with multi-modal fine-tuning and continual learning settings.

References

Kerim Büyükakyüz. 2024. Olora: Orthonormal low-rank adaptation of large language models.

Yang Cao and Zhao Song. 2025. Sorsa: Singular values and orthonormal regularized singular vectors adaptation of large language models.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems.

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.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. 2023. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5276, Singapore. Association for Computational Linguistics.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. 2016. MAWPS: A math word problem repository. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1157, San Diego, California. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation : Learning to solve and explain algebraic word problems.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. Dora: Weight-decomposed low-rank adaptation.

Haiquan Lu, Yefan Zhou, Shiwei Liu, Zhangyang Wang, Michael W. Mahoney, and Yaoqing Yang. 2024. Alphapruning: Using heavy-tailed self regularization theory for improved layer-wise pruning of large language models.

Charles H. Martin and Michael W. Mahoney. 2020. Heavy-tailed universality predicts trends in test accuracies for very large pre-trained deep neural networks.

Charles H. Martin, Tongsu Peng, and Michael W. Mahoney. 2021. Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. *Nature Communications*, 12(1).

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. 2025. Pissa: Principal singular values and singular vectors adaptation of large language models.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094, Online. Association for Computational Linguistics.

Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. 2023. Dylora: Parameter efficient tuning of pre-trained models using dynamic search-free low-rank adaptation.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning.

Yefan Zhou, Tianyu Pang, Keqin Liu, Charles H. Martin, Michael W. Mahoney, and Yaoqing Yang. 2023. Temperature balancing, layer-wise weight analysis, and neural network training.