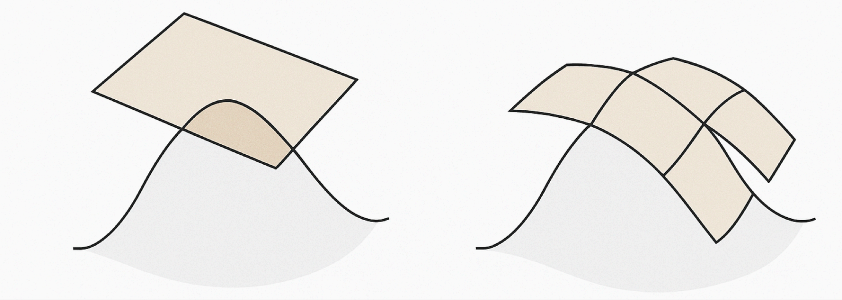
AdaMSS: Adaptive Multi-Subspace Approach for Parameter-Efficient Fine-Tuning

Jingjing Zheng, Ph.D student

Content

- Background and Motivation
- Proposed Methodology (AdaMSS)
- Theoretical Guarantees
- Experimental Results
- Future Work

Motivation



(a) (b)

Figure 2: (a) Single subspace-based Methods; (b) Multi-subspace-based Methods.

- Existing methods assume a single low-dimensional subspace for weight updates.
- Trade-off: Larger subspace increase model fixability but also increase the raise computational cost.
- Can we preserve expressiveness without increasing the parameter count?
- **Multi-subspace models** offer greater flexibility and an enhanced ability to capture diverse information.

Proposed Methodology: AdaMSS

Does Model weights follows the multi-subspace distribution?

- Low rank Representation (LLR) is used to study the multi-subspace distribution in weights:

$$\min_{\boldsymbol{Z}} \operatorname{rank}(\boldsymbol{Z}) \quad s.t.\ \hat{\boldsymbol{W}}_0 = \hat{\boldsymbol{W}}_0 \boldsymbol{Z}$$

The optimal solution $Z_0^\star$ is **low rank!**

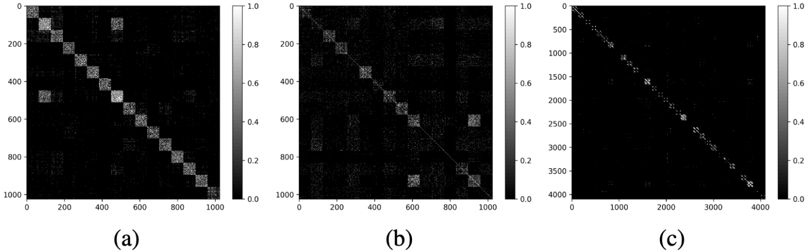
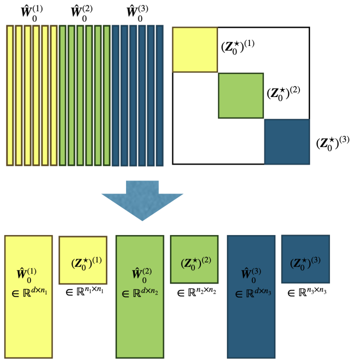


Figure 3: Illustration of multiple subspaces structure in pretrained network weights: an approximate block-diagonal structure of $Z_0^\star$ from the first layer of the pretrained (a) ViT-Large (query), (b) Roberta-Large (query), and (c) LLaMA 2-7B (query).

The optimal solution $Z_0^\star$ also exhibits **an approximate block-diagonal structure**, indicating that the columns of weights are **grouped into different subspaces.**

Proposed Methodology: AdaMSS



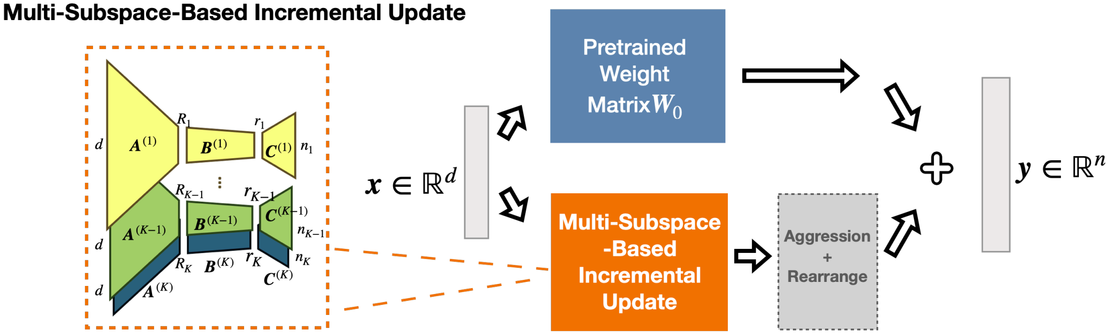


Figure 4: Illustration of the proposed Multi-Subspace-Based Adaptation Framework

Example:

$$\hat{W}_0^{(1)} \approx \hat{W}_0^{(1)}(Z_0^\star)^{(1)} = A^{(1)}B_w^{(1)}B_z^{(1)}C^{(1)} = A^{(1)}B^{(1)}C^{(1)}$$

$$rank(\hat{W}_0^{(1)}) = R_1, rank((Z_0^\star)^{(1)}) = r_1$$

Only $H^{(k)} = [B^{(k)}; (C^{(k)})^\top] \in \mathbb{R}^{(R_k+n_k)\times r_k}$ are trainable.

- **Expressiveness ↔ Efficiency:** Reduces the number of trainable parameters from $(d+n)r$ to $\sum_{k=1}^{K}(n_k r_k + r_k R_k) \leq r \max_{k=1,2,\cdots,K}(n_k + R_k)$ while maintaining the expressiveness.
- **AdaMSS: the proposed Multi-Subspace-Based Adaptation Framework with** Adaptive budget allocation during training by calculating the importance score.

Theoretical Guarantees

Theorem1. *Let* $g(\cdot)$ *be a* $\mathscr{L}(g)$*-Lipschitz loss function from* $(f_{\mathbf{w}}(\mathbf{x}), \mathbf{y})$ *to* $[0,1]$*, where* $f_{\mathbf{w}} \in \mathscr{F}_{\text{AdaMSS}} = \{f_{\boldsymbol{W}}(\boldsymbol{x}) = \phi(\boldsymbol{x}\boldsymbol{W}) \mid \boldsymbol{W} = \boldsymbol{W}_0 + \boldsymbol{A}\boldsymbol{B}\boldsymbol{C}, (\boldsymbol{A}^{(k)})^\top \boldsymbol{A}^{(k)} = \boldsymbol{I}, \|\boldsymbol{B}^{(k)}\boldsymbol{C}^{(k)}\|_2 \leq B_k\}$ *and* $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{X} \times \mathbb{Y}$*,* $\mathbb{X} \subseteq \mathbb{R}^d$ *and* $\mathbb{Y}$ *are feature space and output space, respectively. For any* $\delta > 0$*, the following holds with probability at least* $1 - \delta$ *for a randomly chosen i.i.d. samples* $\mathbb{S} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$*:*

$$\mathbb{E}[g(f_{\mathbf{w}}(\mathbf{x}), \mathbf{y})] \leq \frac{1}{m}\sum_{i=1}^{m} g(f_{\mathbf{w}}(\mathbf{x}_i), \mathbf{y}_i) + \sqrt{\mathscr{L}(g)\pi}\,\mathscr{L}(\phi)\sum_{k=1}^{K} \hat{R} B_k \sqrt{\frac{r_k n_k}{m}} + \sqrt{\frac{9\log\frac{2}{\delta}}{2m}},$$

where $n = \sum_{k=1}^{K} n_k$*,* n_k *denotes the width of the weight matrix* $\boldsymbol{C}^{(k)}$*,* $\mathscr{L}(\phi)$ *is Lipschitz constant for function* ϕ*,* $\boldsymbol{X} = [\boldsymbol{x}_1^\top, \ldots, \boldsymbol{x}_m^\top] \in \mathbb{R}^{d\times m}$ *for the samples* $\{\boldsymbol{x}_i\}_{I=1}^m$*, and* $\max_{i=1,2,\cdots,m} \|\boldsymbol{x}_i\| \leq \hat{R}$*.*

AdaMSS achieves better generalization performance than single-subspace methods under the same conditions.

Experimental Results

Model	Method	# Trainable Parameters	SST-2 Acc.	MRPC Acc.	CoLA MCC	QNLI Acc.	RTE Acc.	STS-B PCC	Avg.
RoBERTa-Base	FF	125M	94.8	**90.2**	63.6	92.8	78.7	91.2	85.2
	LoRA	0.3M	**95.1**$_{\pm0.2}$	89.7$_{\pm0.7}$	63.4$_{\pm1.2}$	**93.3**$_{\pm0.3}$	78.4$_{\pm0.8}$	**91.5**$_{\pm0.2}$	85.2
	AdaLoRA	0.3M	94.5$_{\pm0.2}$	88.7$_{\pm0.5}$	62.0$_{\pm0.6}$	93.1$_{\pm0.2}$	**81.0**$_{\pm0.6}$	90.5$_{\pm0.2}$	85.0
	DyLoRA	0.3M	94.3$_{\pm0.5}$	89.5$_{\pm0.5}$	61.1$_{\pm0.3}$	92.2$_{\pm0.5}$	78.7$_{\pm0.7}$	91.1$_{\pm0.6}$	84.5
	PiSSA ($r=8$)	0.3M	93.9$_{\pm0.1}$	89.3$_{\pm0.8}$	62.1$_{\pm2.9}$	91.3$_{\pm0.1}$	77.3$_{\pm1.4}$	90.5$_{\pm0.2}$	84.1
	LoRA-PRO	0.3M	94.2$_{\pm0.3}$	90.1$_{\pm0.5}$	64.3$_{\pm0.72}$	92.0$_{\pm0.2}$	80.2$_{\pm1.8}$	90.9$_{\pm0.22}$	**85.3**
	LoRA ($r=1$)	0.055M	93.7$_{\pm0.5}$	89.2$_{\pm0.3}$	62.3$_{\pm3.6}$	90.6$_{\pm0.4}$	79.5$_{\pm0.4}$	80.8$_{\pm20.6}$	82.7
	PiSSA ($r=1$)	0.055M	93.3$_{\pm0.2}$	89.3$_{\pm0.6}$	62.6$_{\pm1.4}$	90.6$_{\pm0.4}$	74.9$_{\pm1.2}$	90.0$_{\pm0.3}$	83.4
	LoRETTA	0.057M	94.6$_{\pm0.5}$	88.3$_{\pm0.7}$	61.8$_{\pm1.3}$	92.7$_{\pm0.2}$	75.1$_{\pm5.3}$	90.5$_{\pm0.1}$	83.8
	WeGeFT	0.049M	94.1$_{\pm0.5}$	89.5$_{\pm0.5}$	63.5$_{\pm1.3}$	91.2$_{\pm0.4}$	78.6$_{\pm1.6}$	90.5$_{\pm0.1}$	84.6
	AdaMSS$_{\text{base}}$ ($r_k=1$)	0.042M	94.6$_{\pm0.2}$	89.2$_{\pm1.0}$	64.3$_{\pm0.9}$	92.4$_{\pm0.1}$	77.2$_{\pm0.7}$	90.6$_{\pm0.1}$	84.7
	AdaMSS ($r_k=1$)	**0.032M**	94.6$_{\pm0.2}$	88.8$_{\pm1.4}$	**64.5**$_{\pm1.1}$	92.4$_{\pm0.1}$	77.3$_{\pm0.7}$	90.4$_{\pm0.1}$	84.7
RoBERTa-Large	FF	356M	**96.4**	**90.9**	68	94.7	86.6	**92.4**	**88.2**
	LoRA	0.8M	96.2$_{\pm0.5}$	90.2$_{\pm1.0}$	68.2$_{\pm1.9}$	**94.8**$_{\pm0.3}$	85.2$_{\pm1.1}$	92.3$_{\pm0.5}$	87.8
	PiSSA ($r=8$)	0.8M	95.5$_{\pm0.2}$	86.9$_{\pm2.6}$	61.1$_{\pm3.4}$	92.1$_{\pm1.7}$	56.8$_{\pm8.2}$	91.8$_{\pm0.4}$	80.7
	LoRA-PRO	0.8M	95.9$_{\pm0.2}$	**90.9**$_{\pm0.4}$	66.7$_{\pm2.0}$	93.0$_{\pm0.5}$	60.5$_{\pm13.5}$	92.0$_{\pm0.1}$	83.2
	LoRA ($r=1$)	0.147M	95.7$_{\pm0.4}$	88.3$_{\pm0.7}$	62.2$_{\pm2.4}$	93.9$_{\pm0.2}$	82.2$_{\pm2.5}$	78.2$_{\pm29.7}$	83.4
	PiSSA ($r=1$)	0.147M	95.2$_{\pm0.2}$	84.9$_{\pm3.4}$	56.6$_{\pm6.2}$	93.4$_{\pm0.3}$	65.9$_{\pm11.3}$	91.3$_{\pm0.2}$	81.2
	LoRETTA	0.132M	96.2$_{\pm0.2}$	90.5$_{\pm0.4}$	**69.5**$_{\pm0.6}$	94.1$_{\pm0.9}$	53.0$_{\pm0.5}$	92.0$_{\pm0.2}$	82.6
	WeGeFT	0.065M	95.0$_{\pm0.3}$	75.7$_{\pm7.7}$	64.0$_{\pm2.0}$	93.7$_{\pm0.3}$	53.6$_{\pm1.2}$	91.4$_{\pm0.3}$	78.9
	AdaMSS$_{\text{base}}$ ($r_k=1$)	0.097M	96.3$_{\pm0.2}$	90.5$_{\pm0.3}$	68.0$_{\pm0.9}$	94.6$_{\pm0.1}$	**87.3**$_{\pm1.0}$	92.0$_{\pm0.0}$	88.1
	AdaMSS ($r_k=1$)	**0.045M**	96.1$_{\pm0.0}$	90.3$_{\pm0.5}$	67.2$_{\pm1.2}$	94.5$_{\pm0.1}$	87.1$_{\pm2.1}$	91.9$_{\pm0.0}$	87.9

Table 1: Natural Language Understanding on GLUE.

AdaMSS achieves accuracy comparable to other PEFT methods while using the fewest trainable parameters.

AdaMSS outperforms most baselines by around 5% in accuracy while using fewer parameters.

Experimental Results

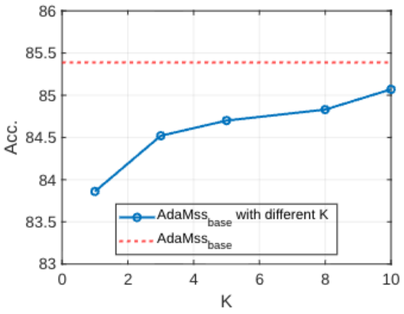
Table 3: Natural Language Generation.

Model	Method	Trainable Parameters	GSM8K	MATH
LLaMA 2-7B	Full FT	6738M	49.05	7.22
	LoRA*	320M	42.30	5.50
	PiSSA* ($r=8$)	19M	44.11	5.84
	LoRA-PRO* ($r=8$)	19M	46.61	6.4
	PiSSA ($r=8$)	4M	36.39	5.35
	LoRETTA ($r=5$)	**0.3M**	37.86	4.6
	AdaMSS*$_{\text{base}}$ ($r_k=3$)	4M	**51.10**	**7.57**
	AdaMSS* ($r_k=3$)	4M	50.80	7.22
	AdaMSS$_{\text{base}}$ ($r_k=3$)	0.8M	44.41	6.05
Mistral-7B	Full FT	7242M	67.02	18.6
	LoRA*	168M	67.70	19.68
	PiSSA* ($r=8$)	20M	**71.00**	20.40
	LoRA-PRO* ($r=8$)	20M	69.59	19.17
	PiSSA ($r=8$)	3M	64.26	16.87
	LoRETTA ($r=5$)	**0.3M**	62.6	15.6
	AdaMSS*$_{\text{base}}$ ($r_k=3$)	4M	70.71	**20.44**
	AdaMSS* ($r_k=3$)	2M	70.74	19.47
	AdaMSS$_{\text{base}}$ ($r_k=3$)	0.5M	65.43	17.74
Gemma-7B	Full FT	8538M	71.34	22.74
	LoRA*	200M	74.90	31.28
	PiSSA* ($r=8$)	25M	75.48	29.59
	LoRA-PRO*	25M	75.90	29.25
	PiSSA ($r=8$)	3M	71.52	27.53
	LoRETTA ($r=5$)	**0.2M**	70.23	26.28
	AdaMSS*$_{\text{base}}$ ($r_k=3$)	6M	75.33	**29.73**
	AdaMSS* ($r_k=3$)	4M	**76.41**	28.64
	AdaMSS$_{\text{base}}$ ($r_k=3$)	0.7M	70.86	27.38

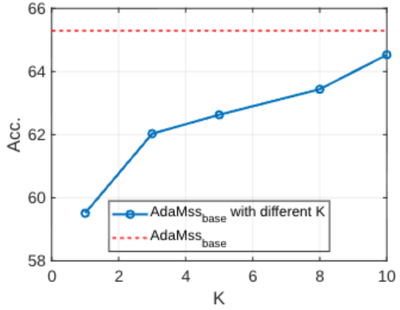
AdaMSS outperforms most baseline by around 5% in accuracy, while using fewer parameters.

AdaMSS achieves the best performance, using less than 1% of the parameters required for Full FF

Figure 5: Ablation study for subspace number K.



(a) StanfordCars



(b) FGVC

Future Work

- Beyond Fine-Tuning: Extend the multi-subspace perspective to **network compression** and **model pruning**.
- **Theoretical insights** of multi-subspace structures in the network weights.

Reference

- [Houlsby et al., ICML 2019] Houlsby, Neil and Giurgiu, Andrei and Jastrzebski, Stanislaw and Morrone, Bruna and De Laroussilhe, Quentin and Gesmundo, Andrea and Attariyan, Mona and Gelly, Sylvain. Parameter-Efficient Transfer Learning for NLP. In *International conference on machine learning*, 2019.
- [Lester et al., EMNLP 2021] Brian Lester, Rami Al-Rfou, Noah Constant. The Power of Scale for Parameter-Efficient Prompt Tuning.
- [Li et al., ACL 2021] Xiang Lisa Li, Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation.
- [Hu et al., ICLR 2022] Edward J Hu,Yelong Shen,Phillip Wallis,Zeyuan Allen-Zhu,Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: low-rank adaptation of large language models. In *International conference on Learning Representations*, *2022*.
- [Zhang et al., ICLR 2023] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, Tuo Zhao. AdaLoRA: Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning.
- [Meng et al., NeurIPS 2024] Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: principal singular values and singular vectors adaptation of large language models. In *Neural Information Processing Systems*, 2024.
- [Zhang et al., ICML 2022] Qingru Zhang, Simiao Zuo, Chen Liang, Alexander Bukharin, Pengcheng He, Weizhu Chen, and Tuo Zhao. Platon: Pruning large transformer models with upper confidence bound of weight importance. In *International conference on machine learning*, pages 26809–26823. PMLR, 2022.
- [Wang et al., NeurIPS 2024] Shaowen Wang, Linxi Yu, and Jian Li. Lora-ga: Low-rank adaptation with gradient approximation. In *Neural Information Processing Systems*, 2024.
- [Liu et al., TPAMI 2012] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2012.

- [Wang et al., NeurIPS 2024] Shaowen Wang, Linxi Yu, and Jian Li. Lora-ga: Low-rank adaptation with gradient approximation. In *Neural Information Processing Systems*, 2024.

- [Zheng et al., NeurIPS 2025] Jingjing Zheng, Wangling Lu, Yiming Dong, Chaojie Ji, Yankai Cao, Zhouchen Lin. AdaMSS: Adaptive Multi-Subspace Approach for Parameter-Efficient Fine-Tuning, In *Neural Information Processing Systems*, 2025.

Thanks for your attention!