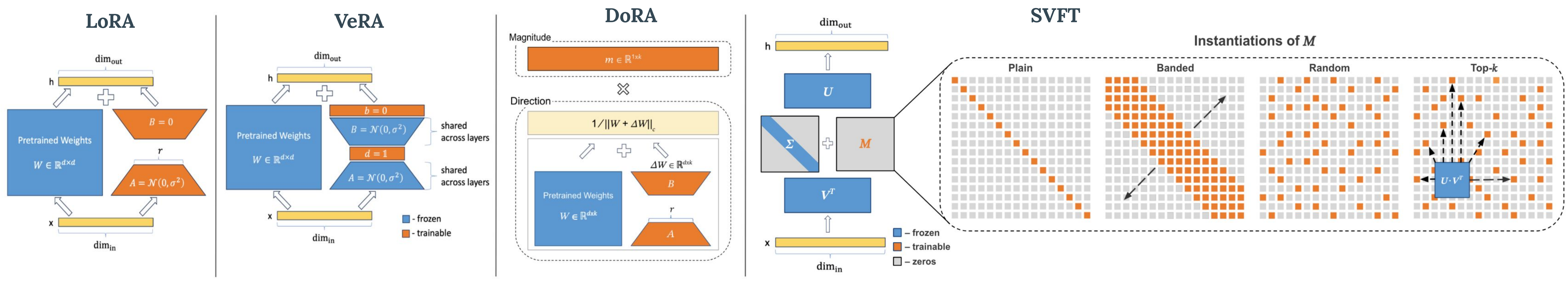
SVFT: Parameter-Efficient Fine-Tuning with Singular Vectors

Vijay Lingam[1*], Atula Tejaswi[1*], Aditya Vavre[1*], Aneesh Shetty[1*], Gautham Krishna Gudur[1*], Joydeep Ghosh[1], Eunsol Choi[1], Alex Dimakis[1], Aleksandar Bojchevski[2], Sujay Sanghavi[1]
University of Texas at Austin[1], University of Cologne[2]   *Equal Contribution



Background

LoRA-like parameter-efficient fine-tuning (PEFT) methods freeze pre-trained model weights $\boldsymbol{W}$ and inject learnable matrices $\Delta\boldsymbol{W}$

LoRA. The weight update $\Delta\boldsymbol{W}$ is constrained to a low-rank decomposition: $\boldsymbol{h} = \boldsymbol{W}_0\boldsymbol{x} + \Delta\boldsymbol{W}\boldsymbol{x} = \boldsymbol{W}_0\boldsymbol{x} + \underline{\boldsymbol{B}\boldsymbol{A}}\boldsymbol{x}$, $\boldsymbol{B} \in \mathbb{R}^{d \times r}$, $\boldsymbol{A} \in \mathbb{R}^{r \times n}$, $r \ll \min(d, n)$

VeRA. A pair of low-rank random matrices is shared between layers and compact scaling vectors are learned: $\boldsymbol{h} = \boldsymbol{W}_0\boldsymbol{x} + \Delta\boldsymbol{W}\boldsymbol{x} = \boldsymbol{W}_0\boldsymbol{x} + \underline{\boldsymbol{\Lambda}_b}\boldsymbol{B}\underline{\boldsymbol{\Lambda}_d}\boldsymbol{A}\boldsymbol{x}$, where $\boldsymbol{A}$ and $\boldsymbol{B}$ are initialized randomly, frozen, and shared across layers, while $\boldsymbol{\Lambda}_b$ and $\boldsymbol{\Lambda}_d$ are trainable diagonal matrices

DoRA. Decomposes pre-trained weight matrices into magnitude and direction components, and applies low-rank updates for directional updates: $\boldsymbol{h} = \underline{\boldsymbol{m}}\frac{\boldsymbol{W}_0 + \Delta\boldsymbol{W}}{\|\boldsymbol{W}_0 + \Delta\boldsymbol{W}\|_c}\boldsymbol{x} = \underline{\boldsymbol{m}}\frac{\boldsymbol{W}_0 + \underline{\boldsymbol{B}\boldsymbol{A}}}{\|\boldsymbol{W}_0 + \underline{\boldsymbol{B}\boldsymbol{A}}\|_c}\boldsymbol{x}$, where $\|\cdot\|_c$ denotes the vector-wise norm of a matrix across each column

Can we achieve higher performance with significantly fewer trainable parameters compared to other PEFT methods?

Formulation of SVFT

Update weight matrices using a sparse combination of their singular vectors: $\boldsymbol{h} = \boldsymbol{W}_0 x + \Delta\boldsymbol{W}x = \boldsymbol{U}(\boldsymbol{\Sigma} + \underline{\boldsymbol{M}})\boldsymbol{V}^T\boldsymbol{x}$, where $\boldsymbol{U}$ and $\boldsymbol{V}$ are frozen, and $\underline{\boldsymbol{M}}$ is a $d_1 \times d_2$ sparse trainable matrix with pre-determined and fixed sparsity pattern.

SVFT leverages the structure and geometry of pre-trained weights to induce perturbations.

Four choices for Ω, the a-priori fixed sparsity pattern of $\underline{\boldsymbol{M}}$,

- **Plain** (SVFT^P) – constrain $\underline{\boldsymbol{M}}$ to be a diagonal matrix (most param-efficient)
- **Banded** (SVFT^B_d) – populate $\underline{\boldsymbol{M}}$ using a banded matrix, progressively making off-diagonals learnable
- **Random** (SVFT^R_d) – populate $\underline{\boldsymbol{M}}$ by randomly selecting k elements to be learnable
- **Top-k** (SVFT^T_d) – compute the alignment between left and right singular vectors as $\boldsymbol{u}_i^T\boldsymbol{v}_j$, and then select the top-k elements to be learnable

Properties of SVFT

a) **Structure**: If $\boldsymbol{M}$ is diagonal, then $\boldsymbol{W}_0 + \boldsymbol{U}\boldsymbol{M}\boldsymbol{V}^T$ has $\boldsymbol{U}$ as its left singular vectors and $\mathrm{sign}(\boldsymbol{\Sigma} + \boldsymbol{M})\boldsymbol{V}^T$ as its right singular vectors. If $\boldsymbol{M}$ is not diagonal, then $\boldsymbol{U}$ and $\boldsymbol{V}$ may no longer be the singular directions of the final matrix.

b) **Expressivity**: Given any target matrix $\boldsymbol{P}$ of size $d_1 \times d_2$, there exists an $\boldsymbol{M}$ such that $\boldsymbol{P} = \boldsymbol{W}_0 + \boldsymbol{U}\boldsymbol{M}\boldsymbol{V}^T$, i.e., if $\boldsymbol{M}$ is fully trainable, any target matrix can be realized.

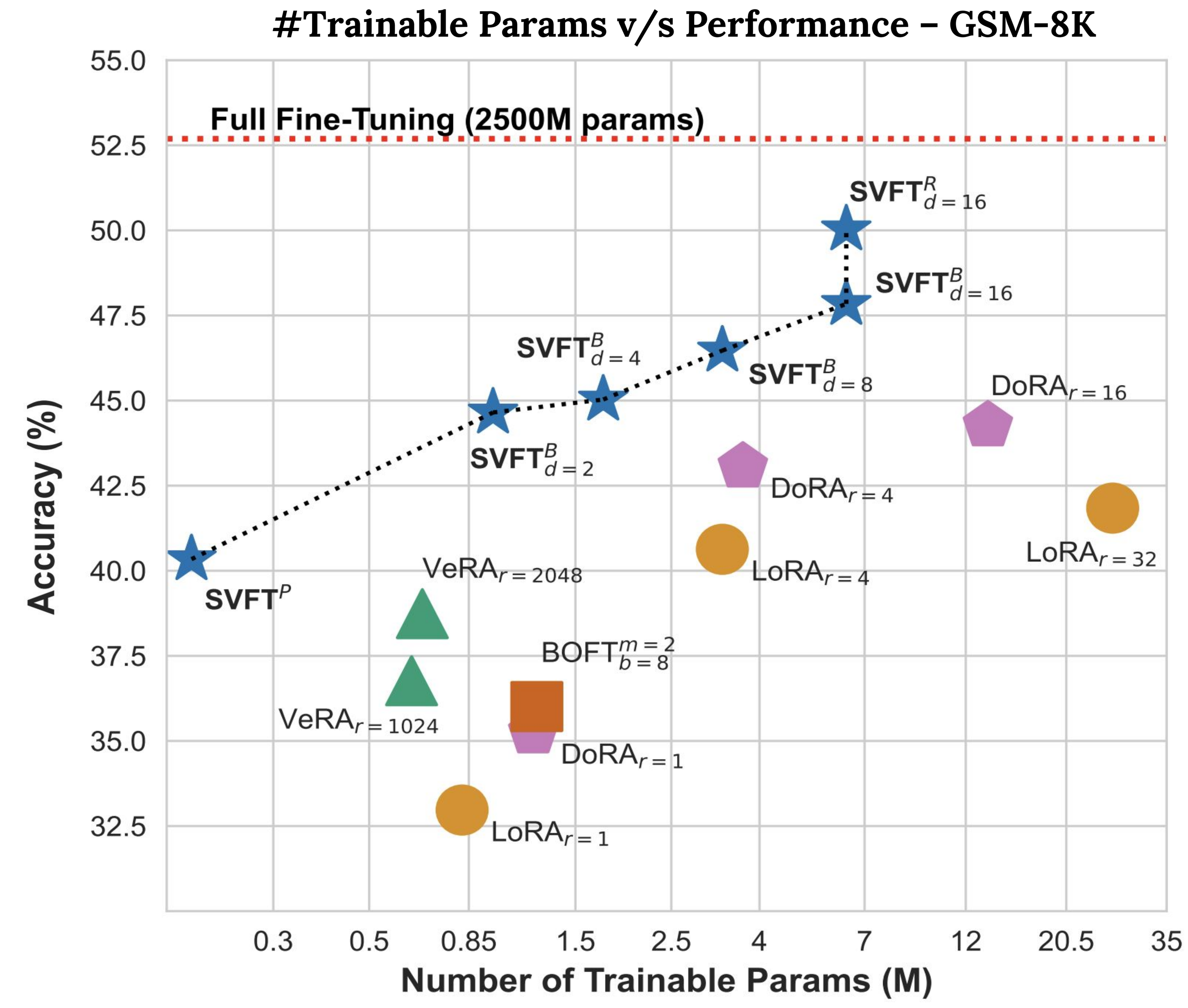
c) **Rank**: If $\boldsymbol{M}$ has k non-zero elements, then the rank of the update $\boldsymbol{U}\boldsymbol{M}\boldsymbol{V}^T$ is at most $\min\{k, \min\{d_1, d_2\}\}$. For the same number of trainable parameters, SVFT can produce a much higher rank perturbation than LoRA (eventually full rank), but in a constrained structured subspace.

Results on fine-tuning with SVFT using different M parameterizations

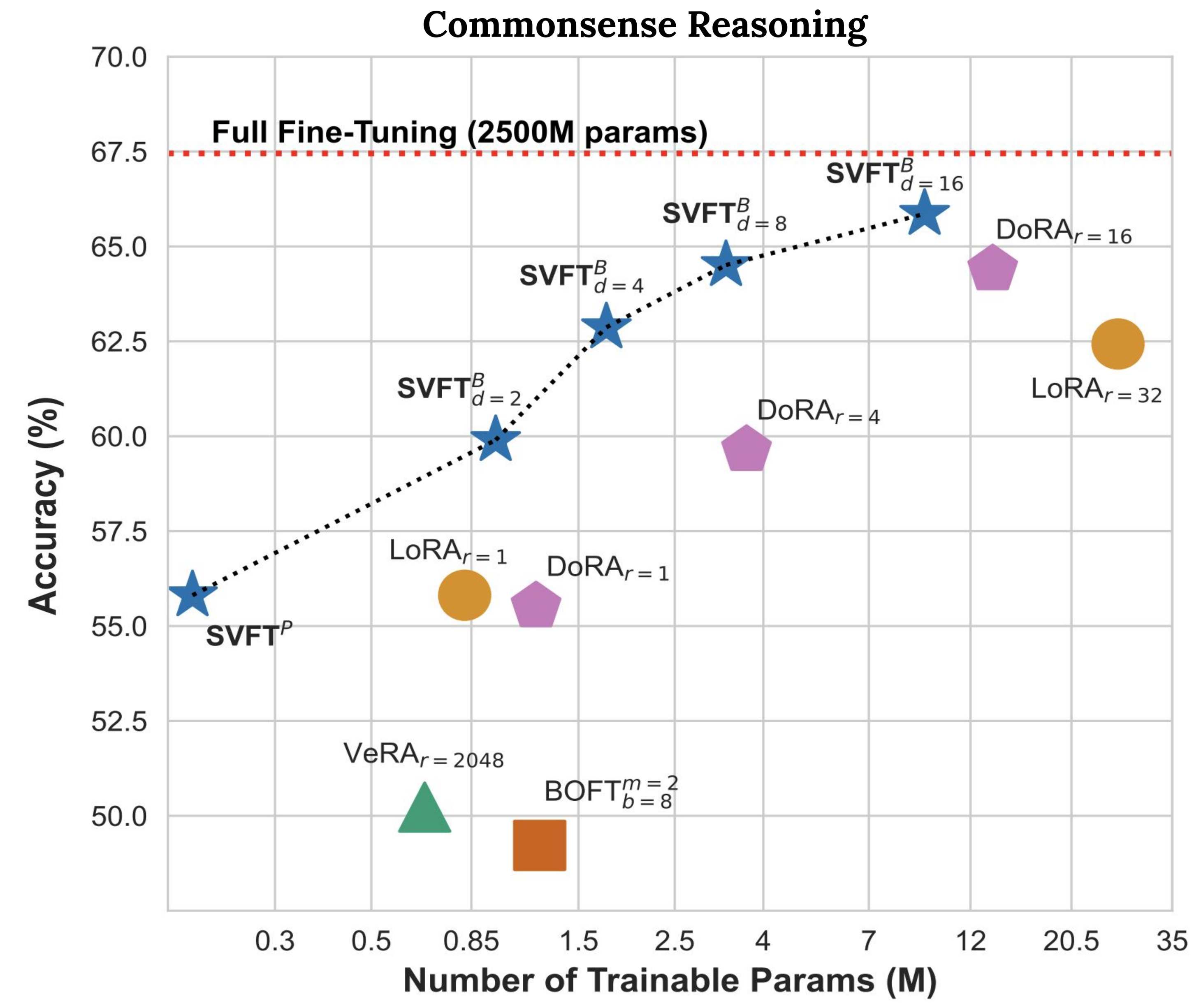
Structure	Gemma-2B			Gemma-7B			LLaMA-3-8B		
	#Params	GSM-8K	MATH	#Params	GSM-8K	MATH	#Params	GSM-8K	MATH
Plain	0.2M	40.34	14.38	0.43M	73.50	27.30	0.48M	69.22	20.44
Banded	6.4M	47.84	15.68	19.8M	76.81	29.98	17.2M	75.43	24.44
Random	6.4M	50.03	15.56	19.8M	76.35	29.86	17.2M	74.07	23.78
Top-k	6.4M	49.65	15.32	19.8M	76.34	29.72	17.2M	73.69	23.96

Experimental Results

#Trainable Params v/s Performance – GSM-8K



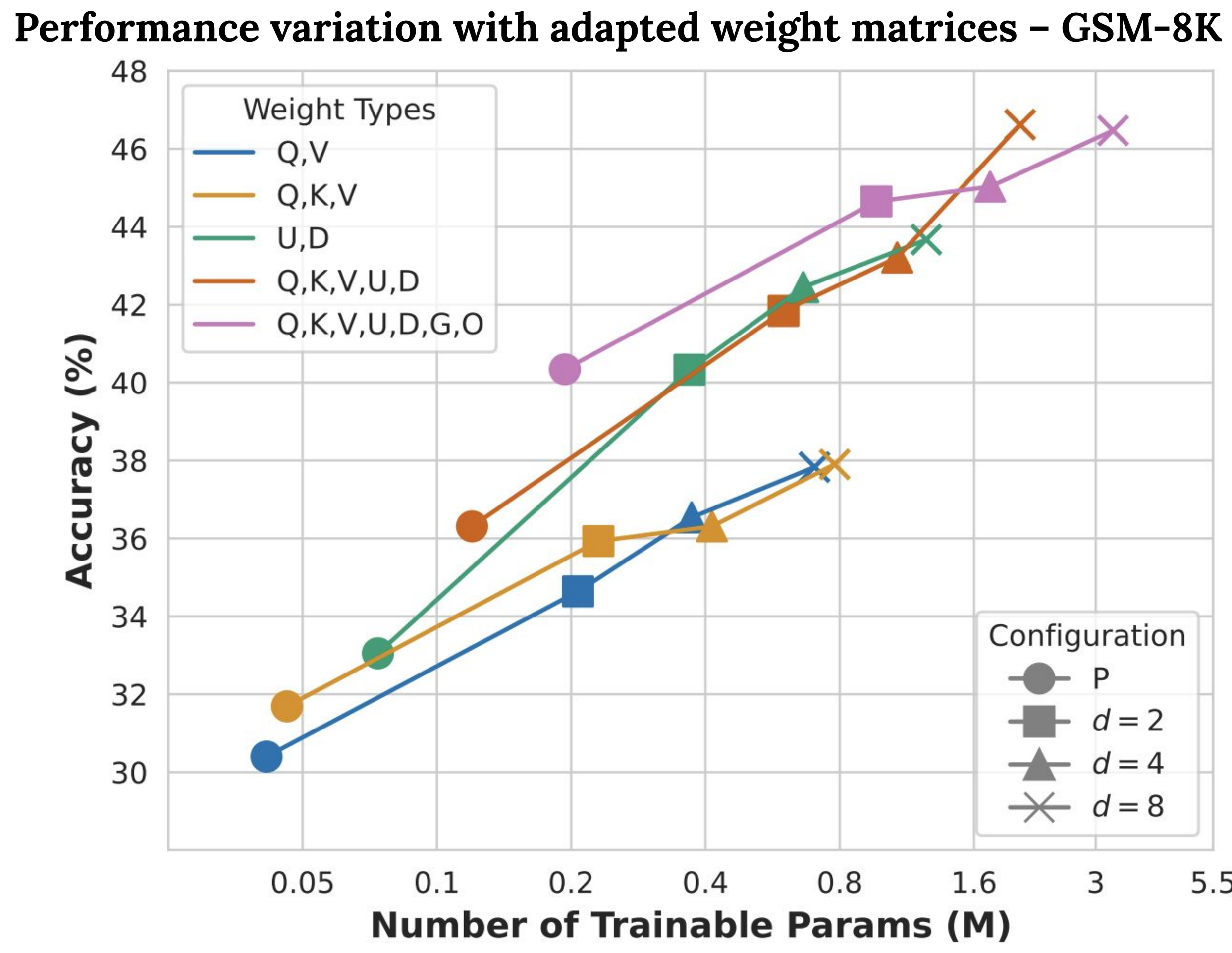
Commonsense Reasoning



Performance on Image Classification Tasks

Method	ViT-B			ViT-L		
	#Params	CIFAR100	Flowers102	#Params	Food101	Resisc45
Head	-	78.25	98.42	-	75.57	64.10
Full-FT	85.8M	85.35	98.37	303.3M	77.83	76.83
$\mathrm{LoRA}_{r=8}$	1.32M	84.10	<u>99.23</u>	3.54M	77.13	**79.62**
$\mathrm{DoRA}_{r=8}$	1.41M	85.03	**99.30**	3.76M	76.41	<u>78.32</u>
$\mathrm{BOFT}^{b=4}_{m=4}$	0.11M	<u>85.54</u>	98.59	2.95M	**78.42**	74.70
$\mathrm{LoRA}_{r=1}$	0.16M	84.86	96.88	0.44M	75.97	78.02
$\mathrm{DoRA}_{r=1}$	0.25M	84.46	99.15	0.66M	75.90	78.02
$\mathrm{VeRA}_{r=256}$	24.6K	83.38	98.59	0.06M	75.97	72.44
SVFT^P	18.5K	83.85	98.93	0.05M	75.95	71.97
$\mathrm{SVFT}^B_{d=2}$	0.27M	84.72	**99.28**	0.74M	<u>77.94</u>	**79.70**
$\mathrm{SVFT}^B_{d=8}$	0.93M	**85.69**	98.88	2.5M	**78.36**	73.83

Performance variation with adapted weight matrices – GSM-8K with Gemma-2B



Code

Paper

Workshop on Advancing Neural Network Training: Computational Efficiency, Scalability, and Resource Optimization (WANT@ICML 2024)