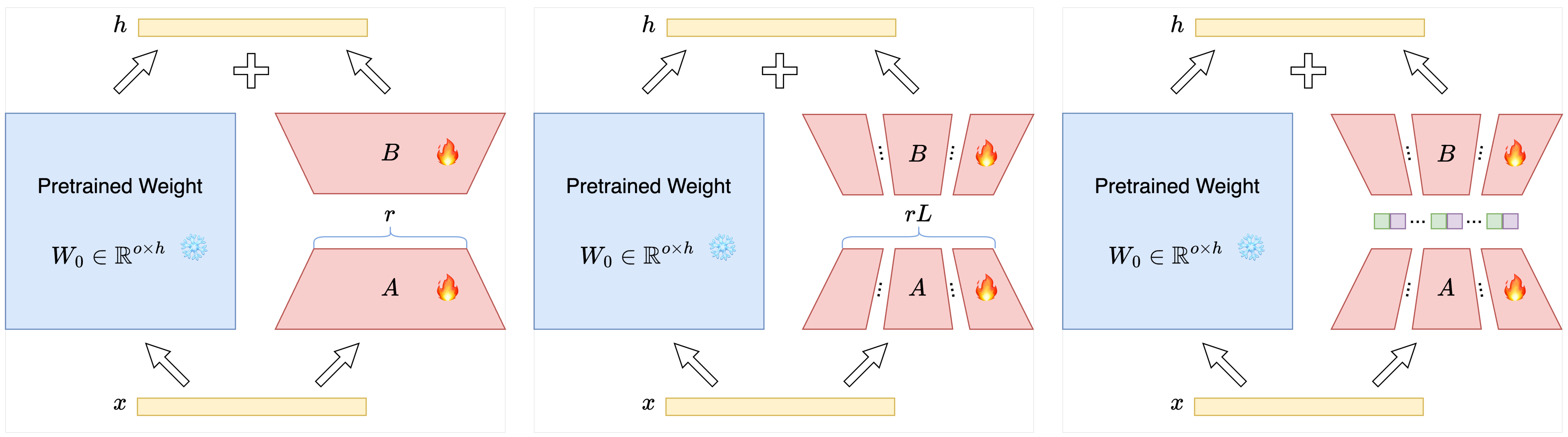
# MoS: Unleashing Parameter Efficiency of Low-Rank Adaptation with Mixture of Shards

Sheng Wang*, Liheng Chen*, Pengan Chen, Jingwei Dong, Boyang Xue, Jiyue Jiang, Lingpeng Kong, Chuan Wu

## Motivation




(a) LoRA (b) Pure Sharing (c) Pure Sharing (Differentiated)

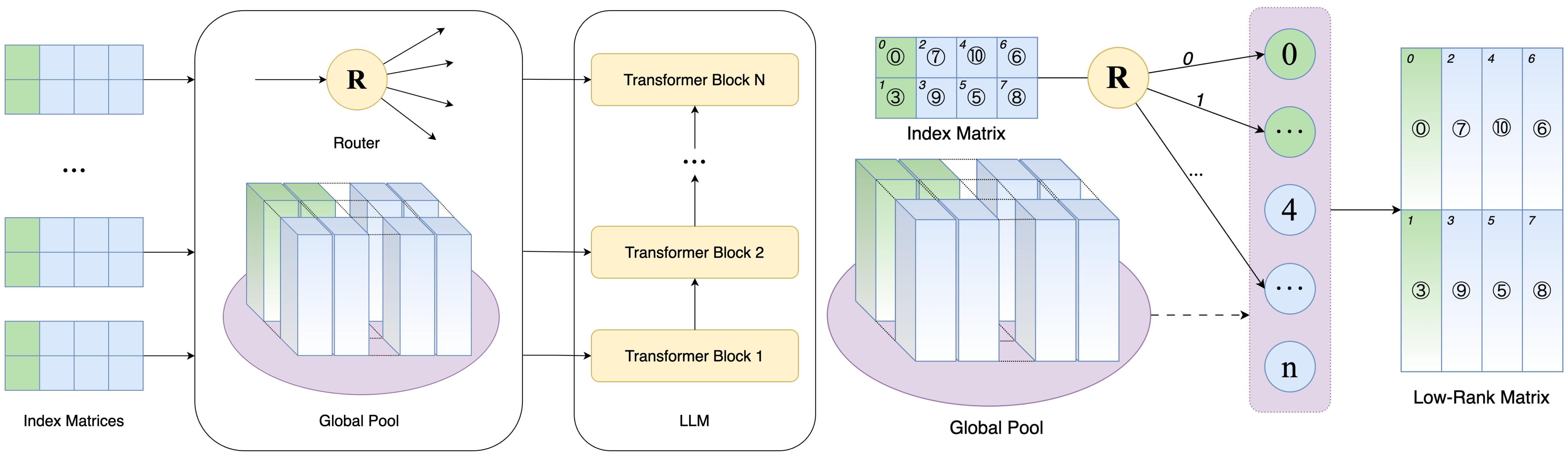
### Sharing & Differentiation

- Pure parameter sharing does not necessarily boost up the parameter efficiency of LoRA.
- Differentiation reverses the detrimental effects of pure sharing mechanism.

### Key Contribution

- Guided by the above high-level sharing insights, we introduce a more ($8\times$) parameter-efficient finetuning method named **M**ixture **o**f **S**hards (MoS), which incorporates both inter-layer and intra-layer sharing schemes, facilitating the concurrent serving of numerous customized models with saliently reduced GPU memory overhead.

## Method




(a) Inter-Layer Sharing (b) Intra-Layer Sharing

### Global Sharing ◁ Sharing

$$\Delta\mathbf{W} = \mathbf{B}\mathbf{A} = \mathbf{B}^p\mathbf{A}^p$$

- A globally shared pool for each linear layer type consists of multiple independently initialized and trained vector pairs, while all vector pairs within the low-rank matrix pairs (*i.e.*, $\mathbf{A}$ and $\mathbf{B}$) for each layer are sampled from the pool.

### Subset Selection ◁ Differentiation

$$\Delta\mathbf{W}^k = \mathbf{B}^k\mathbf{A}^k = \mathbf{B}^p\Lambda^k\mathbf{A}^p = \sum_{i=1}^{eL} m_i^k \cdot \mathbf{b}_i^p \otimes \mathbf{a}_i^p$$

- Select a specific number of vector pairs from the shared matrices (*i.e.*, global pools).

### Pair Dissociation & Vector Sharding ◁ Differentiation

$$\Delta\mathbf{W}^k = \mathbf{B}^k\mathbf{A}^k = \mathrm{Route}^c(\mathbf{B}^p, \mathbf{I}_b^k)\,\mathrm{Route}^r(\mathbf{A}^p, \mathbf{I}_a^k)$$

- Decouple vector pairs into two pools, crop each vector into smaller shards, and sample them separately.

### Shard Privatization ◁ Differentiation

$$\Delta\mathbf{W}^k = \mathbf{B}^k\mathbf{A}^k = \mathrm{Route}^c(\mathrm{Concat}(\mathbf{A}^{pub}, \mathbf{A}^{pri}), \mathbf{I}_b^k)\,\mathrm{Route}^r(\mathrm{Concat}(\mathbf{B}^{pub}, \mathbf{B}^{pri}), \mathbf{I}_a^k)$$

- Partition each global pool into two segments: a public segment that remains shared, and a private segment exclusively accessible to one matrix.

## Experiments

### Motivation

| Method | Rank | # Param. | MMLU | BBH | GSM8K | TyDi QA | | HumanEval | Avg. |
|---|---|---|---|---|---|---|---|---|---|
| | | | EM | EM | EM | F1 | EM | P@1 | |
| LoRA | 2 | 5.00M | 44.77 | 36.22 | 26.28 | 48.67 | 35.70 | 18.24 | 34.98 |
| Pure Sharing | 64 | 5.00M | 43.61 | 35.15 | 26.54 | 49.12 | 36.04 | 15.53 | 34.33 |
| + Random Scaling | 64 | 5.00M | 43.98 | 35.30 | 29.04 | 49.03 | 35.73 | 15.58 | 34.77 |
| + Subset Selection | 64 | 5.00M | 45.56 | 36.76 | 28.18 | 50.33 | 37.22 | 18.64 | 36.12 |

Table 1: Results of LLaMA2-7B with different sharing and differentiation methods across diverse instruction following datasets. "+ Random Scaling" and "+ Subset Selection" denote the individual integration of them into the "Pure Sharing" scheme.

### Main results

| Method | Rank | # Param. | MMLU (factuality) | BBH (reasoning) | GSM8K (reasoning) | TyDi QA (multilinguality) | | HumanEval (coding) | Avg. |
|---|---|---|---|---|---|---|---|---|---|
| | | | EM (0-shot) | EM (3-shot) | EM (8-shot, CoT) | F1 (1-shot, GP) | EM (1-shot, GP) | P@1 (0-shot) | |
| Vanilla (chat)† | - | - | 41.18 | 0.00 | 3.03 | 17.40 | 0.10 | 0.64 | 10.39 |
| Vanilla (no-chat)† | - | - | 41.53 | 33.43 | 15.47 | 49.18 | 35.35 | 13.57 | 31.42 |
| LoRA | 2† | 5.00M | 44.77 | 36.22 | 26.28 | 48.67 | 35.70 | 18.24 | 34.98 |
| | 8† | 19.99M | 46.55 | 36.92 | 31.11 | 50.50 | 36.89 | 19.37 | 36.89 |
| | 16† | 39.98M | 46.70 | 36.43 | 31.34 | 50.97 | 37.64 | 18.73 | 36.97 |
| | 64 | 159.91M | **47.10** | 37.78 | **31.43** | 51.65 | 38.07 | 19.12 | 37.53 |
| VeRA† | 256 | 1.42M | 42.51 | 35.10 | 22.69 | 48.39 | 36.38 | 18.90 | 34.00 |
| Tied LoRA† | 280 | 4.99M | 44.36 | 35.76 | 25.47 | 50.16 | 37.15 | 18.68 | 35.26 |
| PRoLoRA† | 4/8 | 5.00M | 45.85* | 36.45* | 27.57 | 49.94* | 36.59* | 19.75* | 36.03 |
| MoS | 4/8 | 5.00M | <u>46.09</u> | <u>37.29</u> | <u>28.43</u> | <u>50.21*</u> | <u>37.19*</u> | <u>19.12</u> | <u>36.39</u> |
| | 16/32 | 19.99M | 47.01* | **37.79** | 30.93 | **51.71** | **38.34** | **20.00***  | **37.63** |
| MoS-sp | 16/32 | 19.99M | 46.64* | 36.69 | 30.17 | 50.27 | 36.90 | 18.60* | 36.54 |
| MoS-vs | 16/32 | 19.99M | 46.47* | 37.52 | 31.77 | 50.90 | 37.98 | 18.69* | 37.22 |
| MoS-pd | 16/32 | 19.99M | 46.23* | 36.17 | 30.71 | 51.40 | 37.94 | 16.77* | 36.54 |

Table 2: Results of LLaMA2-7B across multiple instruction-following datasets using different methods. The symbols "-sp", "-vs", and "-pd" indicate the ablation of shard privatization, vector sharding, and pair dissociation. "$*$" denotes the optional higher ranks. Underlined values indicate the best performance with 5.00M trainable parameters, while bold values denote the best results across all configurations.

### Hyperparameter Robustness.

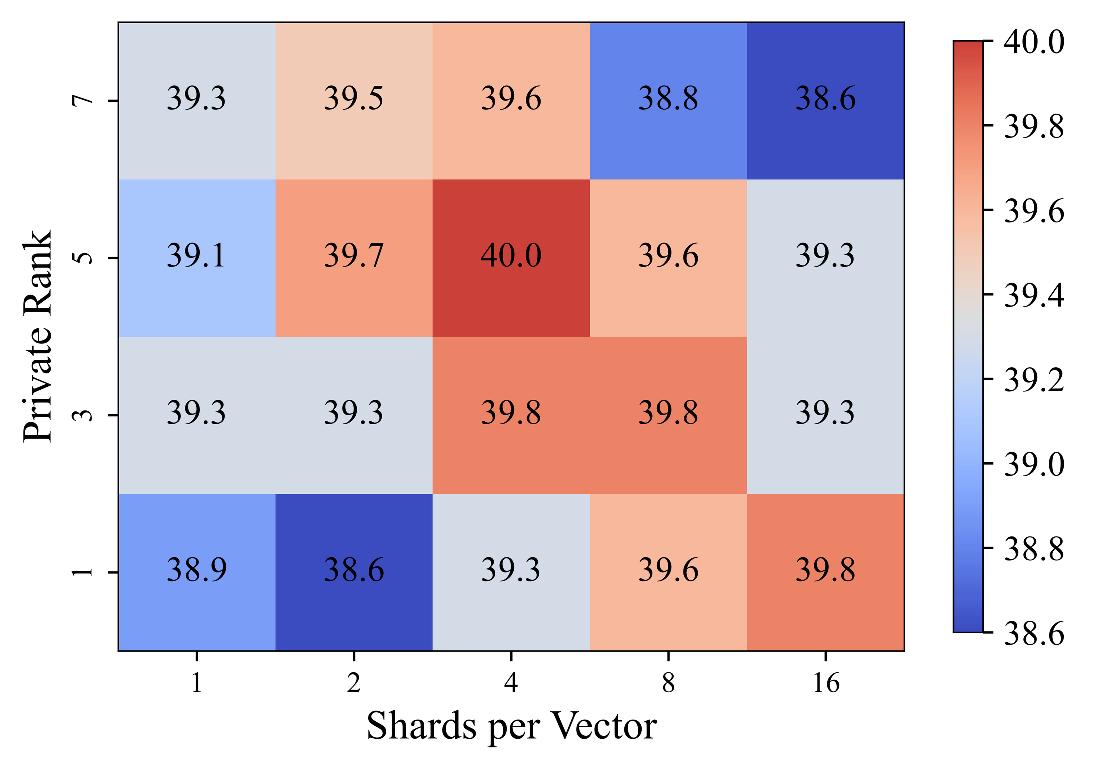



Figure 1: Performance of MoS with the rank of 8 with respect to private ranks and shards per vector given a specific parameter budget on the LLaMA3.2-3B model and BBH benchmark.

- Specific Parameter Budget. MoS achieves higher parameter efficiency than LoRA, while keeping better practical feasibility than other baselines in a capacity constrained scenario (*i.e.*, 5M parameter budget).
- Specific Performance Target. MoS achieves 4/6 performance targets and an average improvement from 36.97 to 37.63, with 1/8 of trainable parameters, indicating seven times more tasks/users concurrently.
- Hyperparameter Robustness. For any given private rank, there always exists a suitable range of shard numbers that consistently produce remarkable results (*i.e.*, $\geq 39.8\%$).
- Ablation Study. The ablation studies on pair dissociation, vector sharding, and shard privatization (*i.e.*, MoS-pd, MoS-vs, and MoS-sp) demonstrate consistently inferior performance, highlighting the significant, nearly cost-free improvements brought by these schemes, respectively.