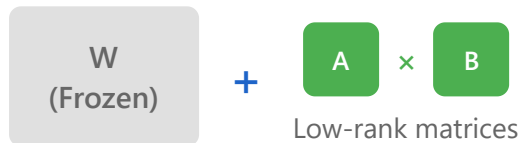


Parameter-Efficient Fine-tuning (PEFT)



LoRA

Low-Rank Adaptation



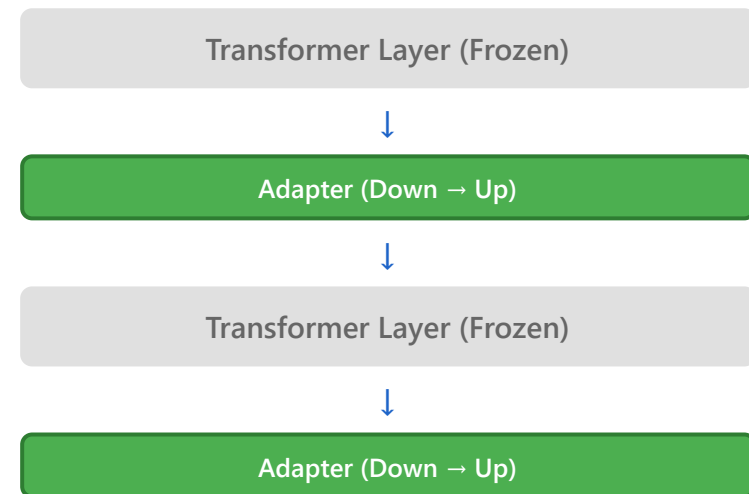
$$W' = W + AB$$

<1% **parameters** trainable



Adapter Layers

Bottleneck Modules



Small modules between layers



Prefix Tuning

Optimize **continuous prompt vectors**



Prompt Tuning

Learn **soft prompts**, freeze model



BitFit

Only tune **bias terms**, freeze weights



Single GPU Training

Fine-tune **large models** on limited hardware

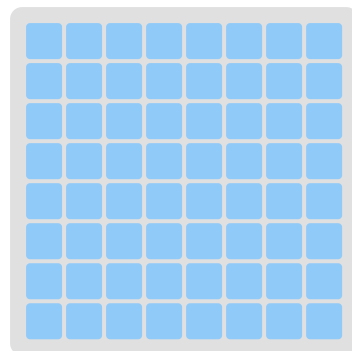


High Efficiency

90-95% of full fine-tuning performance



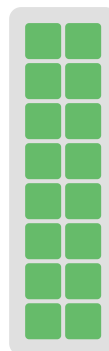
Practical Example: LoRA Parameter Decomposition



W (Frozen)

$8 \times 8 = 64$ parameters

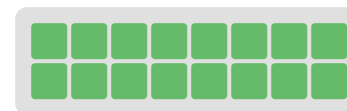
\approx



A (Trainable)

$8 \times 2 = 16$ params

\times



B (Trainable)

$2 \times 8 = 16$ params

64

Original Parameters

32

LoRA Parameters
(16 + 16)

50%

Reduction

With Rank $r = 2$: $(m \times r) + (r \times n) = (8 \times 2) + (2 \times 8) = 32$

Lower rank = fewer parameters

($r=1$: 16 params, $r=2$: 32 params, $r=4$: 64 params)



In large-scale models, over 99% parameter reduction is possible!