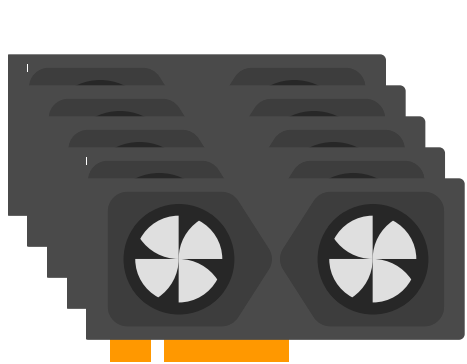


What are **Small Language Models (SLMs)**?



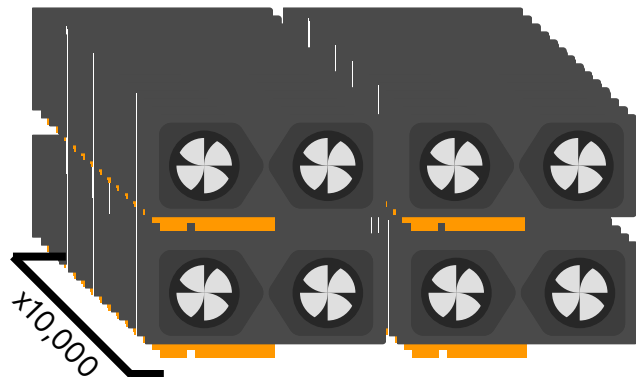
Cost: \$10,000-500,000



Small Cluster



Cost: \$10M - \$500B

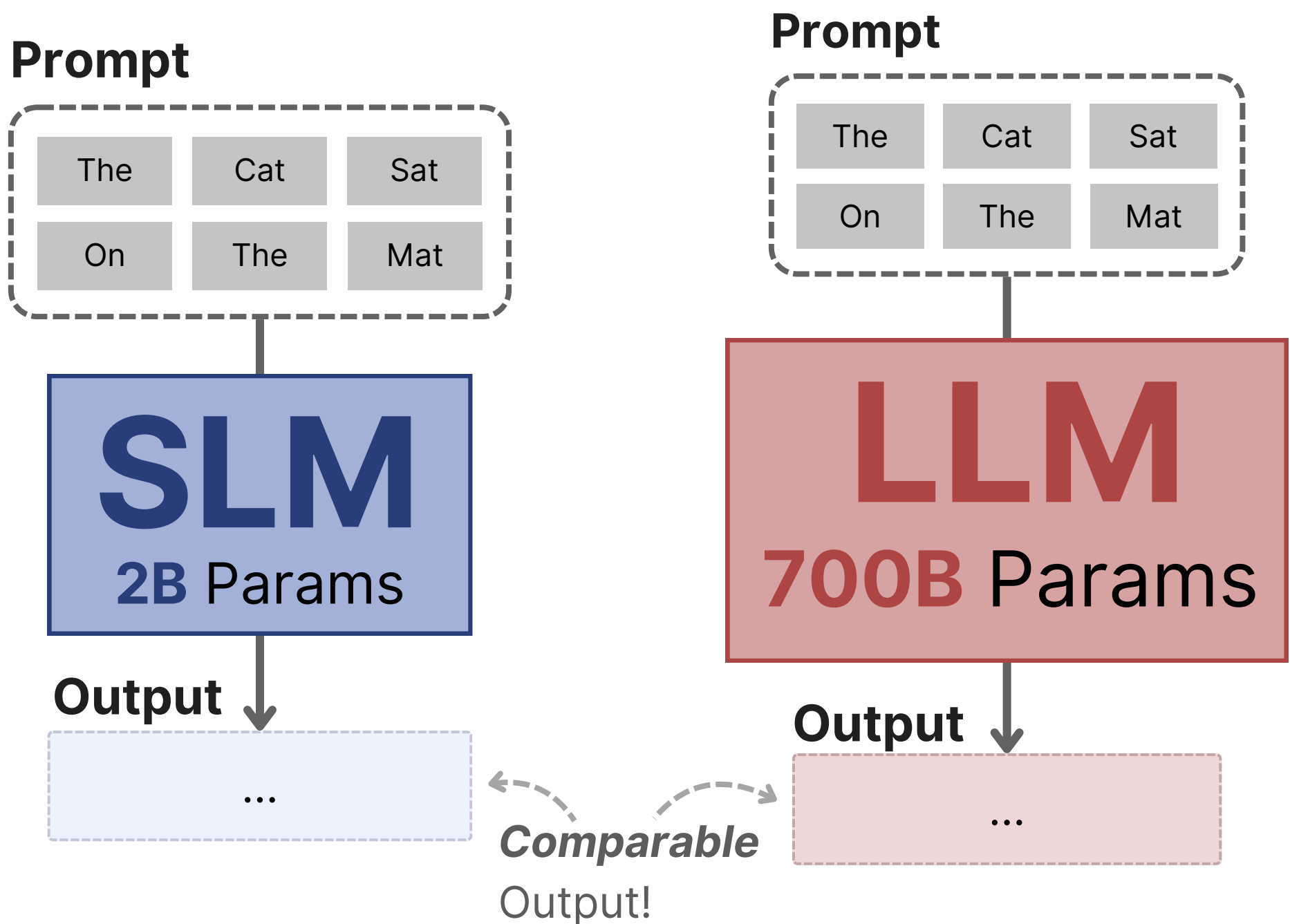


Large Cluster

Swipe For More →

SLMs are LLMs That are **Much Smaller** in Size

(< 10B parameters)



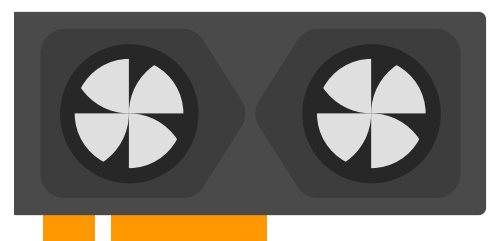
Although small, SLMs show similar performance on task-specific use cases

SLMs are **Particularly Useful**



Cost: \$10,000-500,000

Locally Hostable



Benefits:

- Locally Hostable
- Runs on Consumer-Grade Hardware

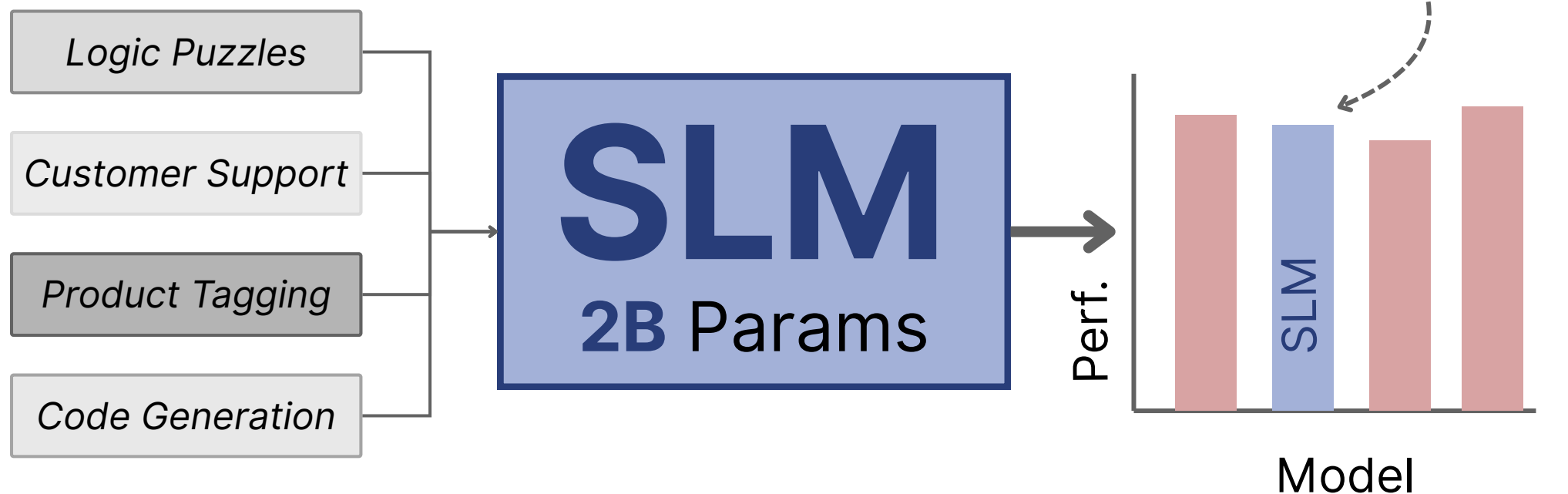
NVIDIA deemed SLMs were the **future of agentic AI**

But why not LLMs?

LLMs Are Good For Broad Knowledge

Why LLMs = **hundreds of billions** of parameters

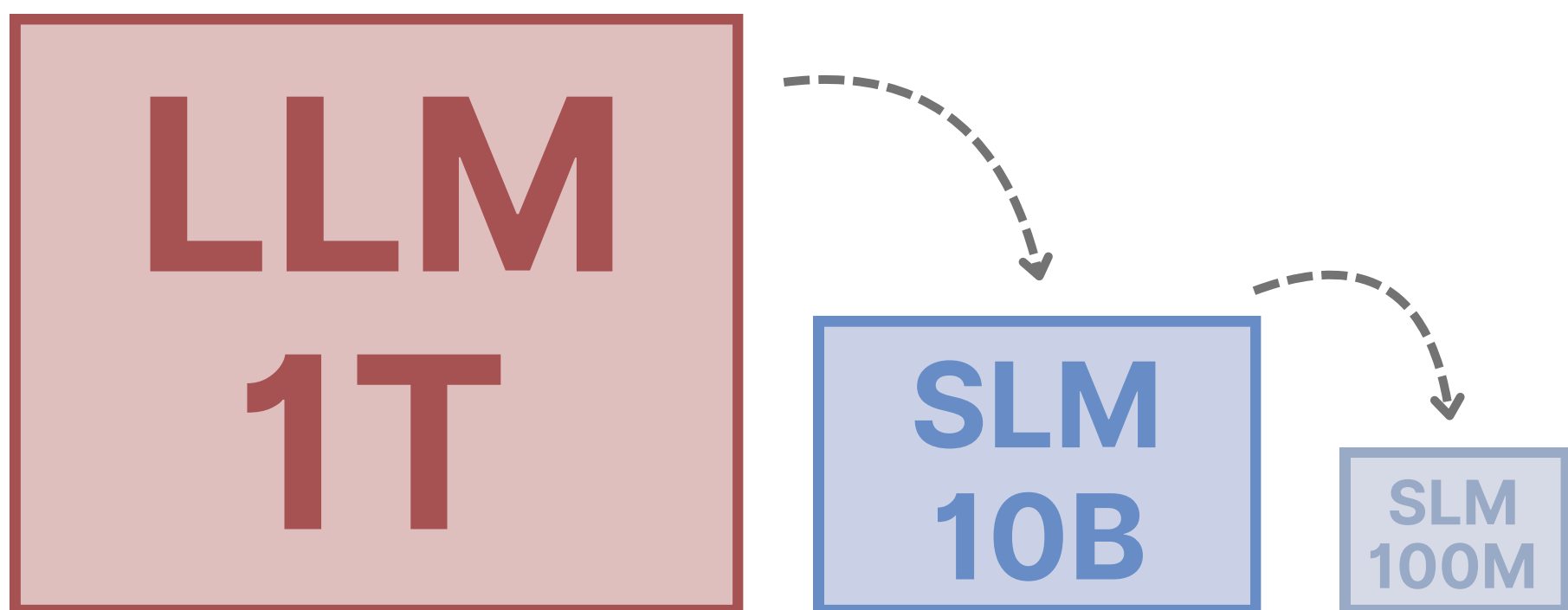
Domain Specific Tasks



SLMs are good for **specializing**
in specific tasks (e.g. *coding*)

How Did SLMs Get So Small?

They use **state-of-the-art** optimization techniques



Examples:

- Quantization
- LoRA Fine-Tuning
- FlashAttention

Quantization: Reduce Precision of Parameters

High-Precision (e.g., FP32)

Quantized (e.g., INT8)

Scalar

3.14

Vector

$$\begin{bmatrix} 3.14 \\ 2.72 \end{bmatrix}$$

Scalar

3

Vector

$$\begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

Matrix

$$\begin{bmatrix} 3.14 & 2.72 \\ 1.62 & 1.41 \end{bmatrix}$$

Tensor

$$\begin{bmatrix} \begin{bmatrix} 3.14 & 2.72 \end{bmatrix} & \begin{bmatrix} 4.5 & 6.2 \end{bmatrix} \\ \begin{bmatrix} 1.62 & 1.41 \end{bmatrix} & \begin{bmatrix} 8.8 & 7.1 \end{bmatrix} \end{bmatrix}$$

Matrix

$$\begin{bmatrix} 3 & 2 \\ 1 & 1 \end{bmatrix}$$

Tensor

$$\begin{bmatrix} \begin{bmatrix} 3 & 2 \end{bmatrix} & \begin{bmatrix} 4 & 6 \end{bmatrix} \\ \begin{bmatrix} 1 & 1 \end{bmatrix} & \begin{bmatrix} 8 & 7 \end{bmatrix} \end{bmatrix}$$

GPU

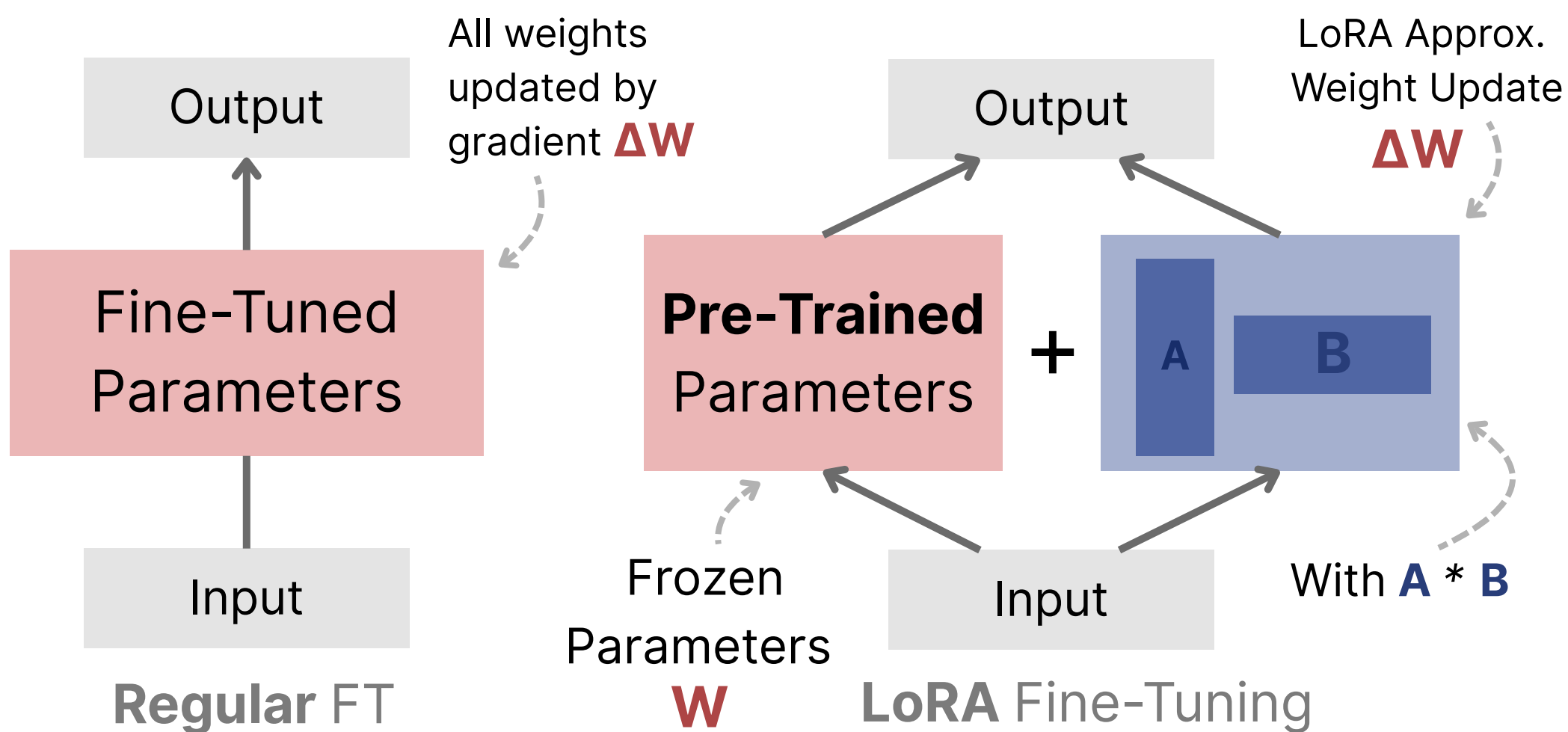
Stored in GPU as **parameters**

[[[0, 0.1, 0.2],[0.2, 0.4, 0.6]], [[-0.4, -0.3, 0.6], [0.7, 0.4, 0.3]]]
[[[0.5, 0.4, 0.2],[0.2, 0.9, 0.5]], [[-0.1, -0.3, 0.5], [0.6, -0.1, 0]]]
[[[0, 0.1, 0.2],[0.2, 0.4, 0.6]], [[-0.1, -0.3, 0.6], [0.7, 0.1, 0.1]]]

Quantization trades **model accuracy** for a smaller size

- Quantizing FP16 → INT4
└→ **3x** size reduction

Fine Tuning: Low Rank Adaptation (LoRA)



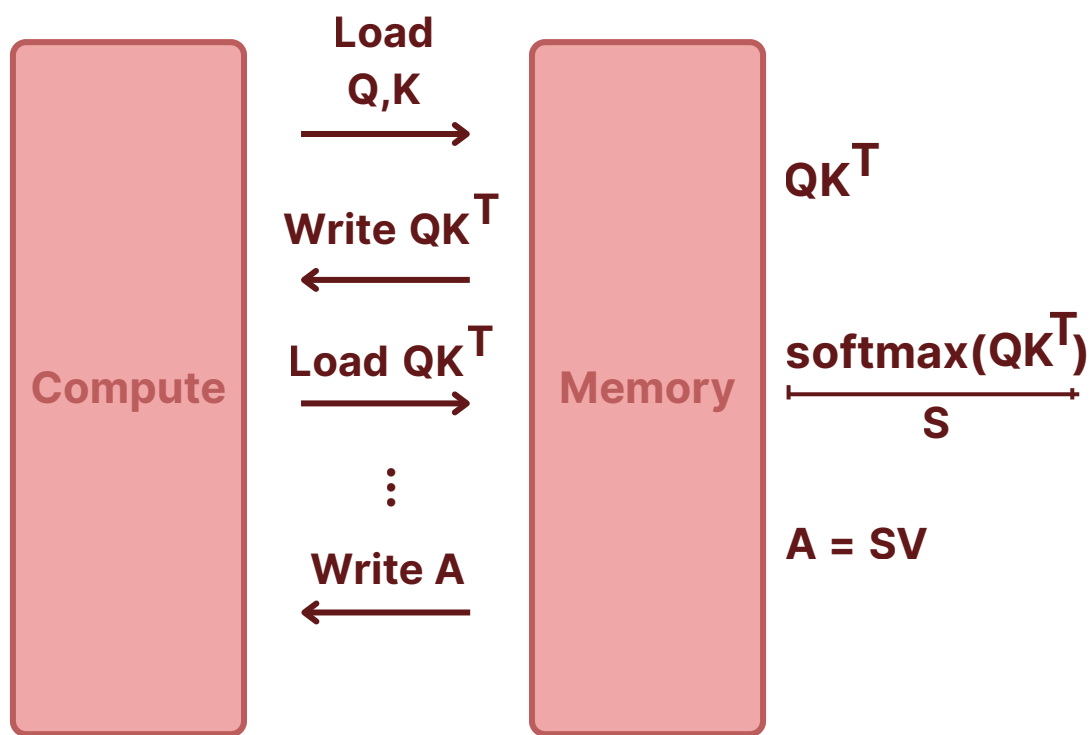
Idea: Add trainable **low-rank matrices** to weights

- Freeze most parameters

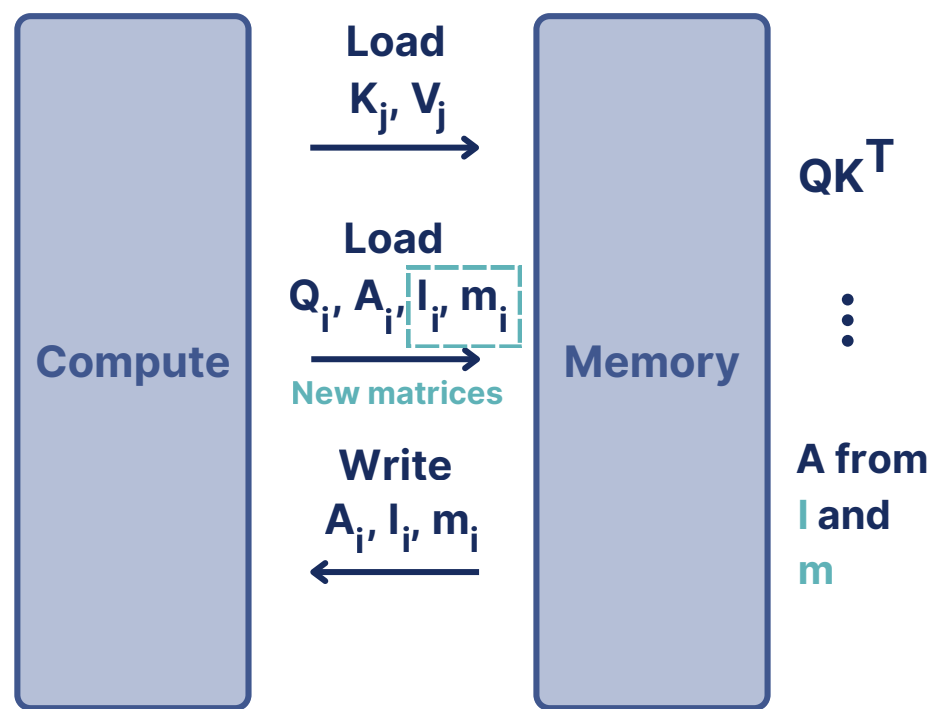
Fraction of parameters used for transfer learning

Optimizing Attention: FlashAttention

Normal Attention



FlashAttention



Fused GPU Kernel:

- Computes efficiently and in parallel

Speeds up attention and
reduces memory overhead

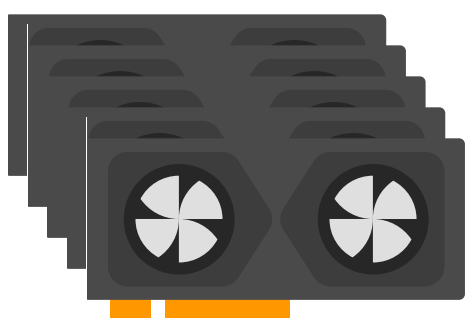
What Was The **Result**?

<10B Parameter Models That are:

- Fast and ***Cheap***
- Memory-Efficient
- **Performant** on task-specific applications



Cost: \$10,000-500,000



Cost: \$10M - \$500B



Like this **Post?**

Accelerate Your Learning With Our **Weekly AI Newsletter**

- Learn Industry Knowledge
Without The Headache

Link in Bio 

Weekly AI
Insights, **Visually**
Explained

Small Language Models (The Future?)

AI, But Simple Issue #68

In recent times, the popularity of transformer-based LLMs and LLM applications such as AI agents has skyrocketed. **Compute is in high demand**, while models soar in parameter count—reaching hundreds of billions and trillions of parameters in the largest LLMs.

If you wanted to run your own **language** model, this would be impossible on consumer-grade hardware. Even **small**-to-medium sized companies don't have the budget to train a model of this size.

Luckily, researchers have been moving towards **quantization** and other techniques to **reduce the compute and VRAM** needed to store, train, and run models. One of the recent methods in research is to reduce the physical size (in parameters) of the models themselves.

This is where **small language models (SLMs)** come in. **Small language** models are neural **language** models that are much **smaller** in size (typically billions of parameters or fewer) than today's massive LLMs (which often have hundreds of billions).

Small Language Models (SLMs) **Large** Language Models (LLMs)