Efficient Fine-Tuning of LLMs

with Low-Rank Adaption (LoRA)

Presented by

Greg Loughnane, Founder & CEO
Chris Alexiuk, Co-Founder & CTO

🎯 ALIGNING OUR AIM

lide

AI MAKERSPACE

BY THE END OF TODAY...

- What is **fine-tuning**, **PEFT**, and **LoRA**

- **How to fine-tune LLMs** with open-source tools from **Hugging Face**

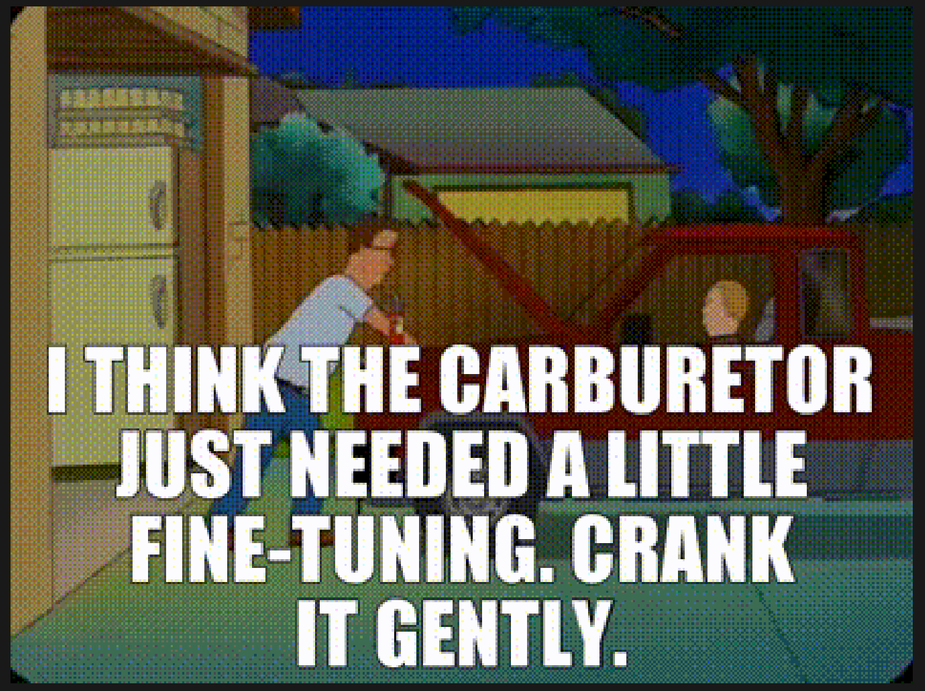
OVERVIEW

- 💪 Prototyping
- ⚖️ Fine-Tuning
- ⏬ Parameter-Efficient Fine-Tuning (PEFT)
- 🎖️ Low-Rank Adaption (LoRA)
- 🤦 FT with Hugging Face
- ❓ Conclusions, QA

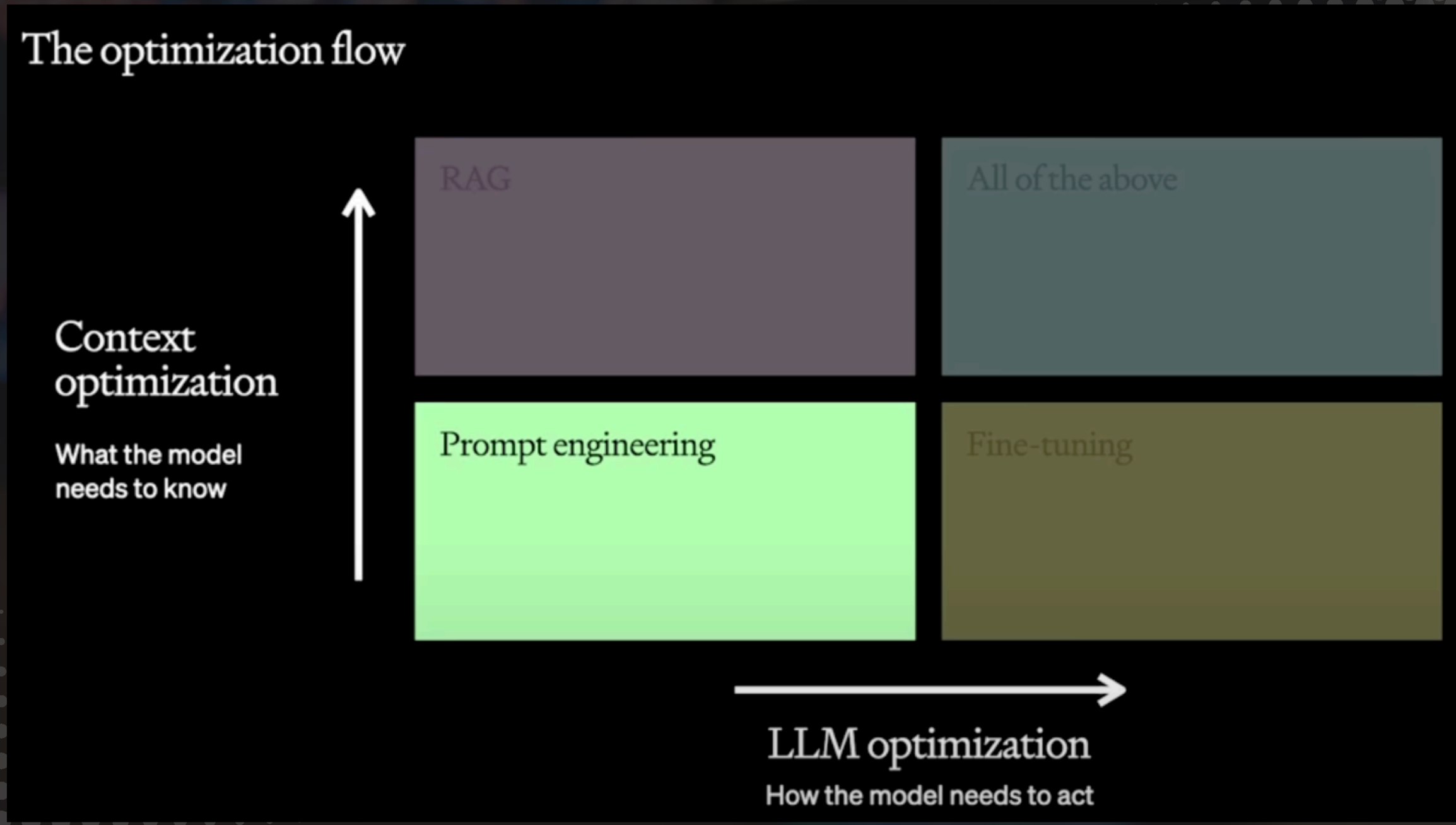
💪 PROTOTYPING

PROTOTYPING LLM APPS

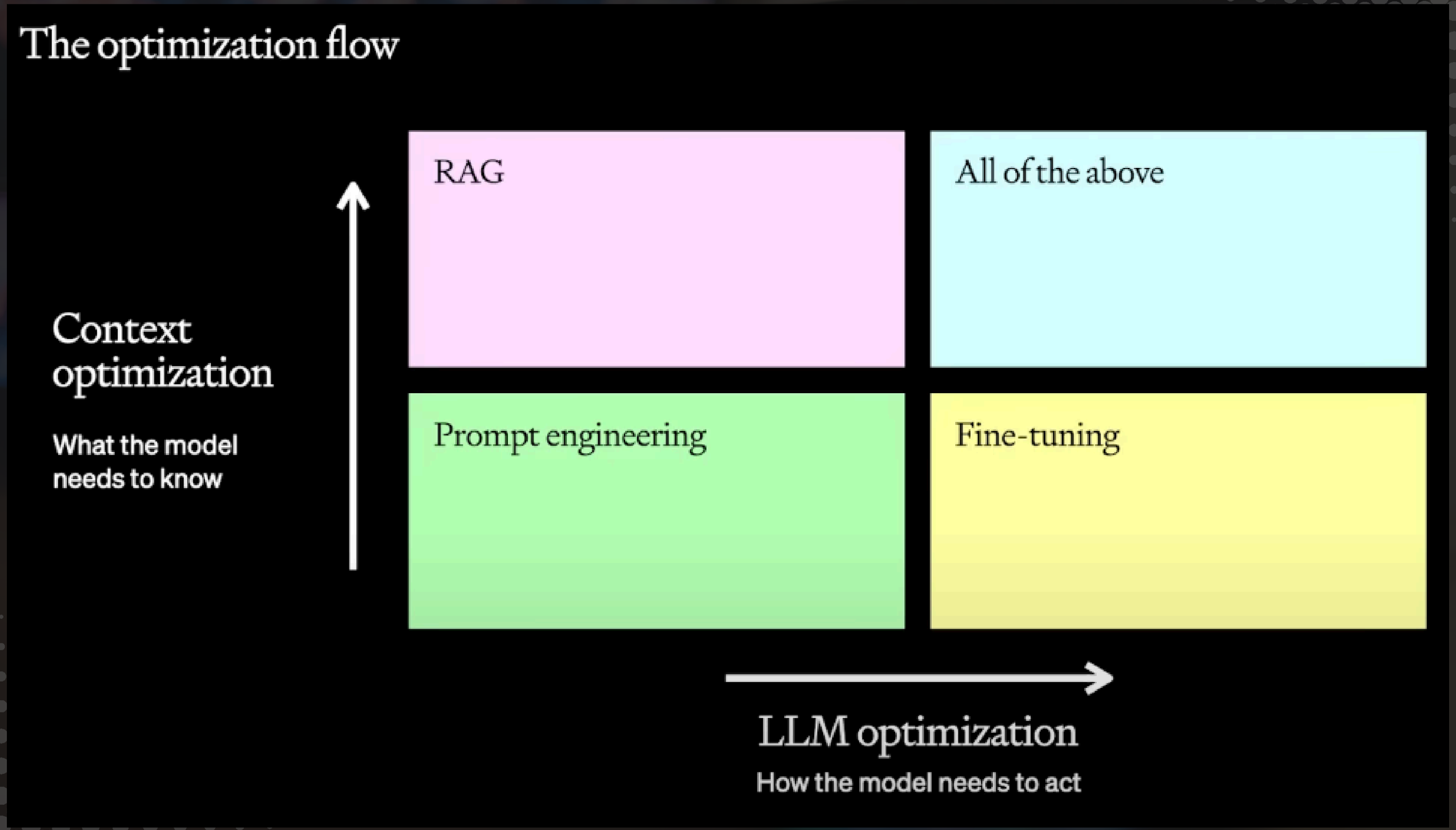
1. Prompt Engineering
2. Question Answering Systems
3. Fine-Tuning Models



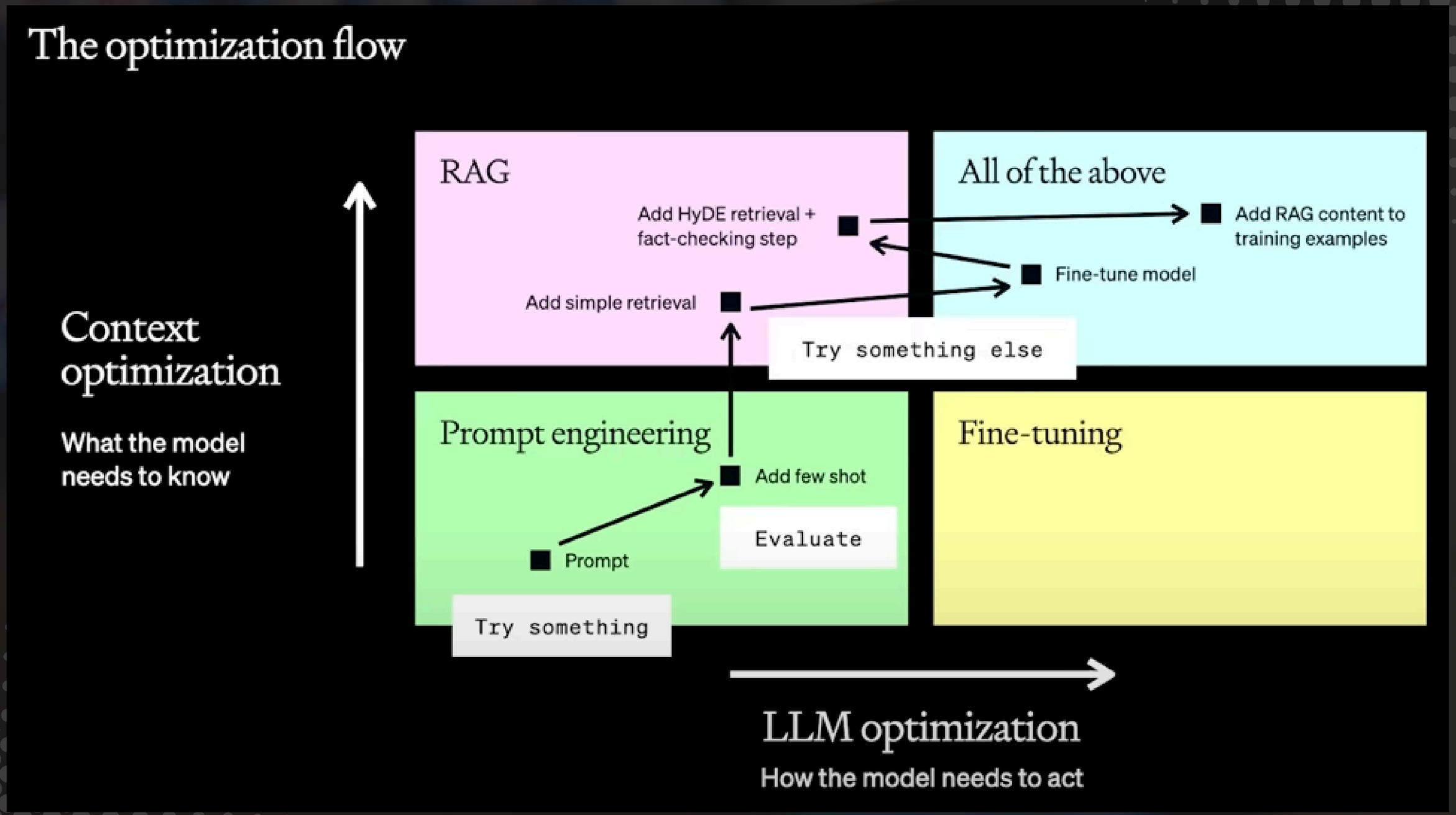
The optimization flow



The optimization flow



The optimization flow



Small Language Models (SLMs)

Efficiency, Transparency , Accuracy, Security

TinyStories: How Small Can Language Models Be and Still Speak...

Language models (LMs) are powerful tools for natural language processing, but they often struggle to produce coherent and fluent text when they are small. Models with around 125M parameters such...

arXiv.org

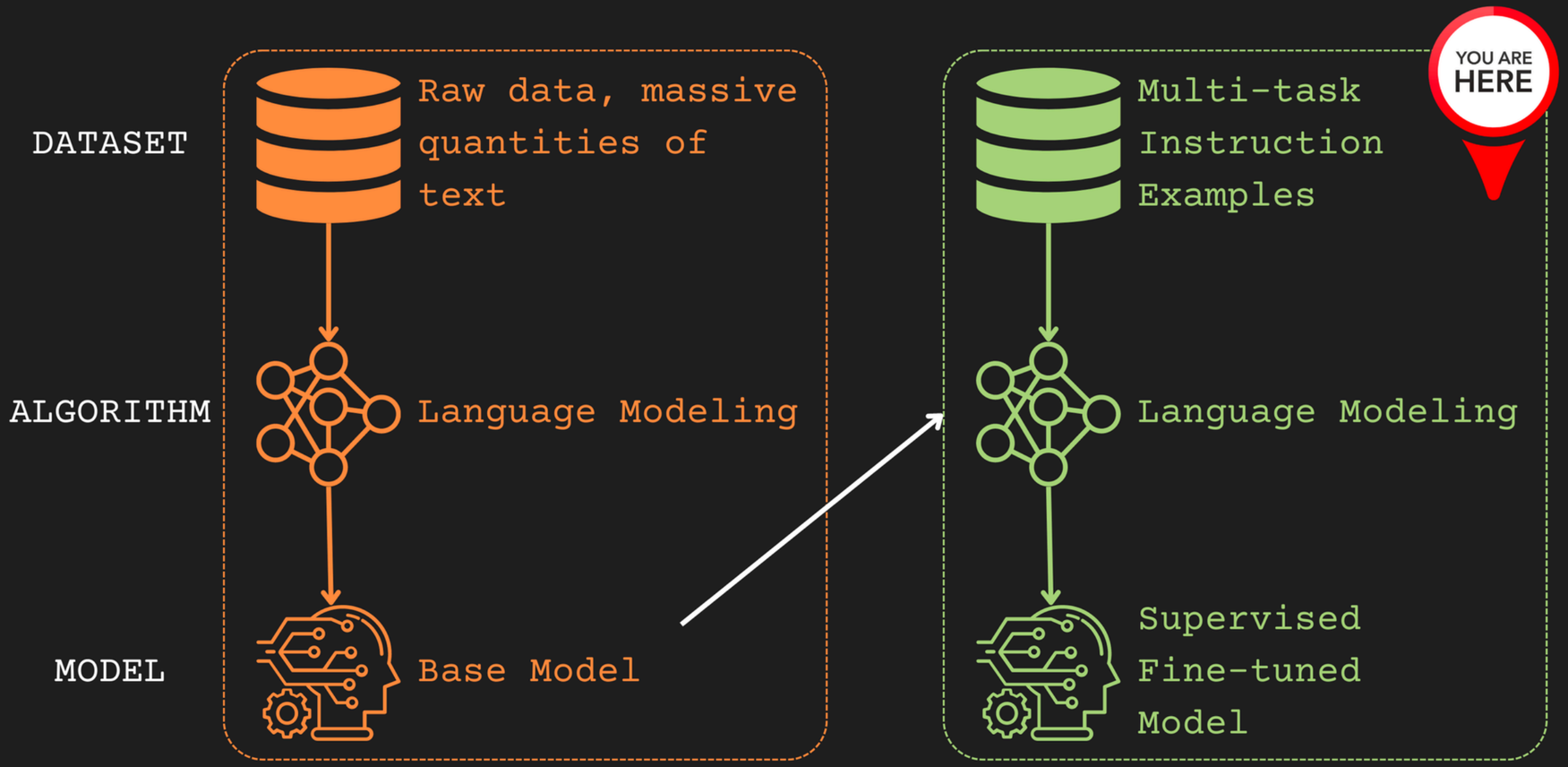
Textbooks Are All You Need

We introduce phi-1, a new large language model for code, with significantly smaller size than competing models: phi-1 is a Transformer-based model with 1.3B parameters, trained for 4 days on 8...

arXiv.org

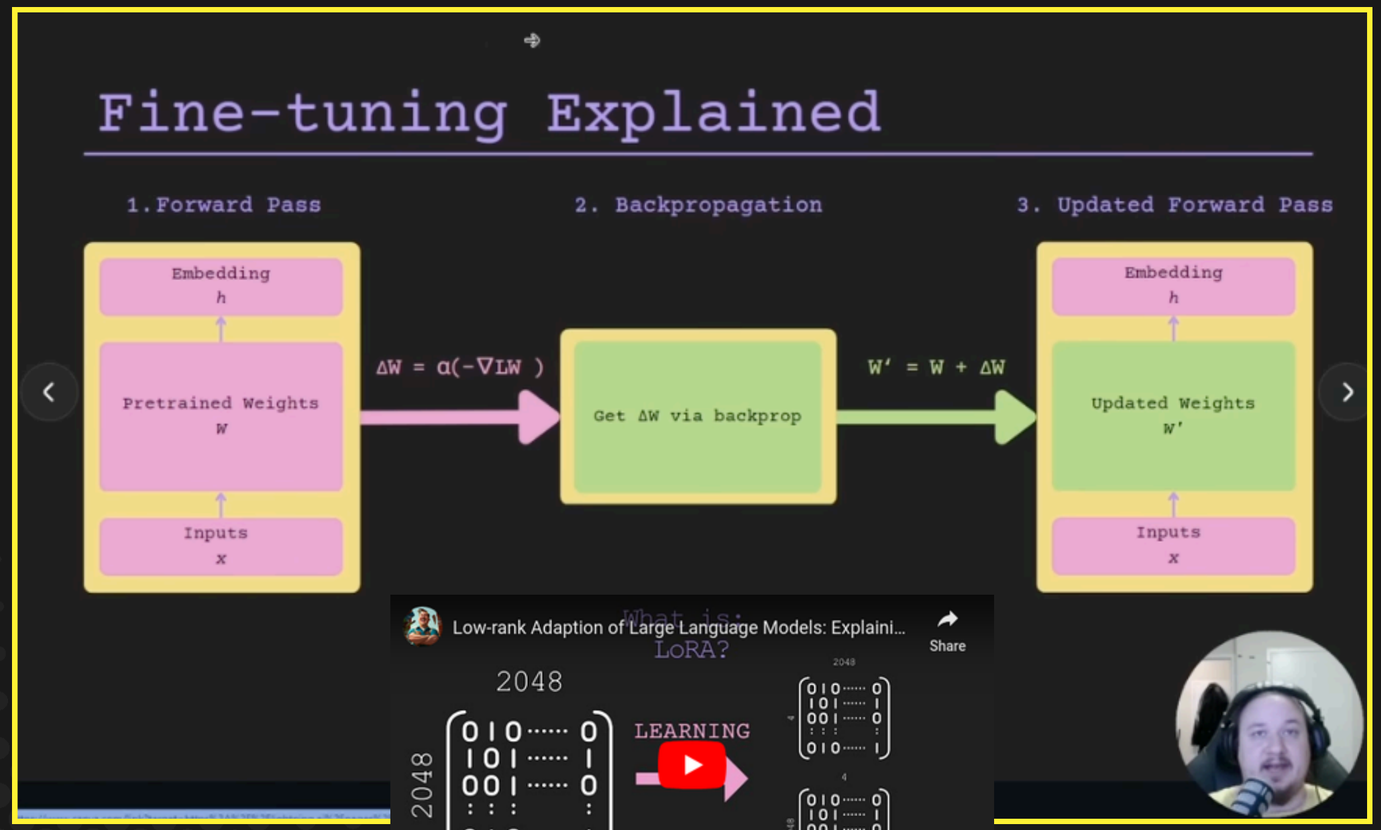
⚖️ FINE-TUNING

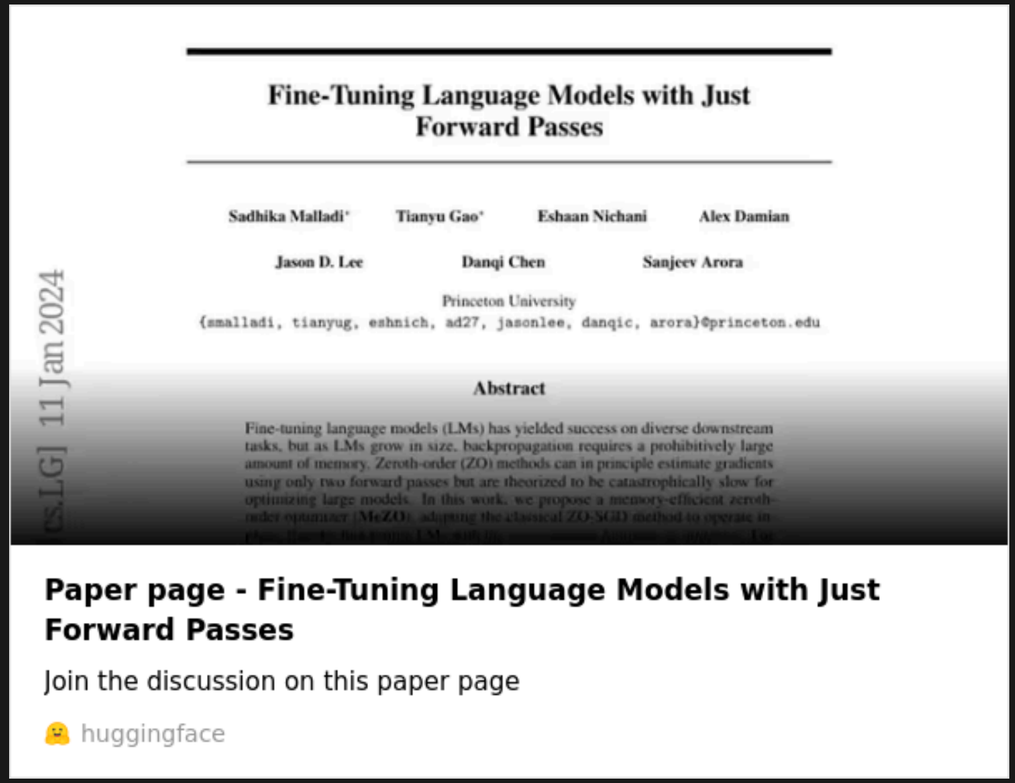
FINE-TUNING LLMS




NYPD
FINE-TUNING
TRAINING
imgflip.com

MECHANICS OF FINE-TUNING

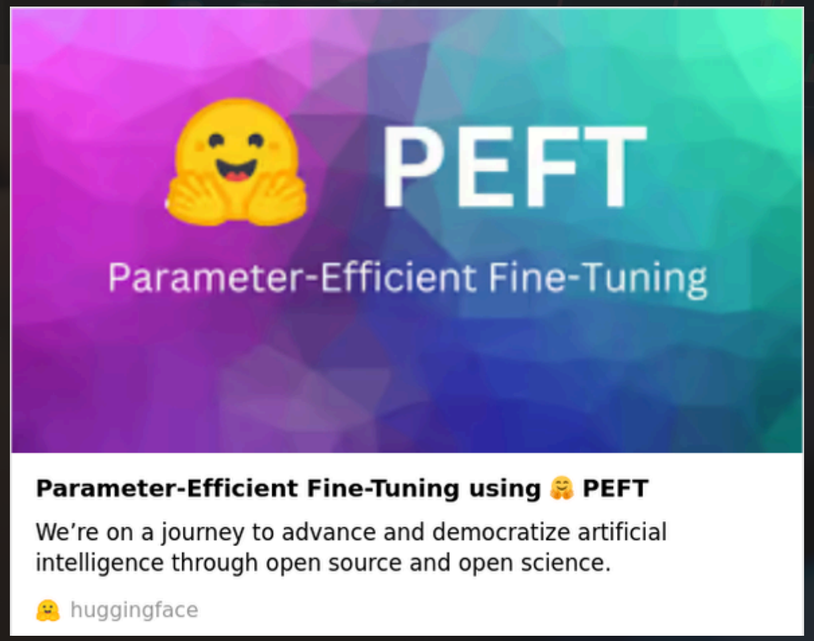




PEFT


I just spent our entire AI budget on fine-tuning a model
But at least we have a working model now, right?
We have a working model, right?

“**Fine-tuning** [pretrained LLMs] on **downstream** datasets results in **huge performance gains** when compared to [off-the-shelf LLMs, zero-shot].”



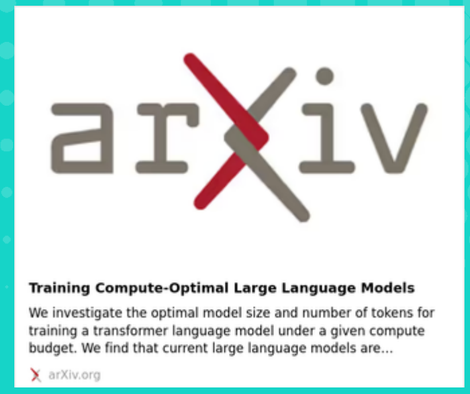
February 10, 2023

WHY PEFT?

LLMs are so big!

7B, 13B, 34B, 70B

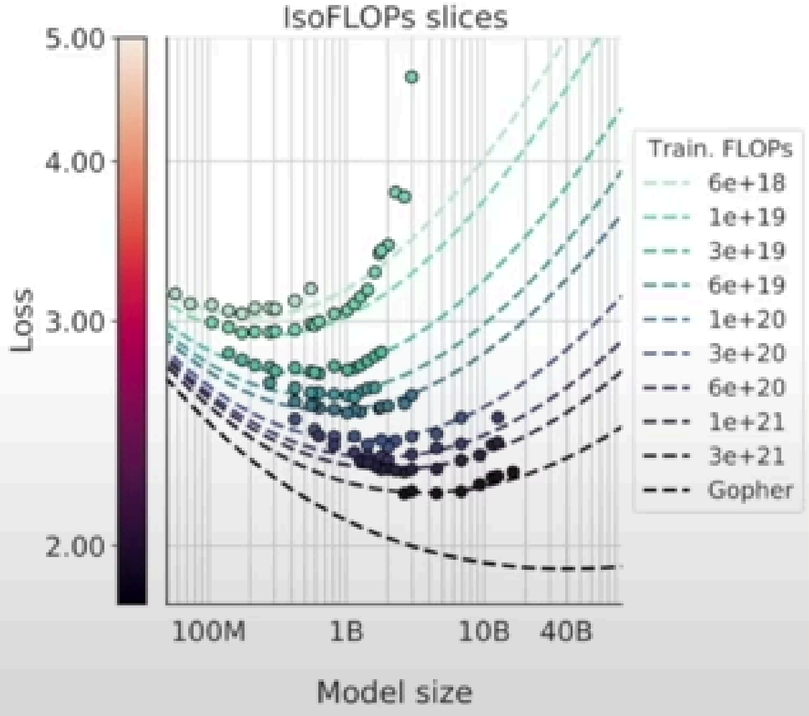
SCALING LAWS

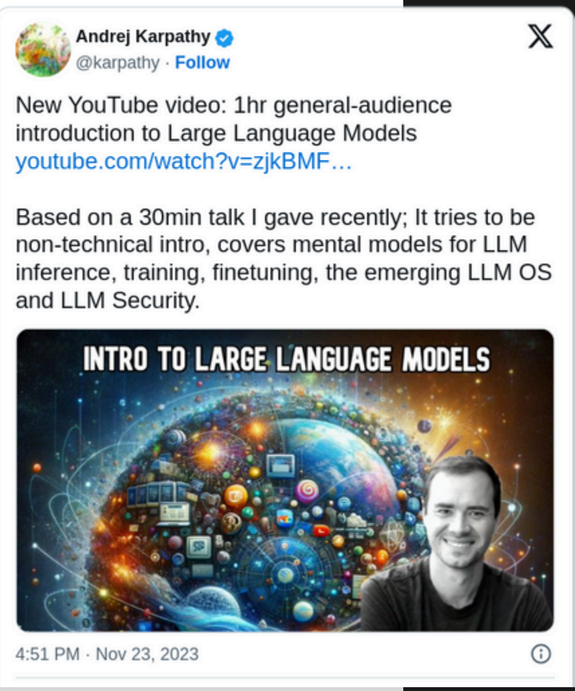


1. N = number of parameters
2. D = amount of text

"If you train a bigger model on more text, we have a lot of confidence that next word prediction will improve." ~ ***Andrej Karpathy***

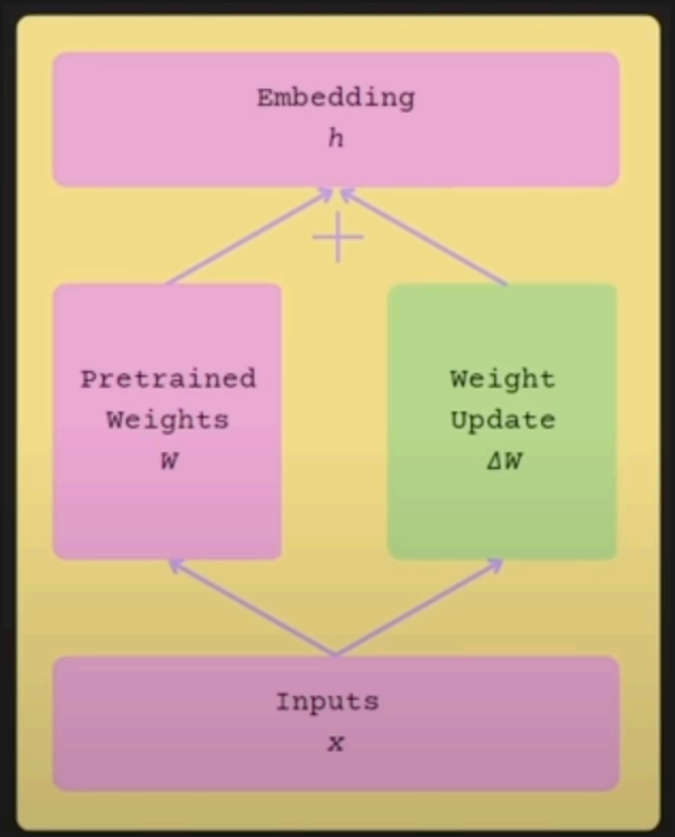
=> We can expect more intelligence "for free" by scaling





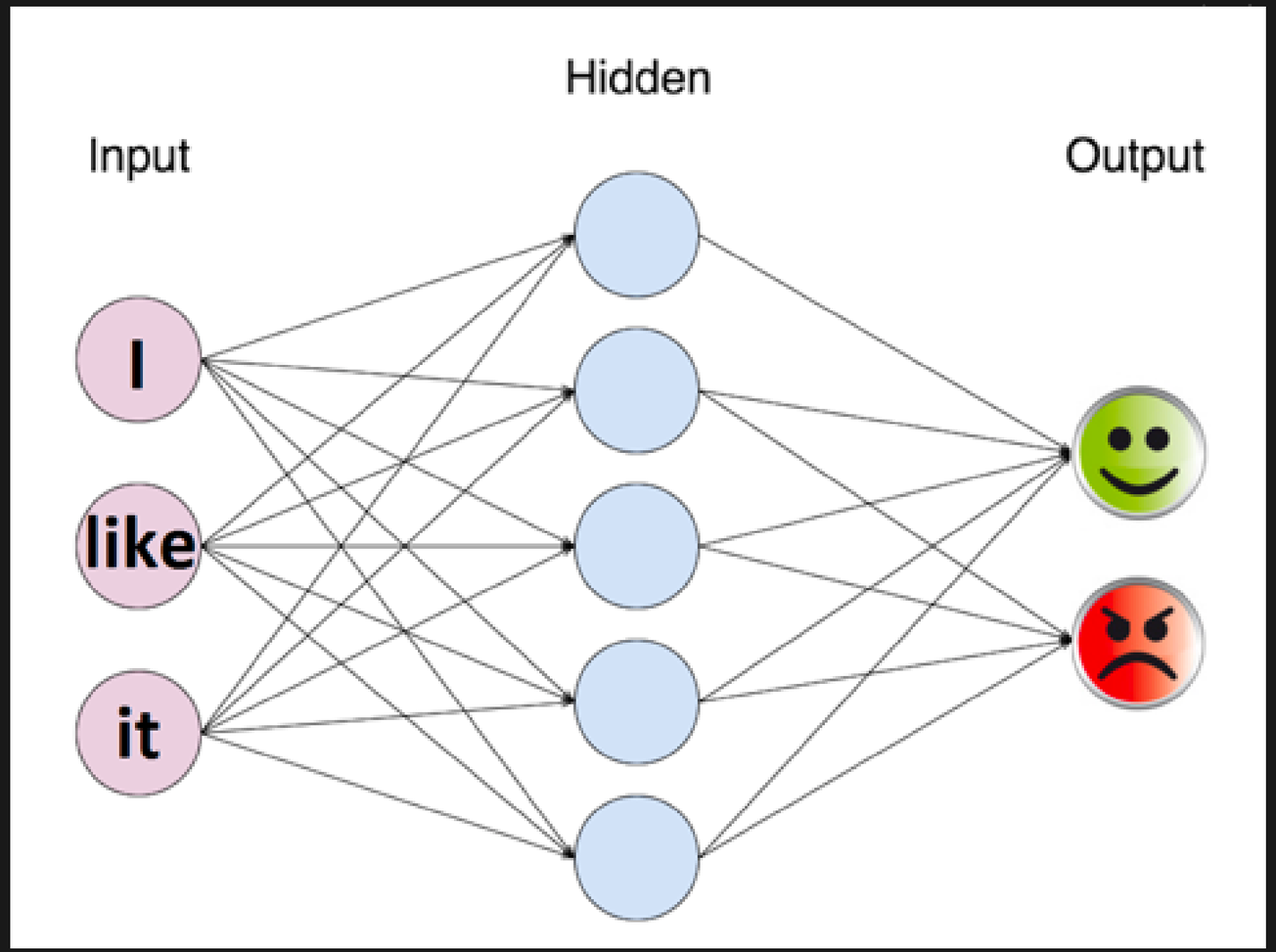
PARAMETERS

- **Weights = parameters**
- Parameters = *Floating point numbers*
- *Full precision = 32-bits ...*



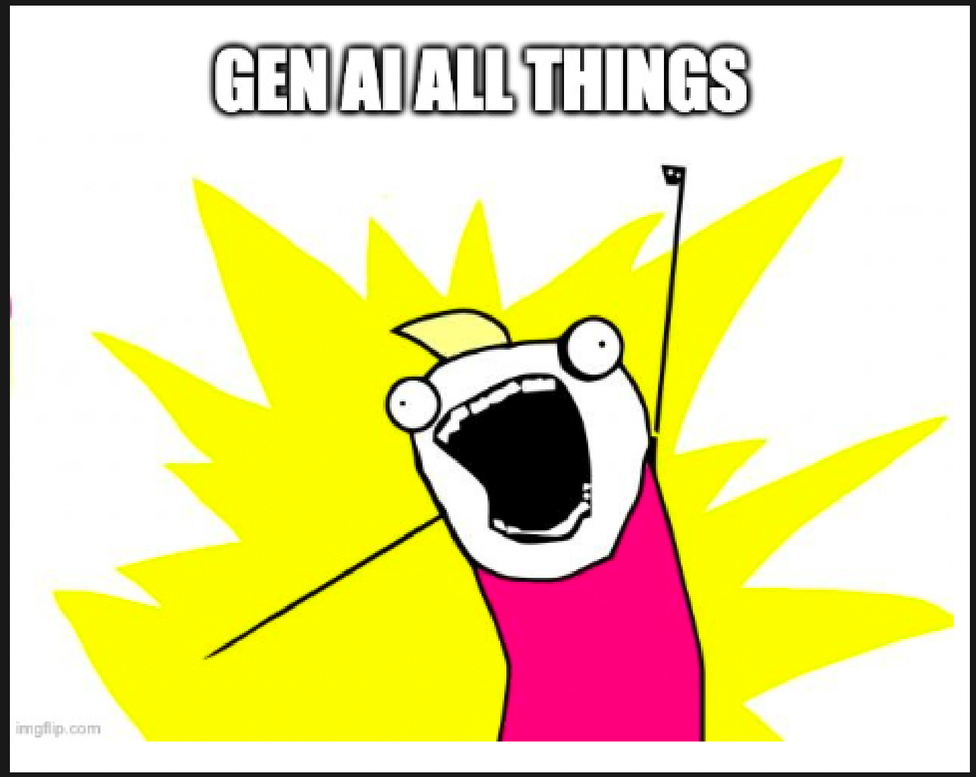
NEURAL NETWORK TRAINING

Pre-trained = mostly tuned!



TWO CHALLENGES

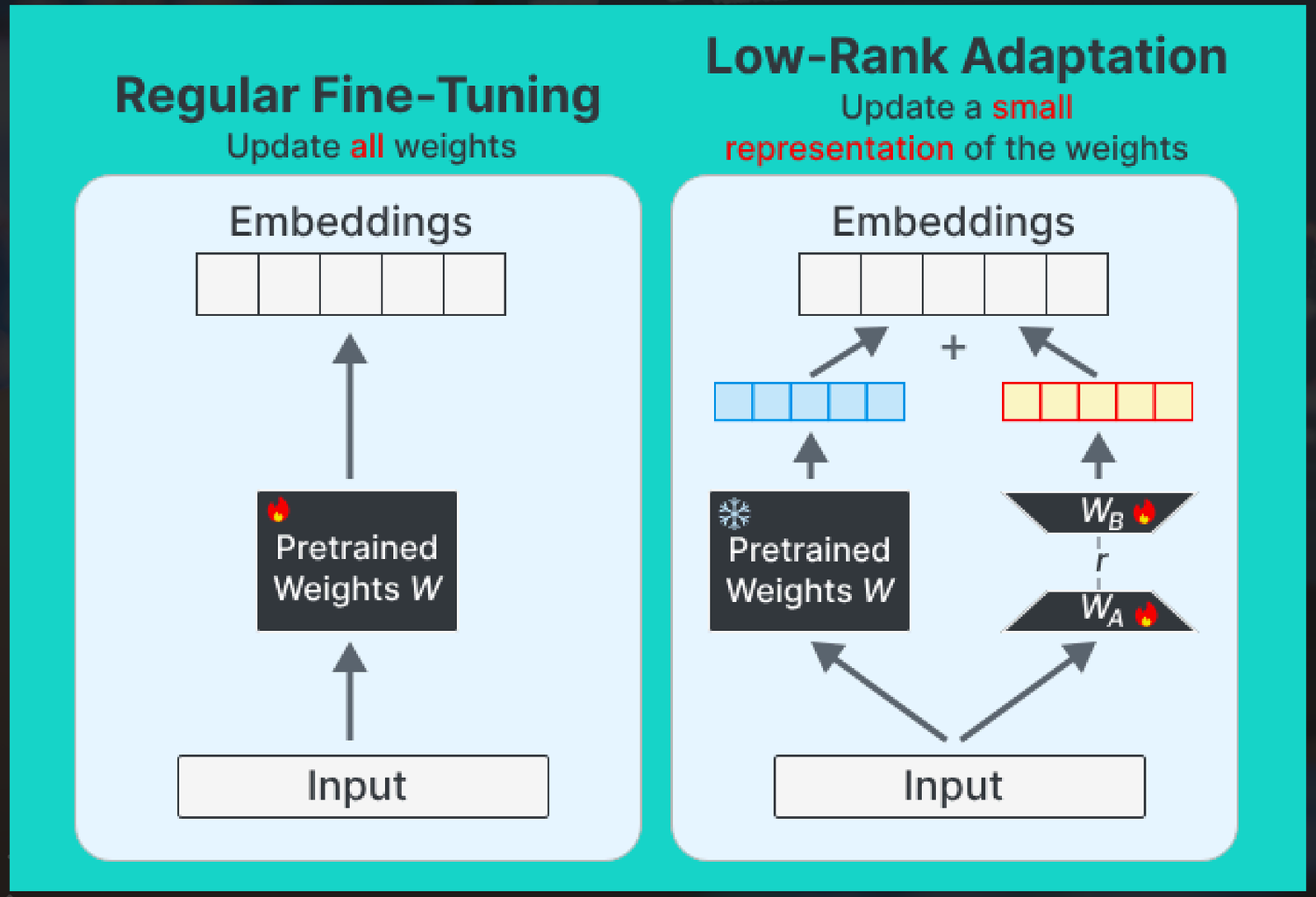
1. **Full fine-tuning** often **infeasible** on consumer hardware.
2. **Storing and deploying** fine-tuned models independently for each downstream task becomes **expensive**.



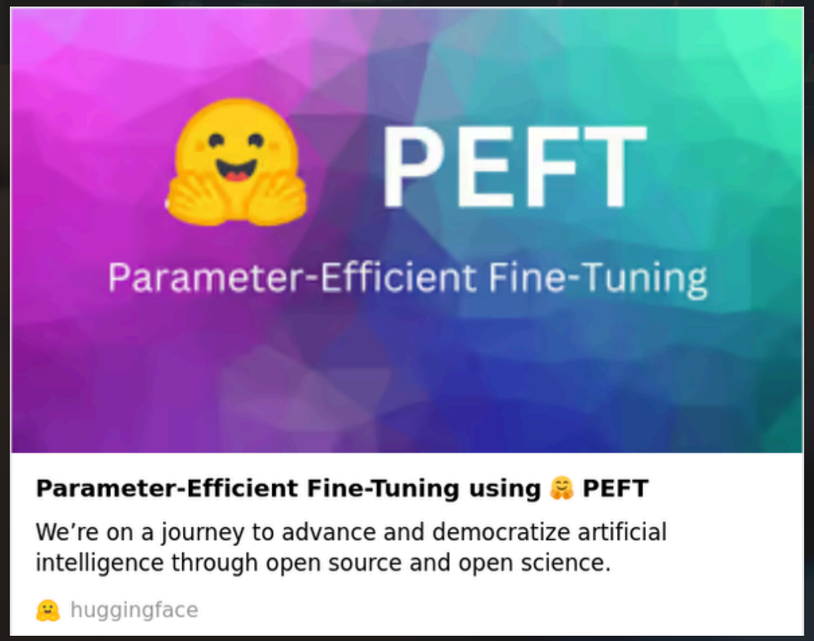
THE SOLUTION

1. Only **fine-tune** a **small number of** (extra) model **parameters**
2. **Better** at:
    a. small data
    b. out-of-domain

LoRA

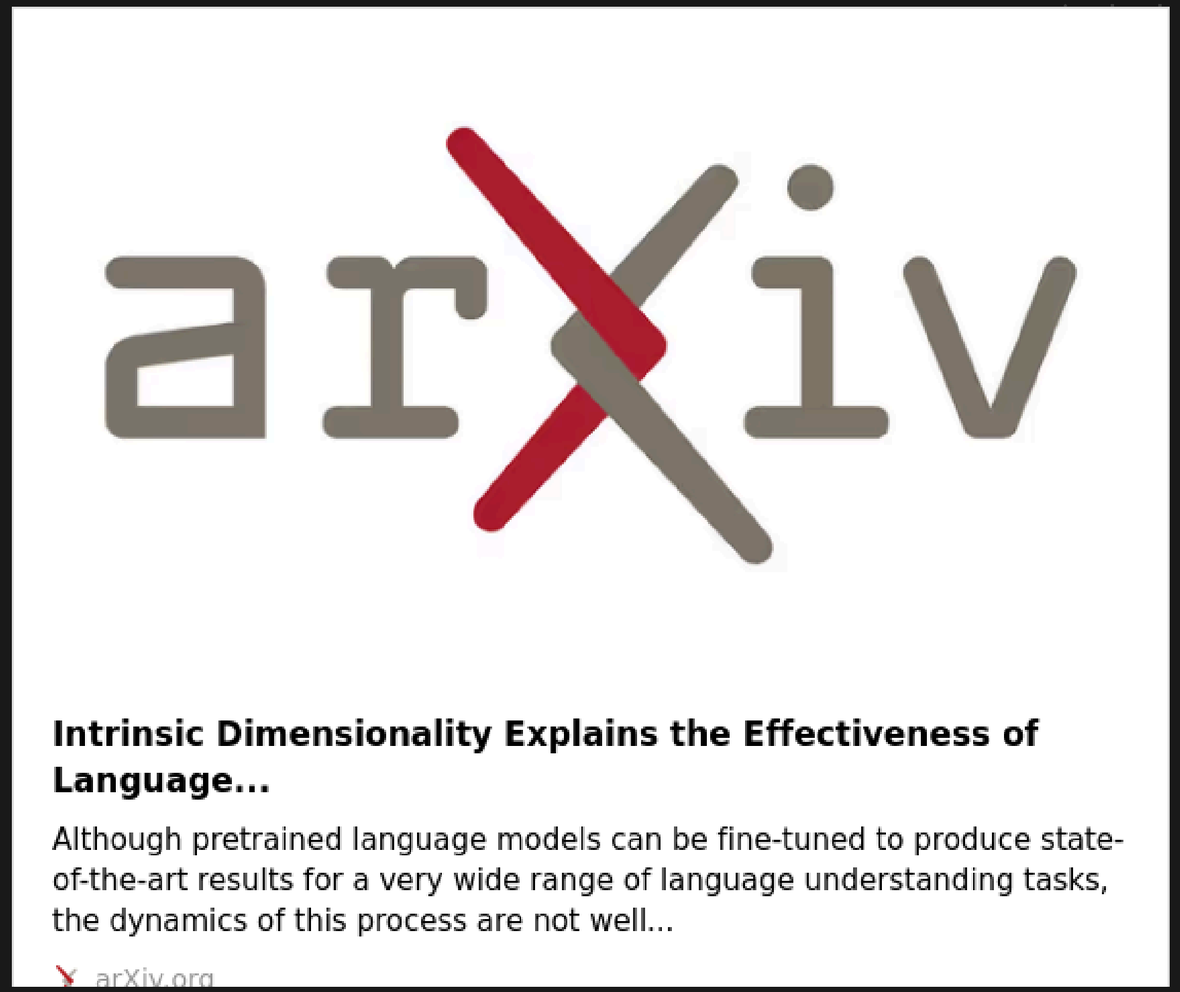

Regular Fine-Tuning
Update all weights
Embeddings
Pretrained Weights W
Input
Low-Rank Adaptation
Update a small representation of the weights
Embeddings
+
Pretrained Weights W
W_B
r
W_A
Input

LoRA is the #1 PEFT Method you should know!



FINE TUNING REALLY WORKS!

“Common pre-trained models have very *low ‘intrinsic dimension’*; [there exists a] re**parameter**ization that is as effective for **fine-tuning** as the full parameter space.”



LoRA Really Works!

Hypothesis:

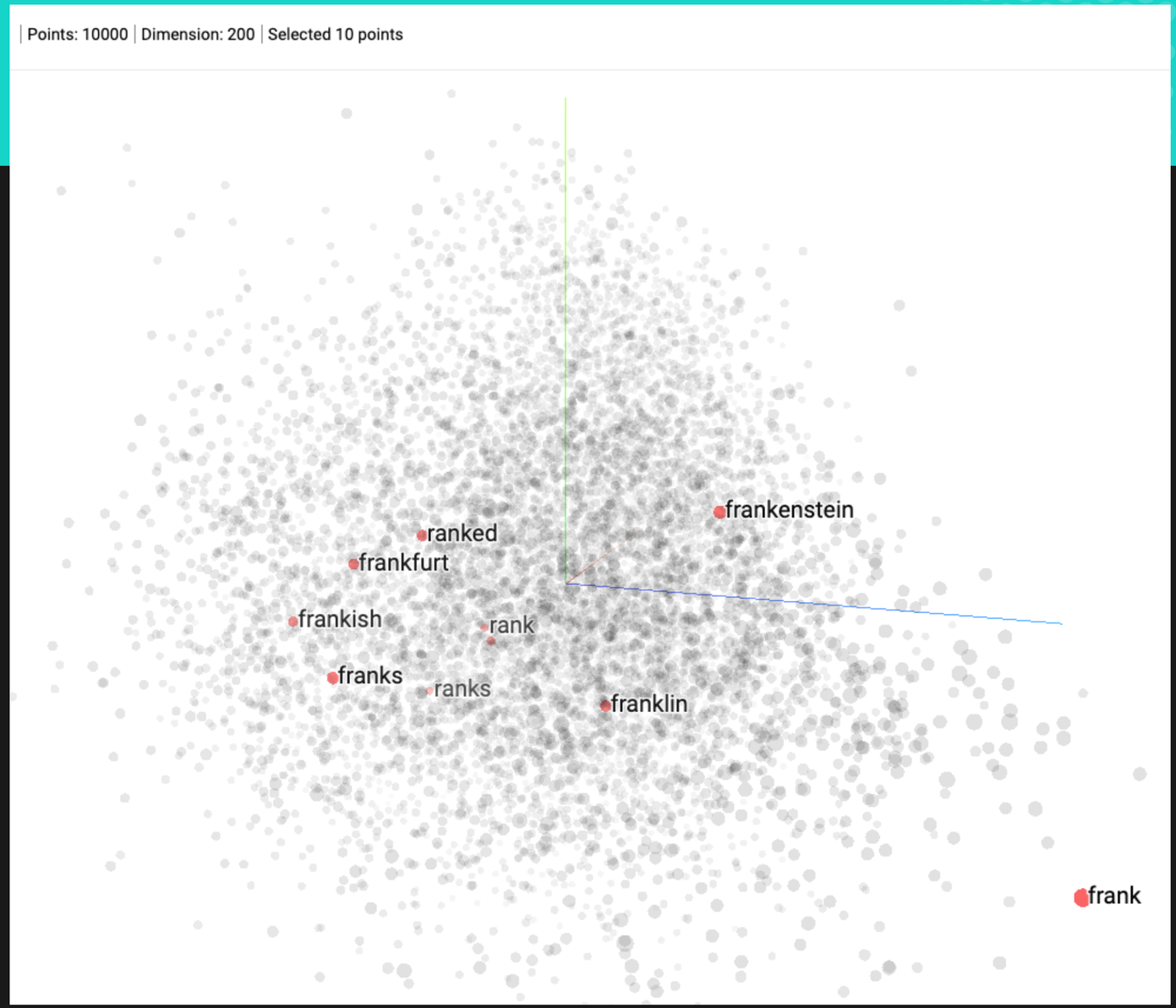
“The updates to the weights also have a low *“intrinsic rank”* during adaption.”

Matrix Rank

Def:

The **number of linearly independent columns** in the weight/parameter matrix

Proxy for the number of **dimensions with unique information**

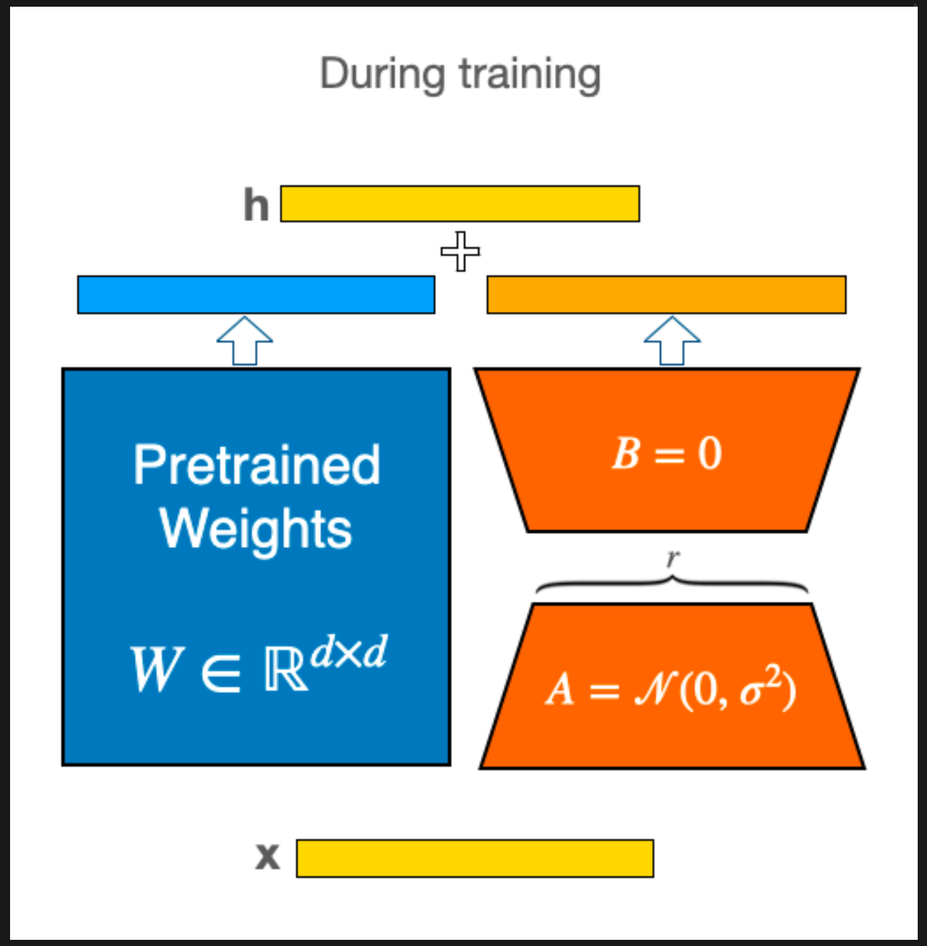


Matrix Decomposition

Convert a difficult matrix computation problem **into several easier tasks**.

How LoRA Works

1. **Freezes** the **pre-trained** model **weights**
2. Injects **trainable rank decomposition matrices** into each layer of the Transformer architecture

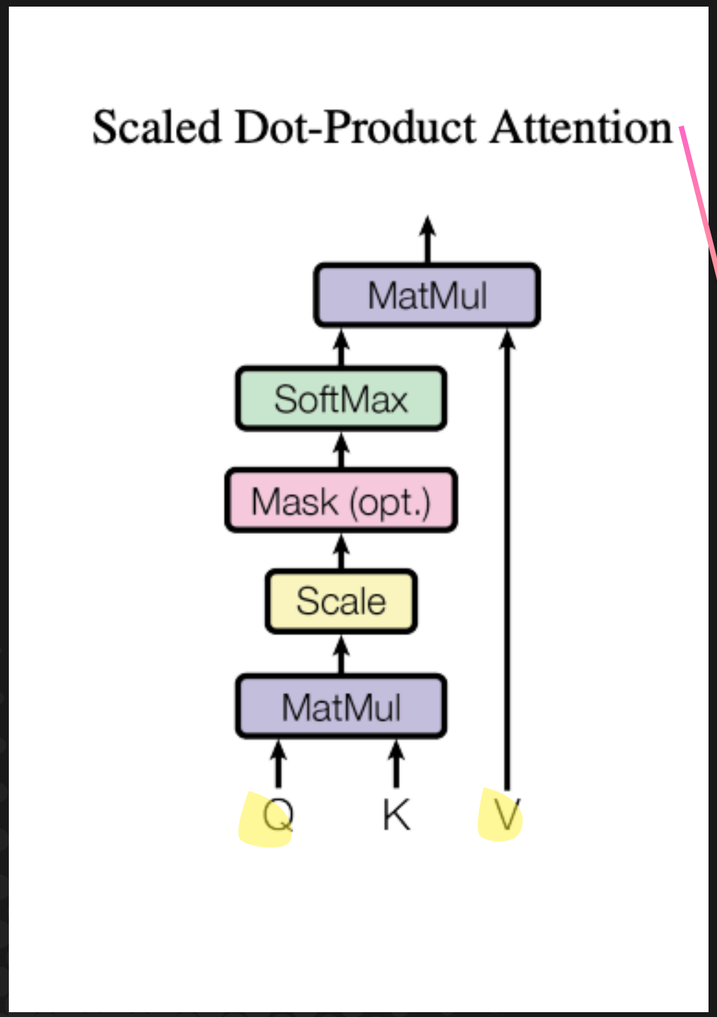


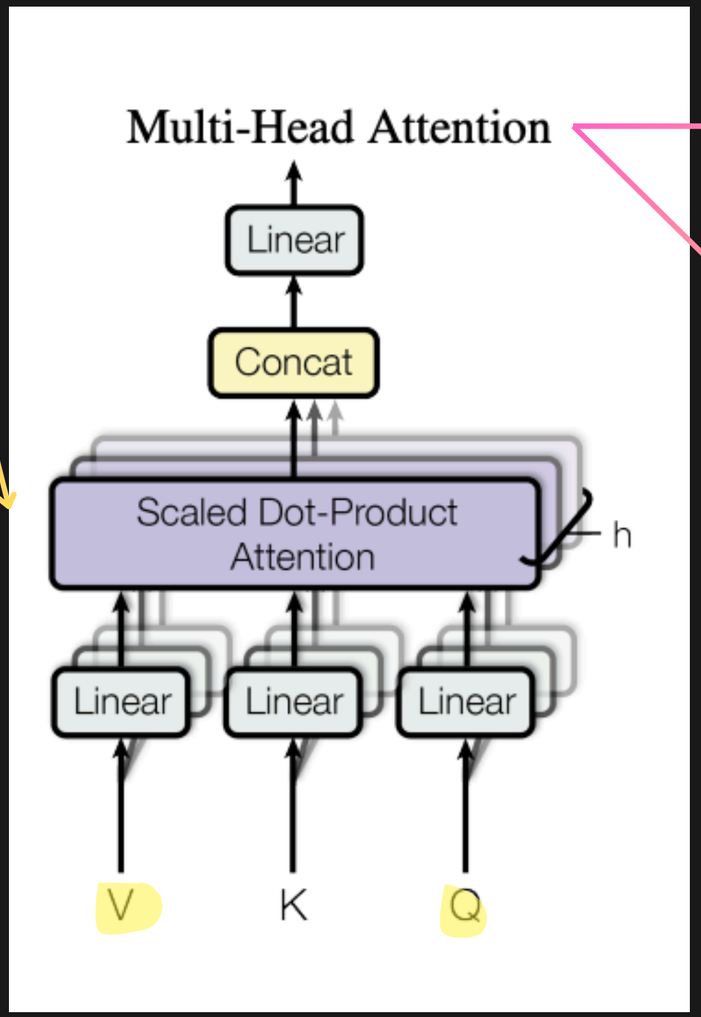
Matrix Decomposition

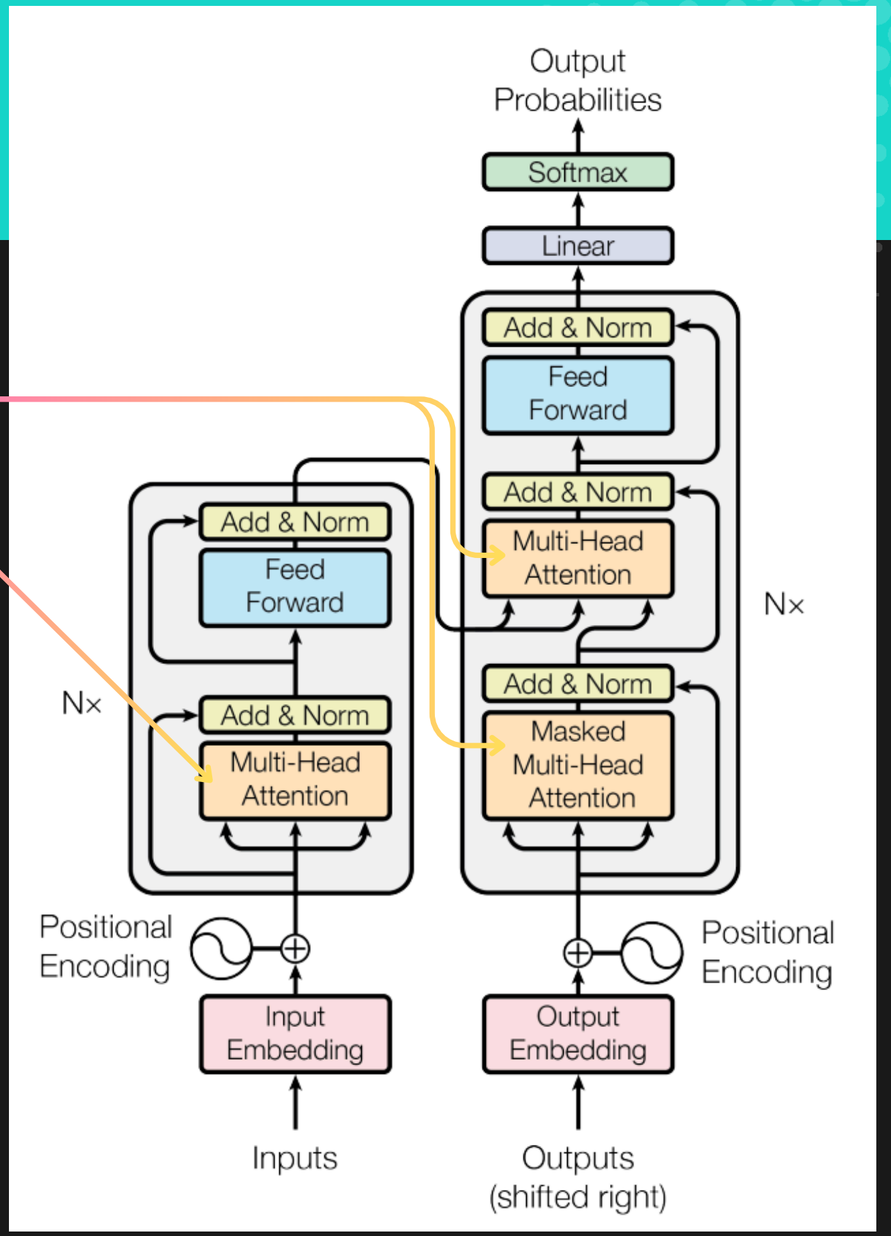
Convert a difficult matrix computation problem **into several easier tasks**.

LoRA and Transformers

Attention

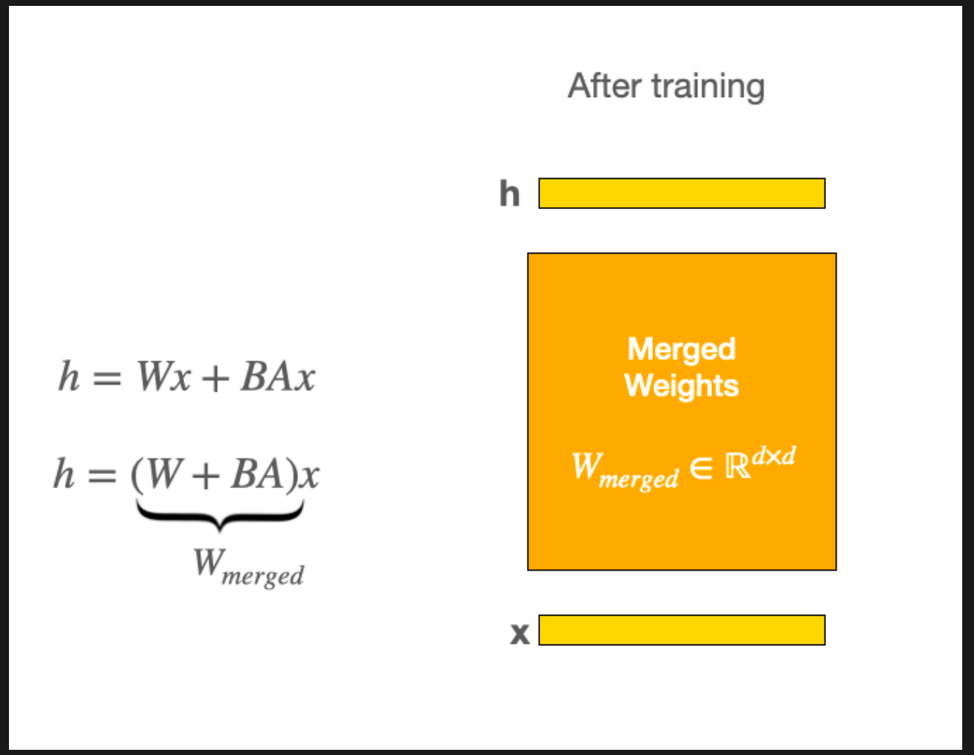

Scaled Dot-Product Attention
MatMul
SoftMax
Mask (opt.)
Scale
MatMul
Q
K
V


Multi-Head Attention
Linear
Concat
Scaled Dot-Product Attention
h
Linear
Linear
Linear
V
K
Q


Output Probabilities
Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Nx
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Nx
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

LoRA Advantages

- More **efficient**
- Adapters for **many tasks**
- **Combine** with other PEFT methods
- Comparable to full fine-tuning
- **No** additional inference **latency**



RECAP

Fine-Tuning

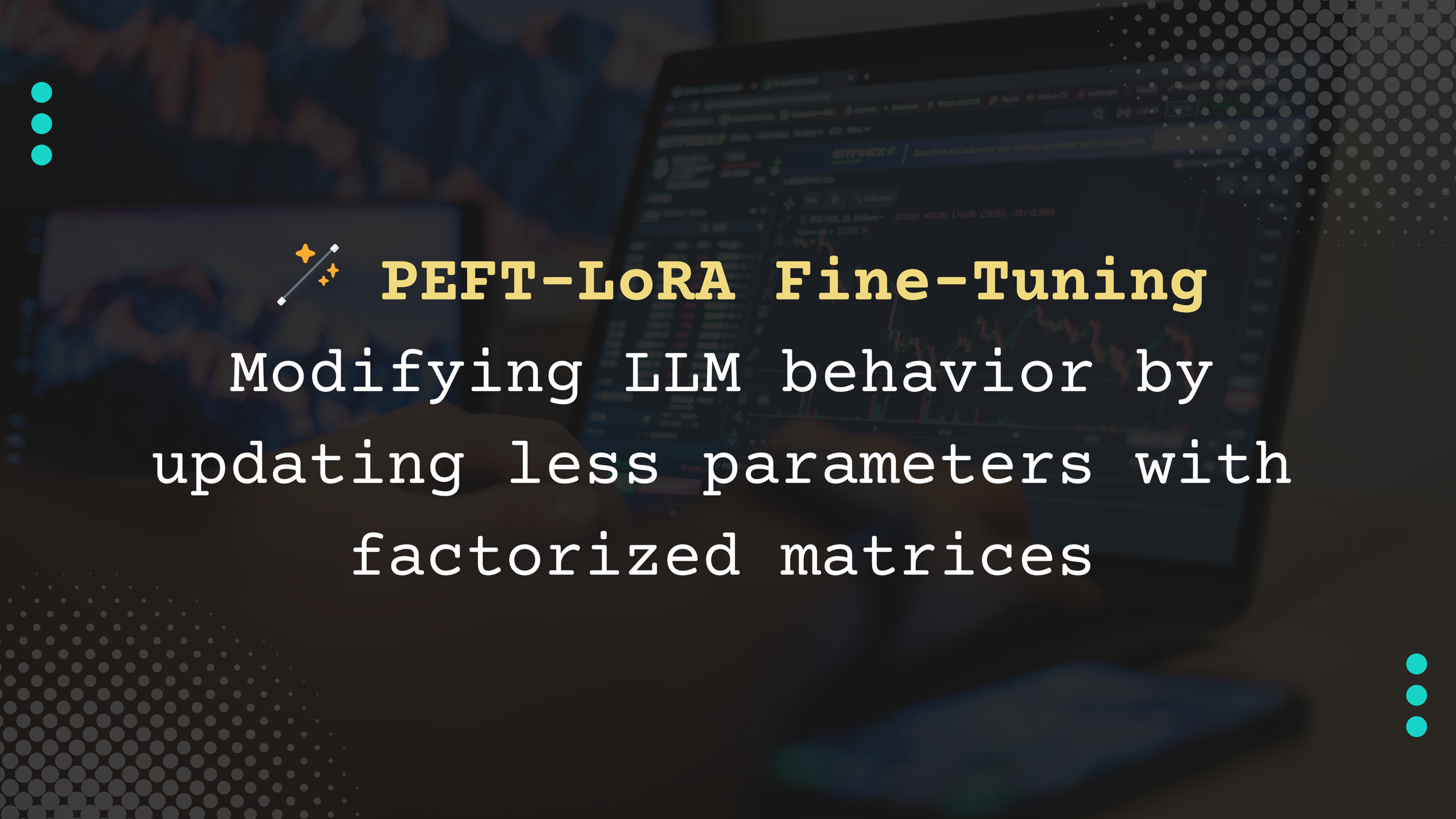
Modifying LLM behavior by updating weights

PEFT

Fine-tuning w/ less parameters

Low Rank Adaption

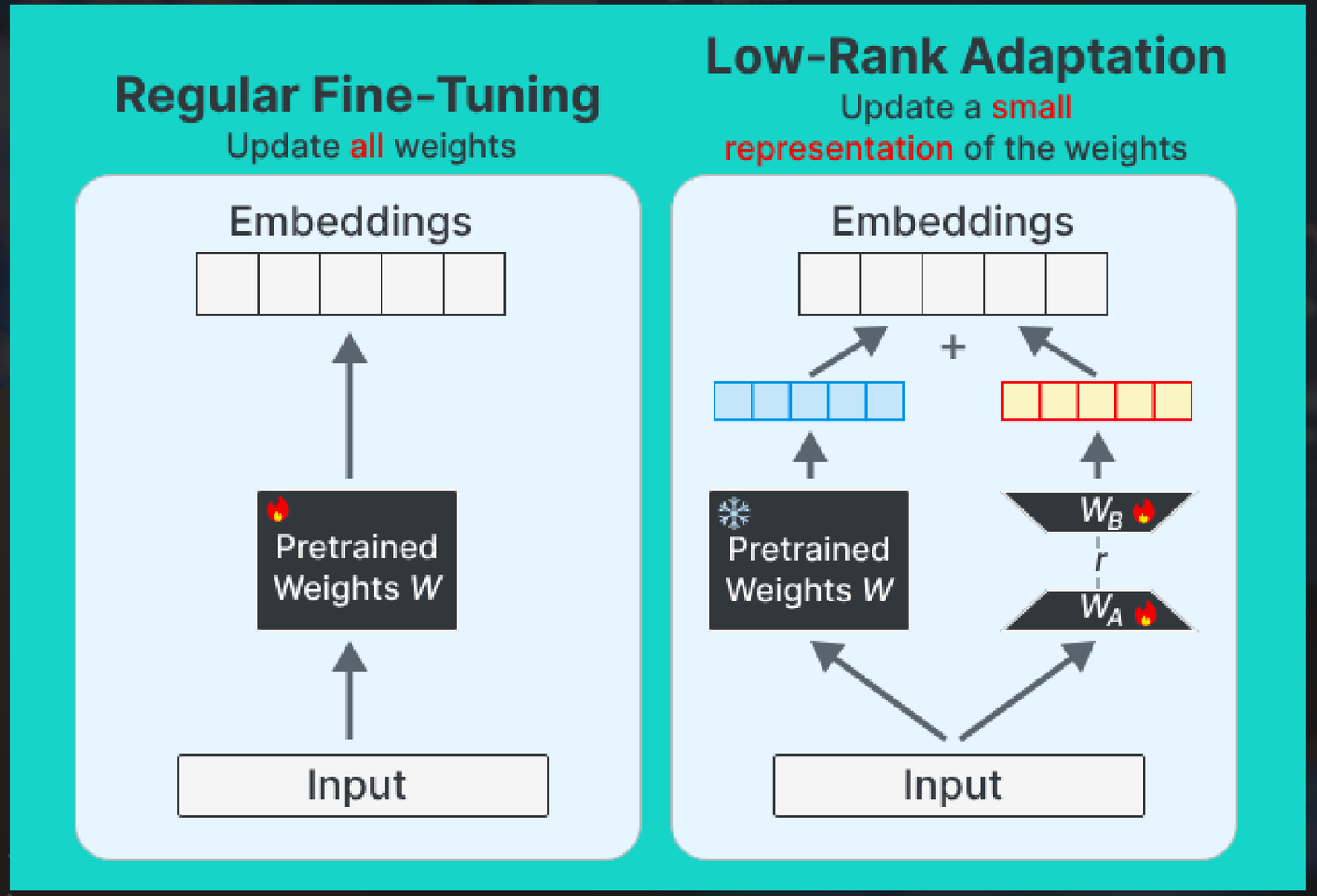
Fine-tuning w/ factorized matrices



PEFT-LoRA Fine-Tuning

Modifying LLM behavior by updating less parameters with factorized matrices

PEFT-LoRA Fine-Tuning



TODAY'S BUILD

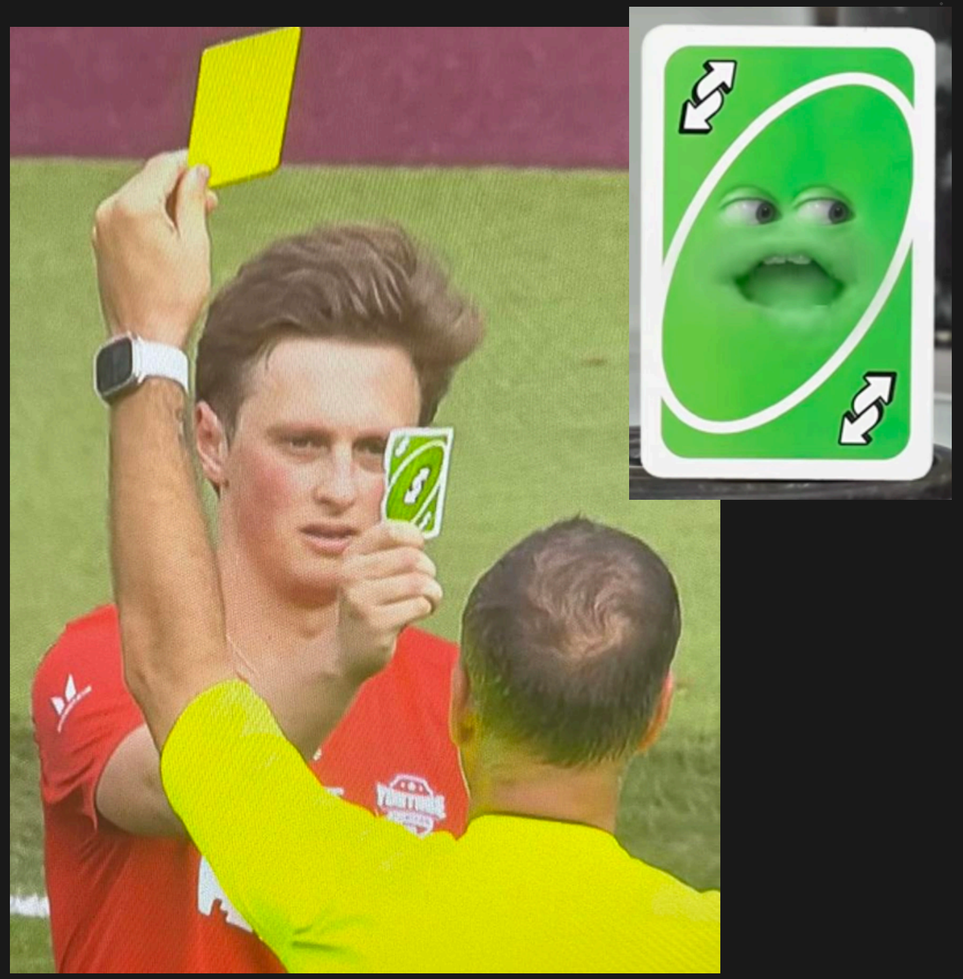
AN UNO REVERSE CARD APPLICATION

Given:

- **Response** (LLM Output)

Predict:

- **Instruction** (Prompt/LLM Input)

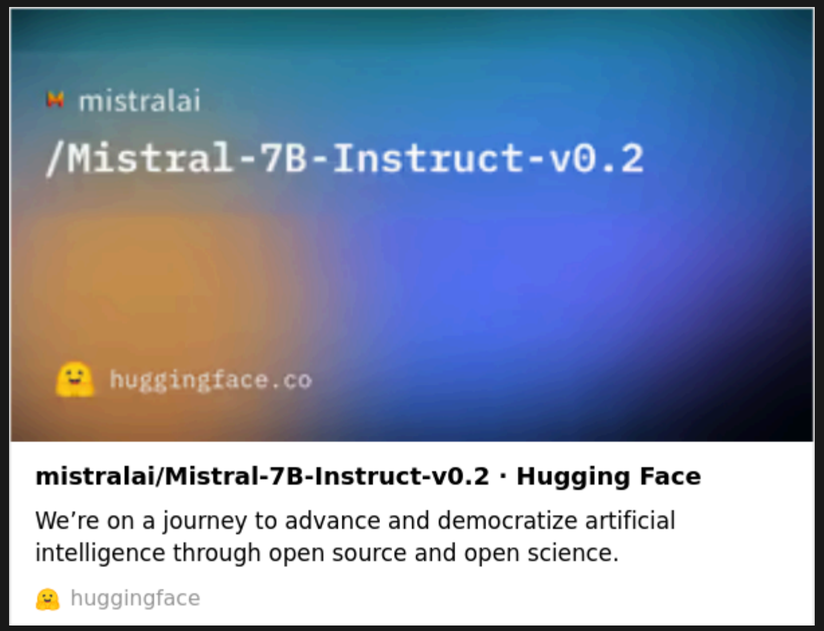


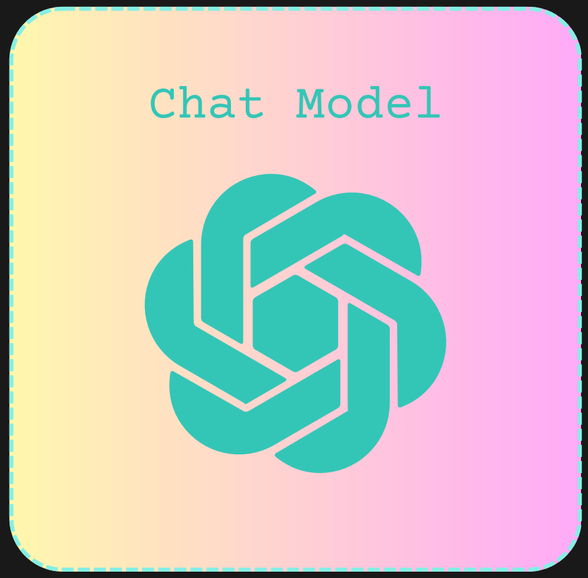
OUR MODEL

CHAT (LLM) MODEL

Chat Model (e.g., LLM)

- Mistral 7B

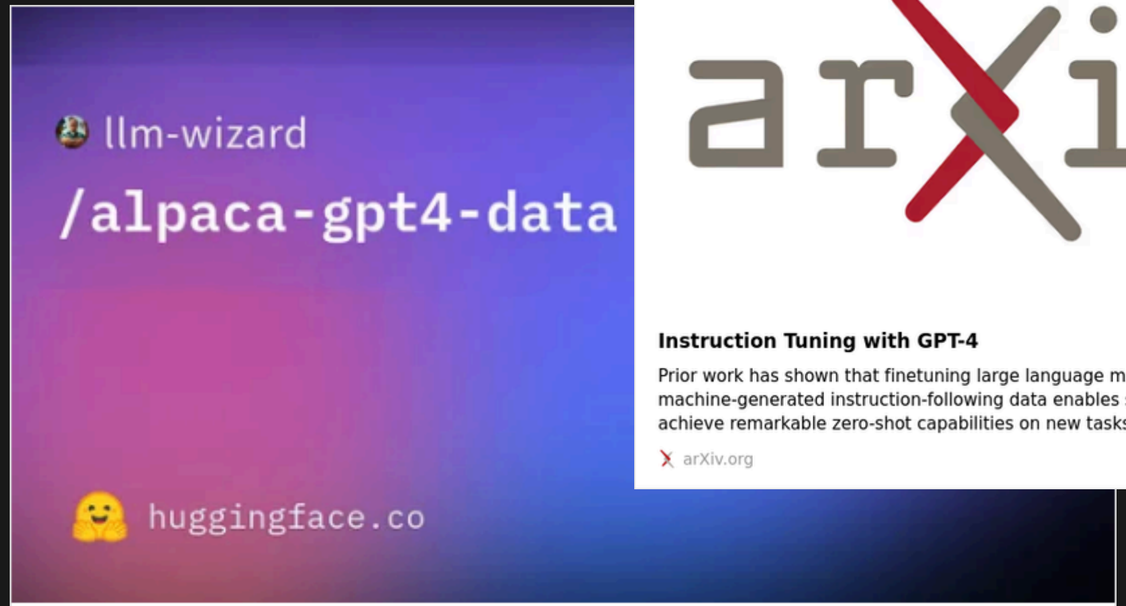




OUR FINE-TUNING DATA

Fine-Tuning Dataset

output string · *lengths* 1 4.57k	instruction string · *lengths* 9 489
1. Eat a balanced and nutritious diet: Make sure…	Give three tips for staying healthy.
The three primary colors are red, blue, and yellow.…	What are the three primary colors?
An atom is the basic building block of all…	Describe the structure of an atom.



llm-wizard/alpaca-gpt4-data · Datasets at Hugging Face

We're on a journey to advance and democratize artificial intelligence through open source and open science.

huggingface

Instruction Tuning with GPT-4

Prior work has shown that finetuning large language models (LLMs) using machine-generated instruction-following data enables such models to achieve remarkable zero-shot capabilities on new tasks,...

arXiv.org

Fine-Tuning with PEFT-LoRA!

Presented by

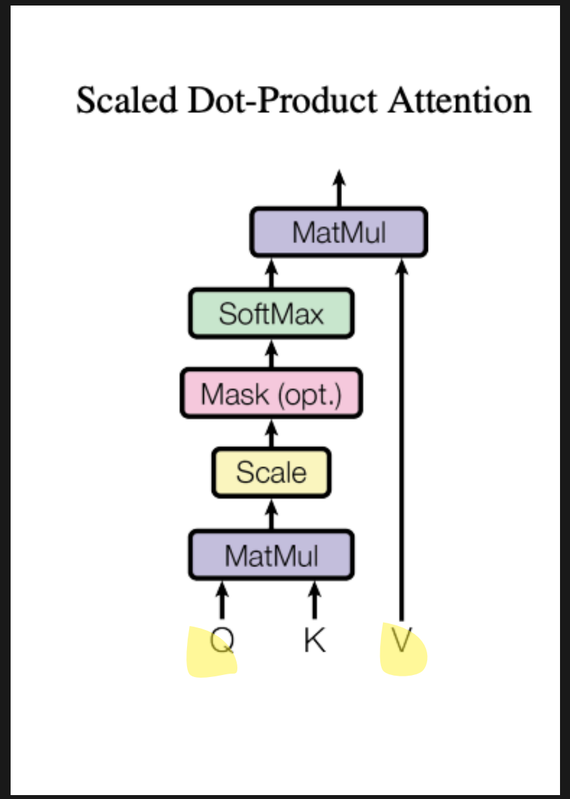
Chris Alexiuk, LLM Wizard 🪄

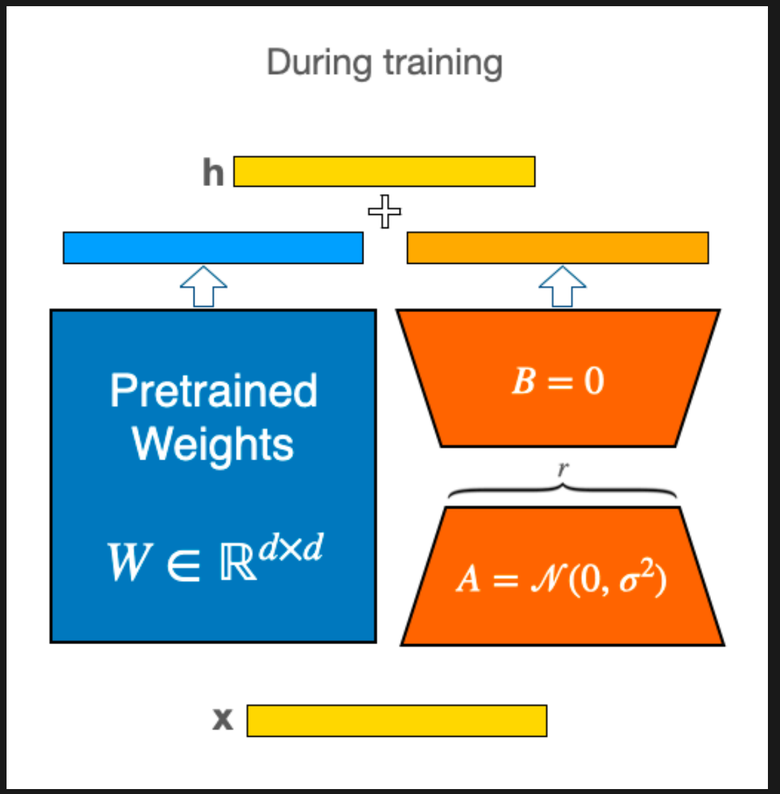
WHAT WE TRAINED

- All parameters = 3,8B
- Trainable parameters = 2,7M

Attention layers = 4096 x 4096

- For Q = 4096 x 4096
  - **LoRA A** = 4096 x 64
  - **LoRA B** = 64 x 4096
- For V = 4096 x 1024
  - **LoRA A** = 4096 x 64
  - **LoRA B** = 64 x 1024



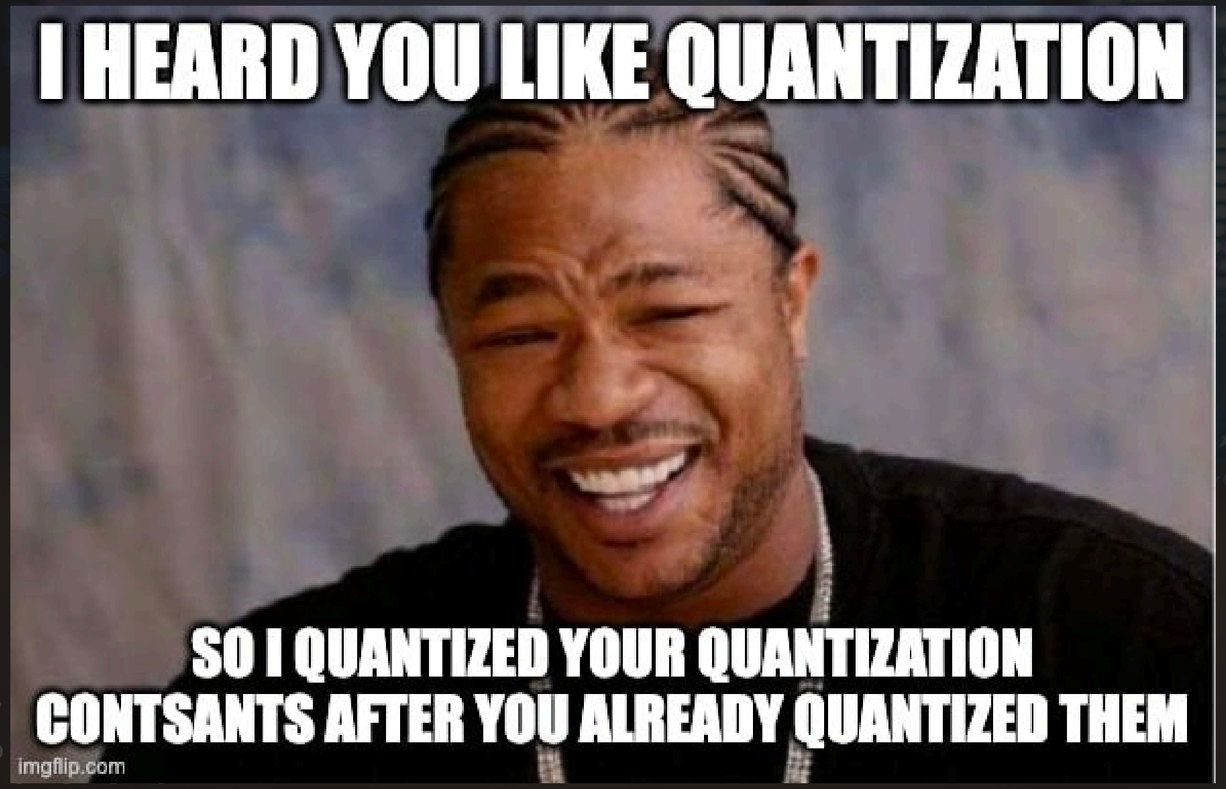


CONCLUSIONS

- 🪄 PEFT-LoRA Fine-Tuning
  - *Modifying LLM behavior by updating less parameters with factorized matrices*

- Fine-Tuning is an important skill for 2024!
  - Helps us leverage the **essential dimensions** of our downstream task!
  - **Small Language Models** (SLMs) are only becoming more popular!

- Next stop … Quantization and QLoRA!



QLoRA: Efficient Finetuning of Quantized LLMs

We present QLoRA, an efficient finetuning approach that reduces memory usage enough to finetune a 65B parameter model on a single 48GB GPU while preserving full 16-bit finetuning task performance....

arXiv.org

QUESTIONS?

Thank you!