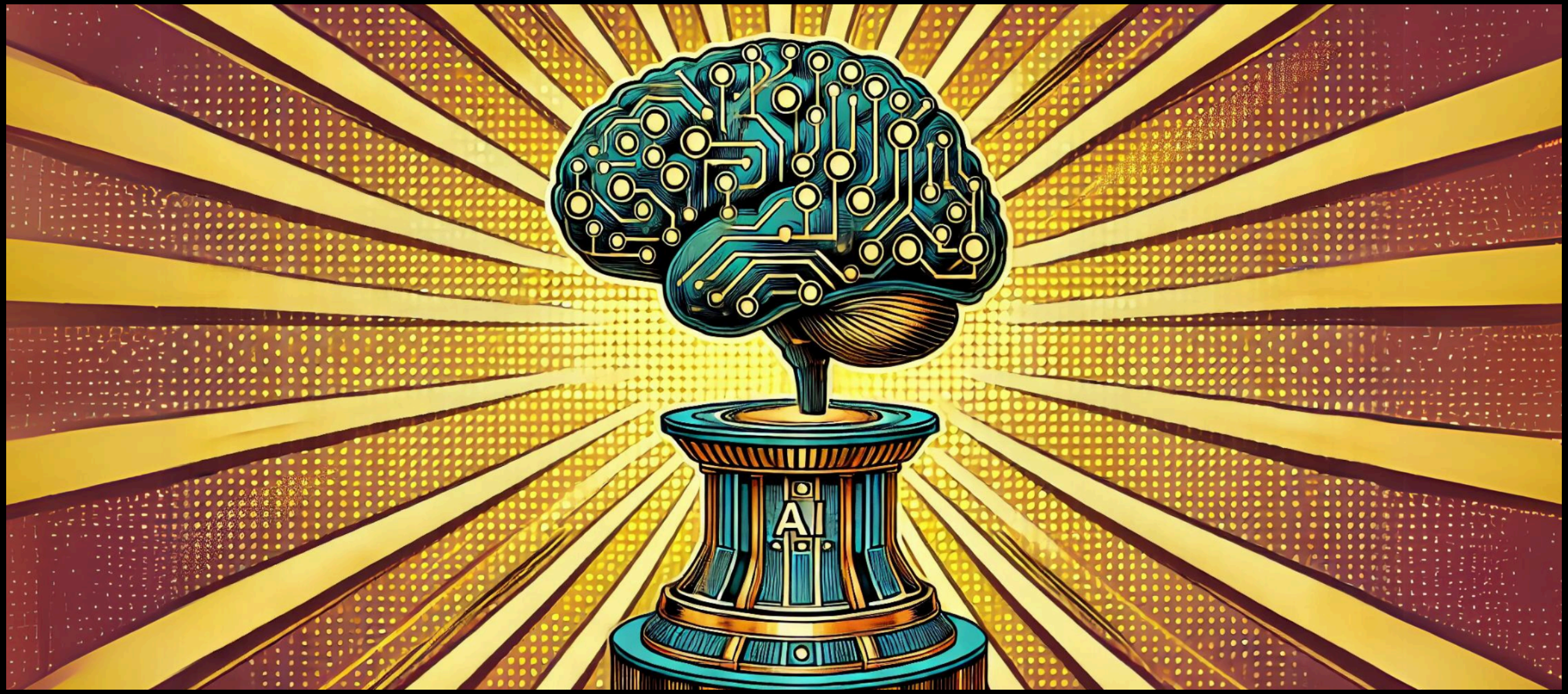
LLM Engineering

MASTER AI & LARGE LANGUAGE MODELS



PROGRESS

Massive week ahead

What you can now do

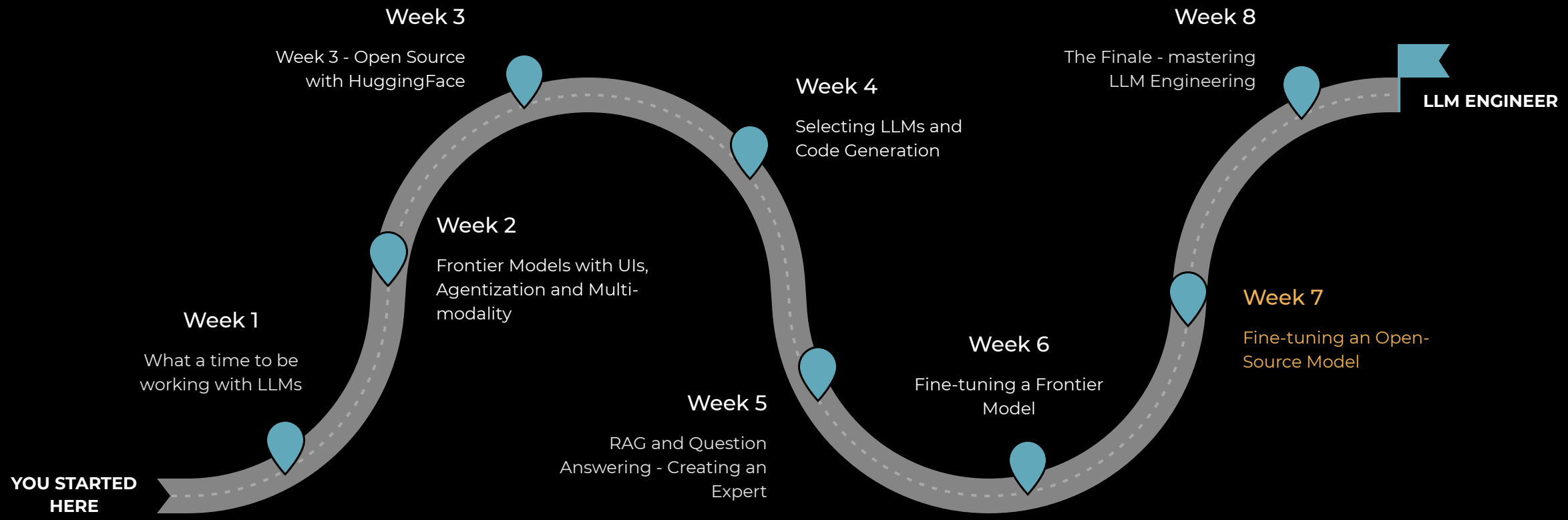
- Generate text and code with Frontier Models including AI Assistants with Tools, and with open-source models with HuggingFace transformers
- Create advanced RAG solutions with LangChain
- Follow a 5 step strategy to solve problems, including dataset curation and making a baseline model with traditional ML and making a Frontier solution, and fine-tuning Frontier Models

---

Today your advanced journey begins - you will soon be able to:

- Explain LoRA for fine-tuning Open Source models
- Describe Quantization and QLoRA
- Explain 3 key hyper-parameters: r, alpha and target modules

Reminder of the 8 weeks to mastery



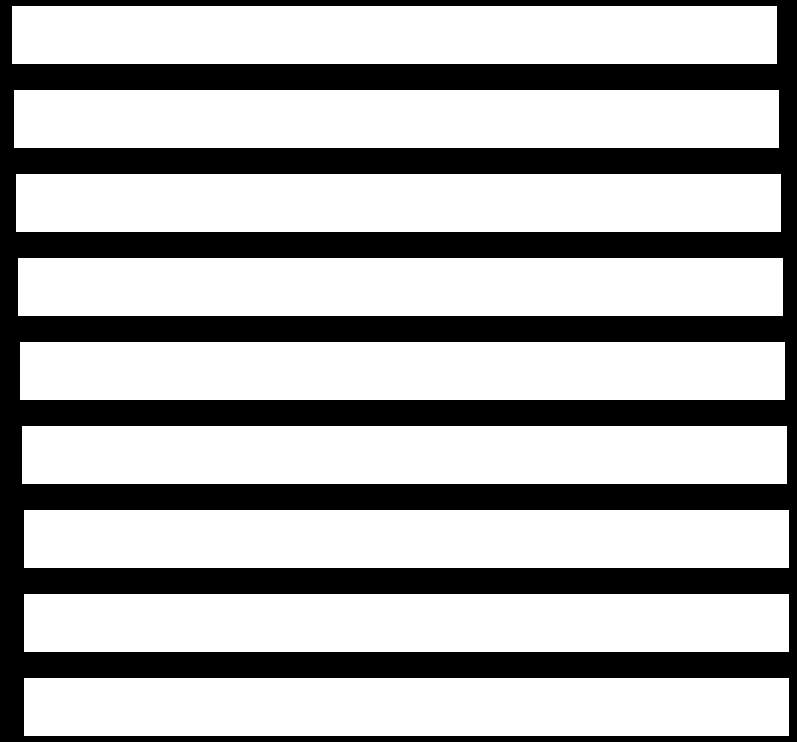
High level explanation of LoRA

Using Llama 3.1 with 8B weights - far too much for us to train on a GPU

- Llama 3.1 8B architecture consists of 32 groups of modules stacked on top of each other, called 'Llama Decoder Layers'
- Each has self-attention layers, multi-layer perceptron layers, SiLU activation and layer norm
- These parameters take up 32GB memory

High level explanation of LoRA

Step 1: Freeze the weights - we will not optimize them

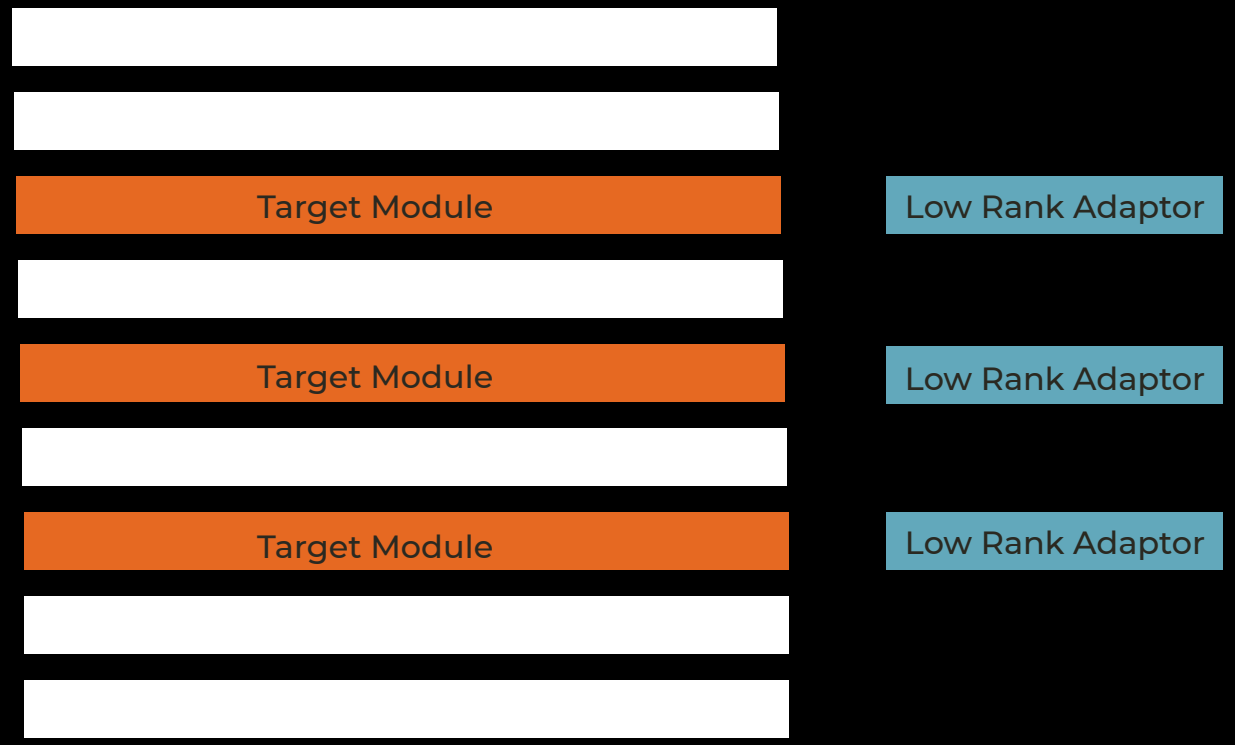


High level explanation of LoRA

Step 2: Select some layers to target, called "Target Modules"

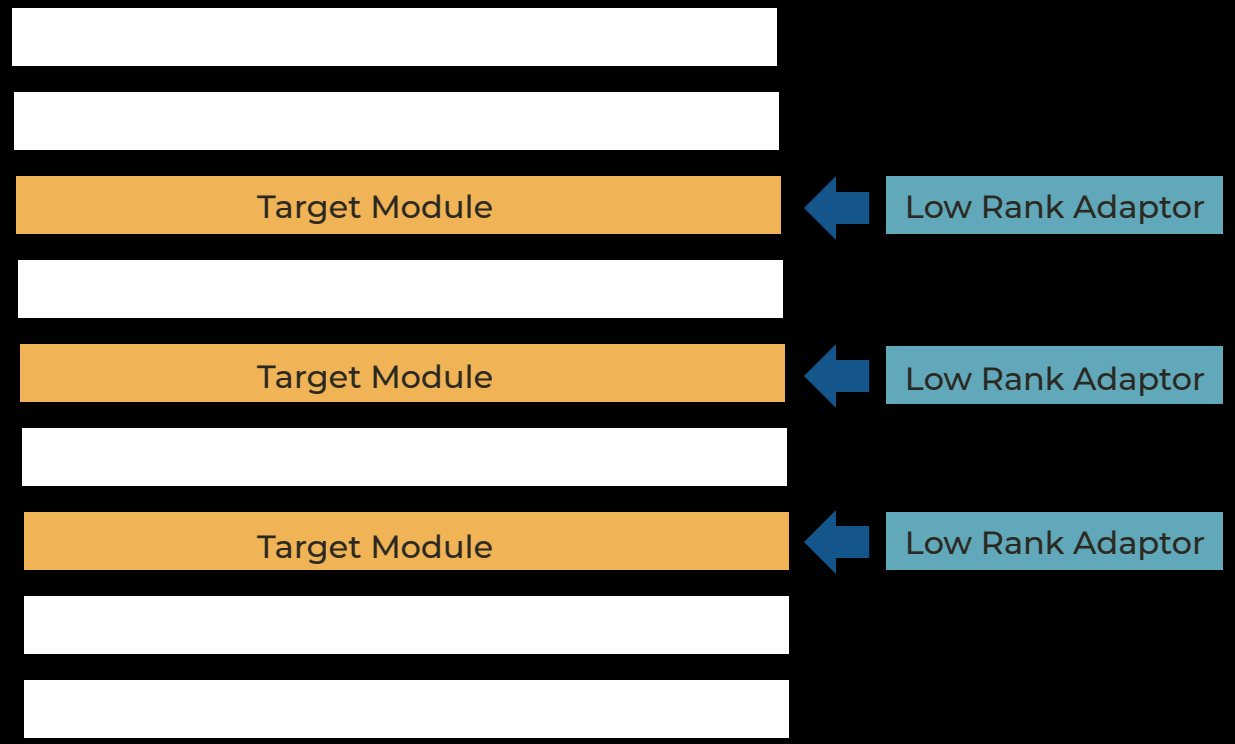
High level explanation of LoRA

Step 3: Create new "adaptor" matrices with lower dimensions, fewer parameters



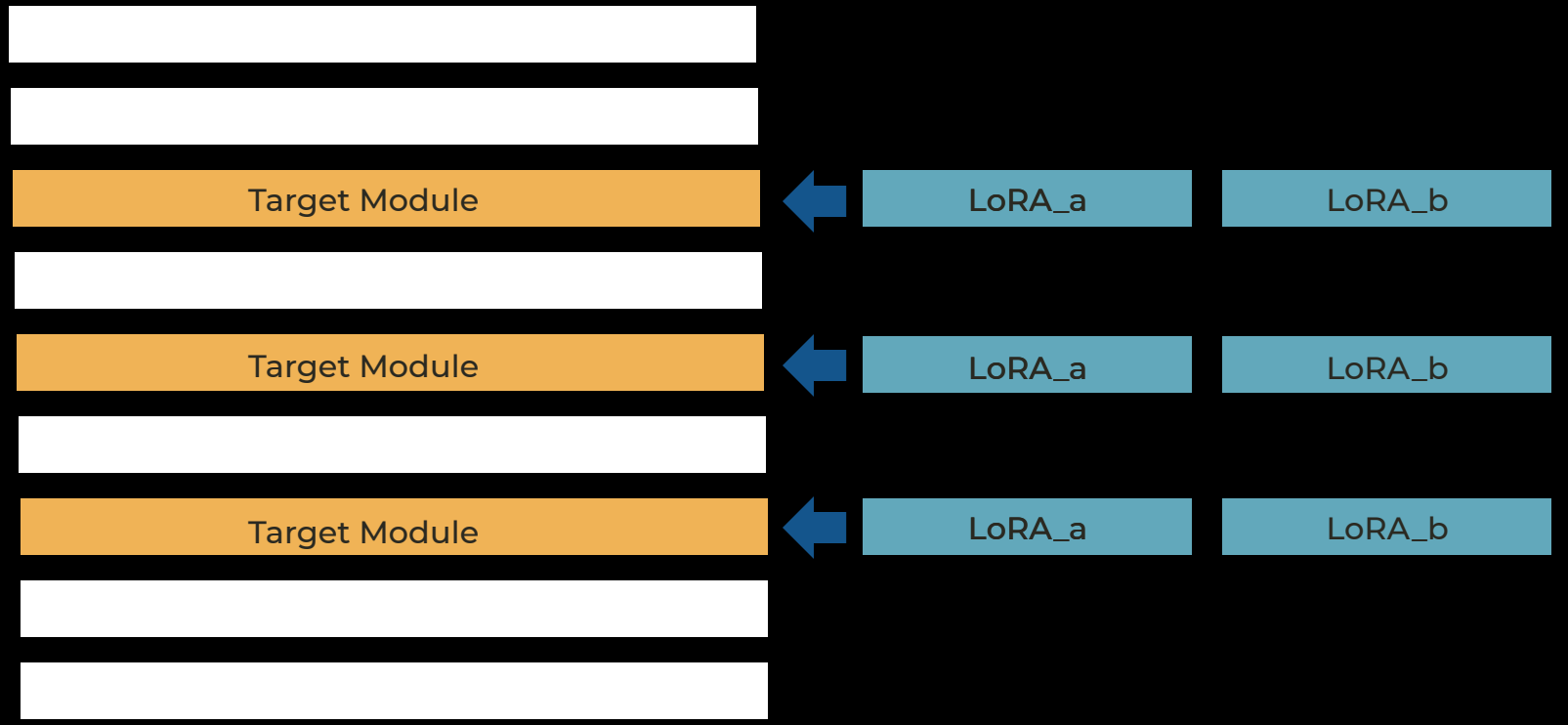
High level explanation of LoRA

Step 4: Apply these adaptors to the Target Modules to adjust them - and these get trained



High level explanation of LoRA

To be more precise: there are in fact two LoRA matrices that get applied



Three Essential Hyperparameters

For LoRA Fine-Tuning

r

The rank, or how many dimensions in the low-rank matrices

RULE OF THUMB:

Start with 8, then double to 16, then 32, until diminishing returns

Alpha

A scaling factor that multiplies the lower rank matrices

RULE OF THUMB:

Twice the value of r

Target Modules

Which layers of the neural network are adapted

RULE OF THUMB:

Target the attention head layers

Quantization - the Q in QLoRA

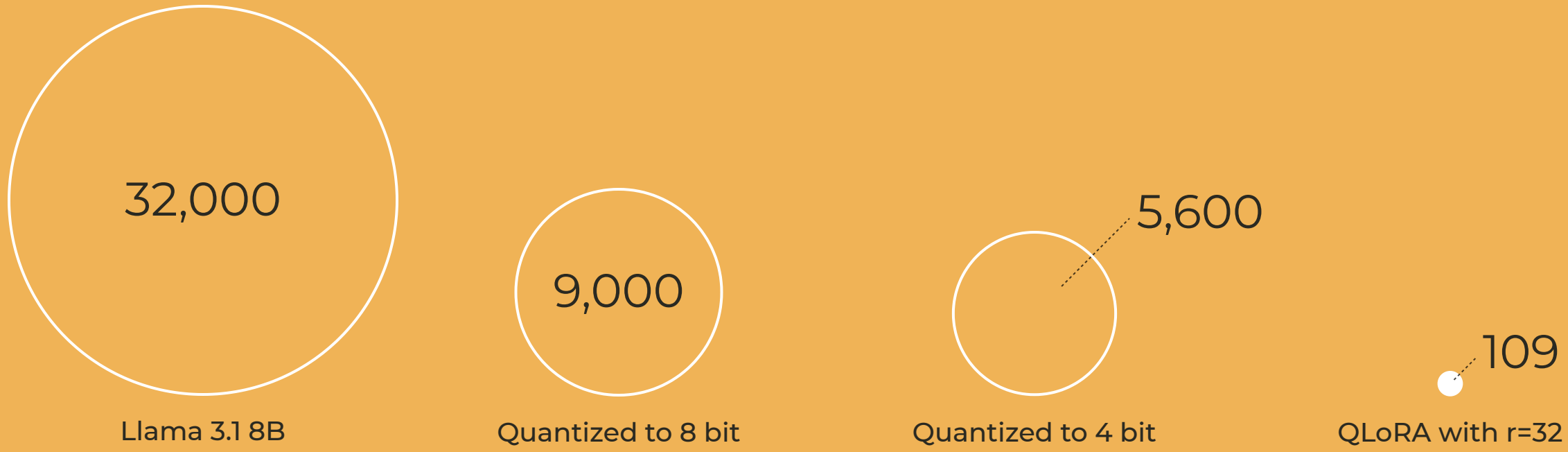
Even the 8B variants are enormous

- 8 Billion * 32 bits = 32GB
- Intuition: keep the number of weights but reduce their precision
- Model performance is worse, but the impact is surprisingly small
- Reduce to 8 bits, or even to 4 bits

  Technical note 1: 4 bits are interpreted as float, not int

  Technical note 2: the adaptor matrices are still 32 bit

Size of Weights in MB



PROGRESS

Essential expertise acquired

What you can now do

- Generate text and code with Frontier Models and Open Source models using APIs and HuggingFace, including tools, assistants and RAG
- Follow a 5 step strategy to solve problems, including dataset curation, making a baseline model, and fine-tuning a Frontier model
- Explain QLoRA for fine-tuning open-source models including defining target modules, r and alpha

---

By next time you will be able to:

- Select an open source model for fine tuning
- Compare instruct and base variants for a task
- Evaluate a base model against a business objective