From Software Engineer to AI Data Scientist

For software engineers exploring a career move into Data Science, GenAI, LLMs

Ed Donner

Co-Founder and CTO, Nebula.io

Pearson

AI

Prepare to step into a new world

THANK YOU!!

For spending 5 hours on Friday with me.

I'll make it worth every minute!

Pearson

THANK YOU!!

For spending 5 hours on Friday with me.

I'll make it worth every minute!

The next 5 hours will be highly educational

THANK YOU!!

For spending 5 hours on Friday with me.

I'll make it worth every minute!

The next 5 hours will be highly educational

…and a whole lot of fun, too

SEGMENT 1

Teaser

SEGMENT 2

ML Crash Course

SEGMENT 1

Teaser

SEGMENT 3

From ML to Deep Learning and Transformers

SEGMENT 2

ML Crash Course

SEGMENT 1

Teaser

SEGMENT 1
Teaser

SEGMENT 2
ML Crash Course

SEGMENT 3
From ML to Deep Learning and Transformers

SEGMENT 4
Time to be an AI Scientist

SEGMENT 1
Teaser

SEGMENT 2
ML Crash Course

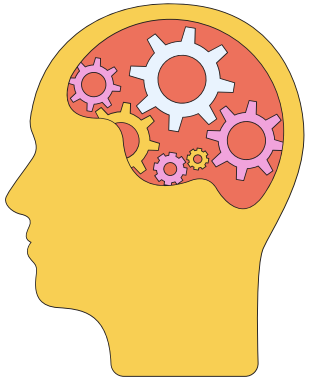
SEGMENT 3
From ML to Deep Learning and Transformers

SEGMENT 4
Time to be an AI Scientist

SEGMENT 5
Career Trajectories in AI Data Science

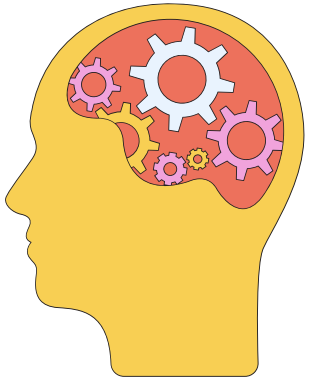
What you will leave with

Foundational knowledge



What you will leave with

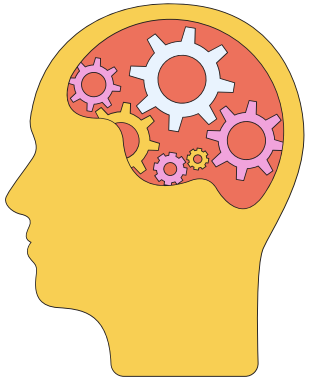
Foundational knowledge



Tools and Techniques to apply to your business

What you will leave with

Foundational knowledge



Tools and Techniques to apply to your business

Positioned to make your career decision

Setting you up to get the most out of the training

- Bring up the code in Jupyter Lab from the course repo - I've posted a link in the chat
- After the training, run the code yourself, confirm the results and reach out with questions
- Ask me questions relentlessly! That's what I'm here for.

And…

Setting you up to get the most out of the training

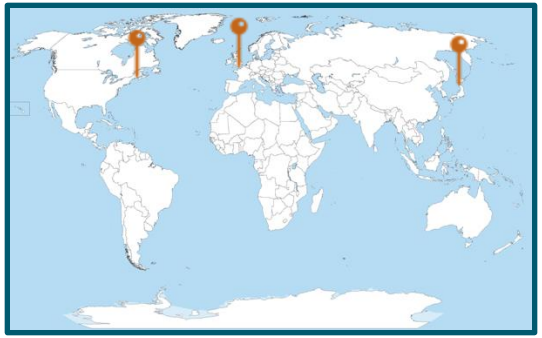
- Bring up the code in Jupyter Lab from the course repo - I've posted a link in the chat
- After the training, run the code yourself, confirm the results and reach out with questions
- Ask me questions relentlessly! That's what I'm here for.

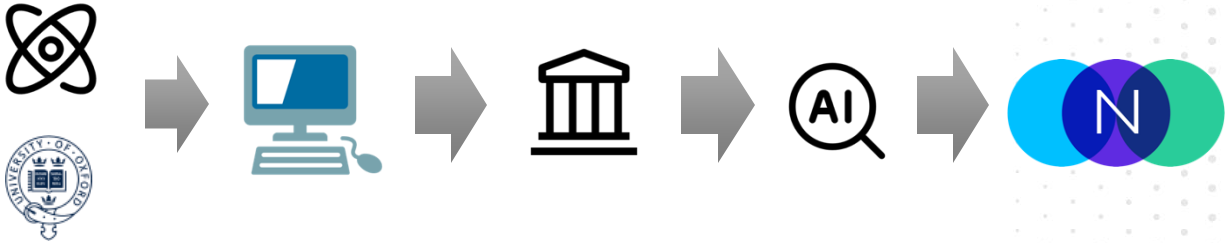
And…

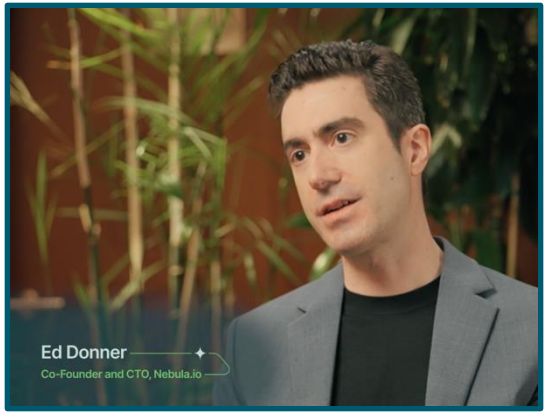
I LOVE EMOJIS!!! ❤️ 👍

Introducing me..

Co-Founder and CTO of Nebula.io; 20-year career as Technology and Data Science leader, entrepreneur and coach







Pearson

Connect with me: https://www.linkedin.com/in/eddonner/

POLL: Introducing YOU

Please select your continent (select one)

- Africa
- Asia
- Europe
- North America
- South America
- Oceania / Australia
- …or Antarctica…

The Cherry On Top

- Should be valuable no matter your experience
  - Too hard? Get intuition, experiment afterwards
  - Too easy? Hang in there…
- Culminating in a MASSIVE project
  1. Autonomous Agentic AI
  2. Proprietary frontier LLM
  3. Powerful RAG pipeline
- And a final treat at the end

NOW – let’s get started!

Segment 1: TEASER

A Brief History of AI

1997: Deep Blue defeats Garry Kasparov

2011: IBM Watson wins Jeopardy!

2016: AlphaGo beats Lee Sedol at Go

1950 1960 1970 1980 1990 2000 2010 2020

1950: Alan Turing publishes seminal paper "Computing Machinery and Intelligence," asking "Can machines think?"

1956: Dartmouth Conference, marking the birth of AI as a field

1958: The Perceptron is invented, an early artificial neural network

First AI Winter

Rise of expert systems, revival of interest in Neural Networks

Growth in Machine Learning including decision trees, Bayesian networks, statistical Natural Language Processing

Second AI winter

2006: Geoff Hinton kick-starts Deep Learning revolution

2012: AlexNet demonstrates the power of Deep Learning in image recognition

2016: Google Translate switches to using Deep Learning

Dizzying Rise of the Transformer

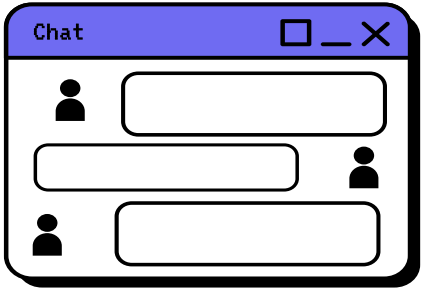
Google OpenAI

2017	2018	2019	2020	2021	2022	2023	2024
Google scientists publish seminal paper “Attention is All You need” proposing a new model architecture called the Transformer	GPT-1	GPT-2	GPT-3		RLHF and ChatGPT	GPT4	GPT4o o1

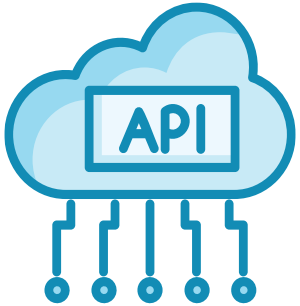
Pearson

3 ways to try them out

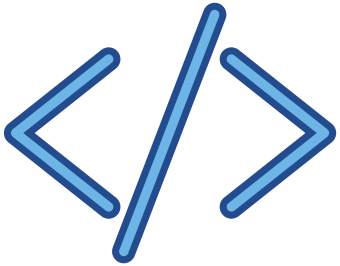
Chat UI



Cloud API



Directly executing Open-Source Models

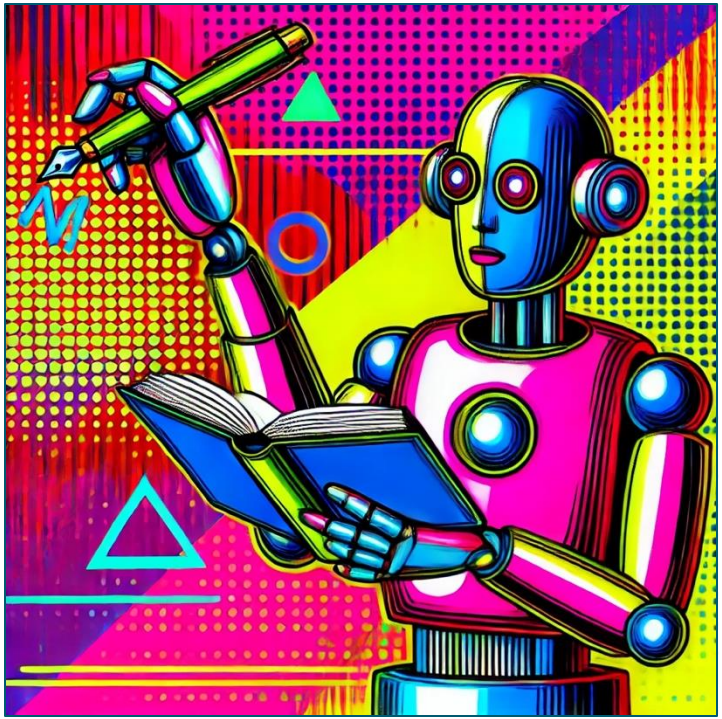


Time for ACTION!

Crash Course in ML

Machine Learning (ML)

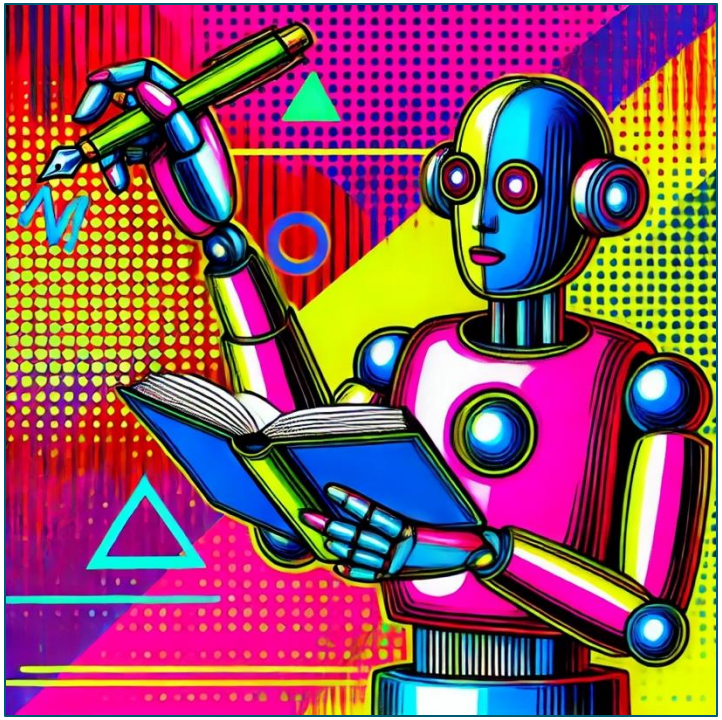
A branch of AI where algorithms learn from data to make predictions without being explicitly programmed.



Machine Learning (ML)

A branch of AI where algorithms learn from data to make predictions without being explicitly programmed.

These days, ML is most often used in the context of traditional ML, pre-Deep Learning & LLMs

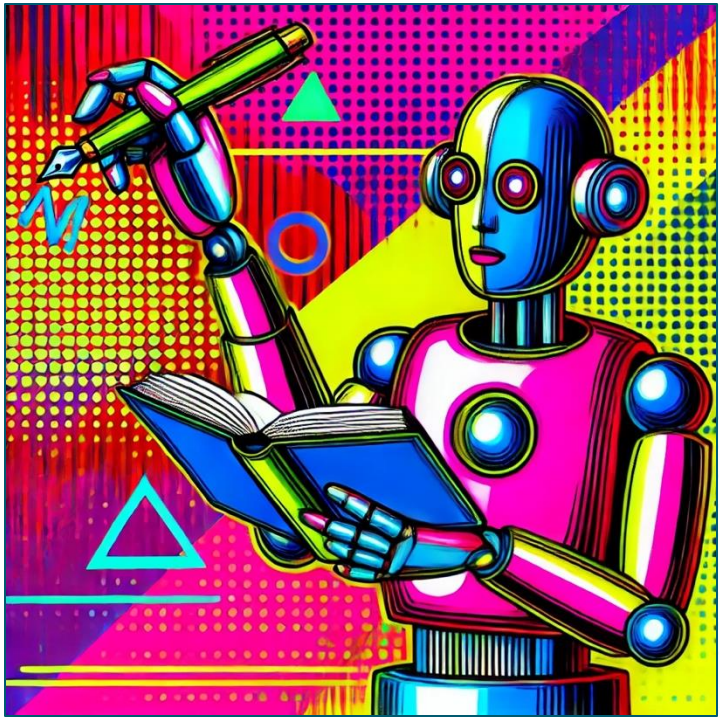


Machine Learning (ML)

A branch of AI where algorithms learn from data to make predictions without being explicitly programmed.

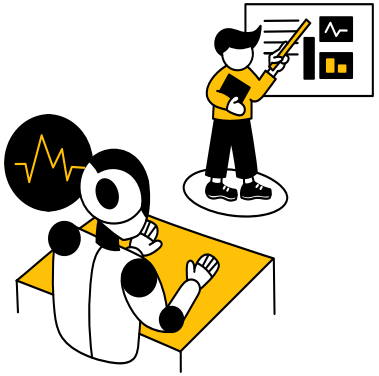
These days, ML is most often used in the context of traditional ML, pre-Deep Learning & LLMs

It's still crucial to understand core ML principles to be an LLM practitioner



Three flavors of ML

Three flavors of ML

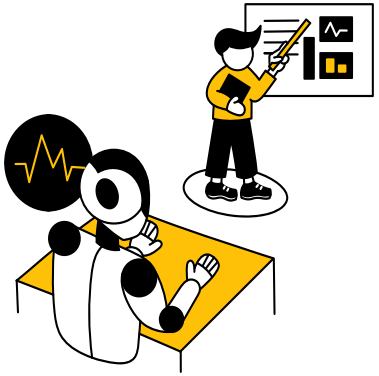


Supervised Learning
Learning from labeled data to make predictions

E.g., predicting house prices based on size, location, condition

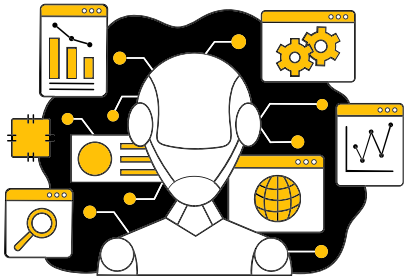
Pearson

Three flavors of ML



Supervised Learning
Learning from labeled data to make predictions

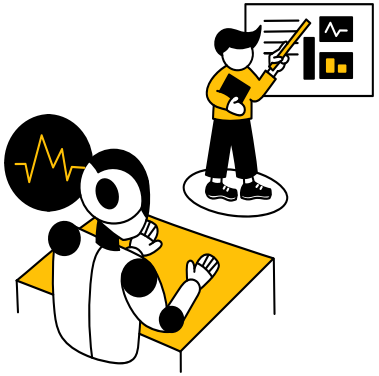
E.g., predicting house prices based on size, location, condition



Unsupervised Learning
Identifying patterns in unlabeled data

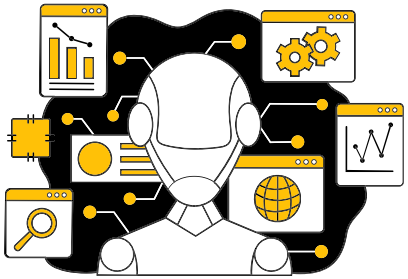
E.g., grouping customers into segments based on purchases

Three flavors of ML



Supervised Learning
Learning from labeled data to make predictions

E.g., predicting house prices based on size, location, condition



Unsupervised Learning
Identifying patterns in unlabeled data

E.g., grouping customers into segments based on purchases

Reinforcement Learning
Learning optimal actions from rewards or penalties

E.g., training a robot to navigate a maze

Core Machine Learning Tasks

Core Machine Learning Tasks

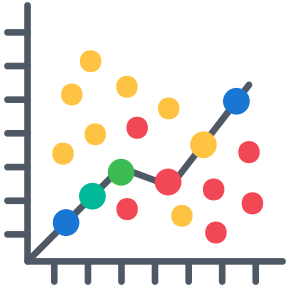
Classification
Assigns inputs to predefined categories

E.g., spam or not spam

Core Machine Learning Tasks

Classification
Assigns inputs to predefined categories

E.g., spam or not spam



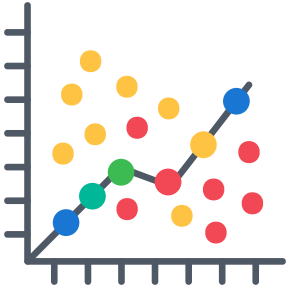
Regression
Predicts continuous values based on inputs

E.g., predicting house prices

Core Machine Learning Tasks

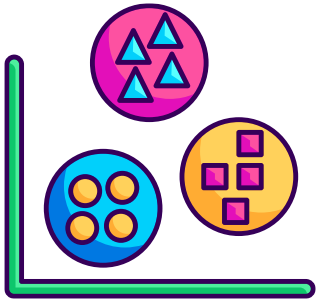
Classification
Assigns inputs to predefined categories

E.g., spam or not spam



Regression
Predicts continuous values based on inputs

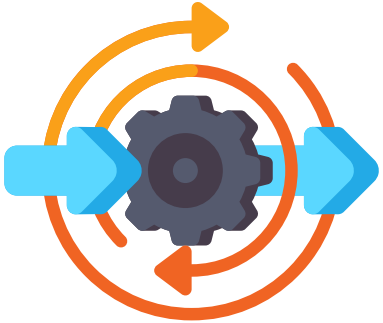
E.g., predicting house prices



Clustering
Groups inputs based on similarity

E.g., customer segmentation

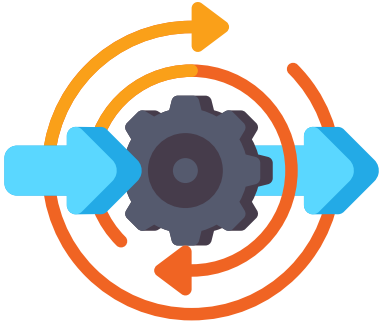
Training vs Inference



TRAINING

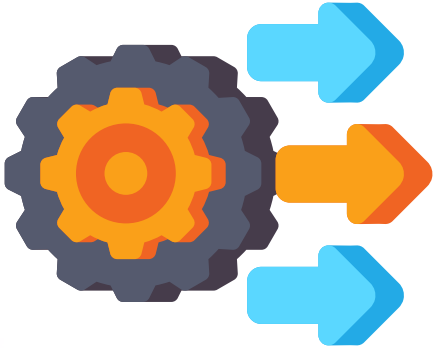
Using historical data to teach a model to make predictions by adjusting its parameters to minimize errors

Training vs Inference



TRAINING

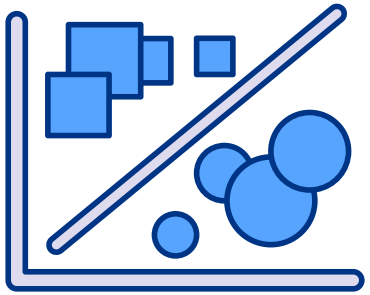
Using historical data to teach a model to make predictions by adjusting its parameters to minimize errors



INFERENCE

Applying a trained model to unseen data to make predictions by extrapolating from what it learned during training

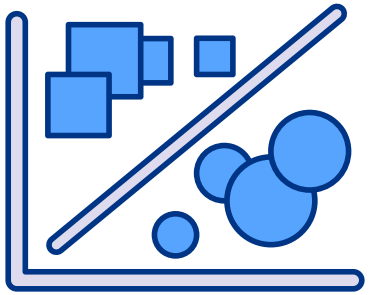
Generalizing vs Overfitting in ML



Generalizing

The ability of a model to perform well on new, unseen data by capturing underlying patterns

Generalizing vs Overfitting in ML



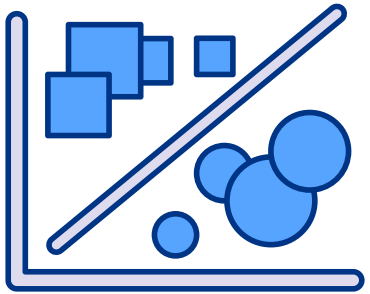
Generalizing

The ability of a model to perform well on new, unseen data by capturing underlying patterns

Overfitting

When a model mimics the training data too closely, including noise or irrelevant details, leading to poor performance on new inputs

Generalizing vs Overfitting in ML



Generalizing

The ability of a model to perform well on new, unseen data by capturing underlying patterns

Overfitting

When a model mimics the training data too closely, including noise or irrelevant details, leading to poor performance on new inputs

The goal of Machine Learning is to GENERALIZE to make accurate predictions on unseen data

Datasets are typically split into 3 buckets

Training Data

The majority of data, used by the model to learn patterns in order to generalize

Datasets are typically split into 3 buckets

Training Data

The majority of data, used by the model to learn patterns in order to generalize

Validation Data

Held-out data used occasionally to investigate how the model is performing on unseen data

Datasets are typically split into 3 buckets

Training Data

The majority of data, used by the model to learn patterns in order to generalize

Validation Data

Held-out data used occasionally to investigate how the model is performing on unseen data

Test Data

Held-out data used sparingly at the end of the process to confirm model performance on unseen data

Datasets are typically split into 3 buckets

Training Data

The majority of data, used by the model to learn patterns in order to generalize

Validation Data

Held-out data used occasionally to investigate how the model is performing on unseen data

Test Data

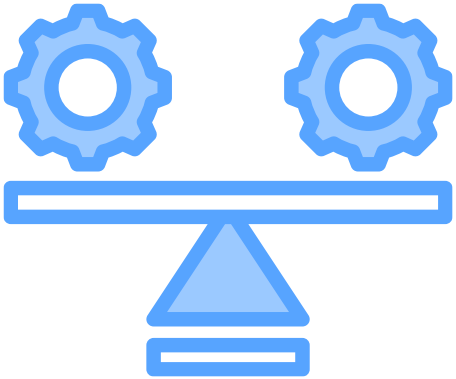
Held-out data used sparingly at the end of the process to confirm model performance on unseen data

"Dataset Contamination" occurs when validation or test data is accidentally provided to the model during training, so that it's no longer unseen

Internal and External settings

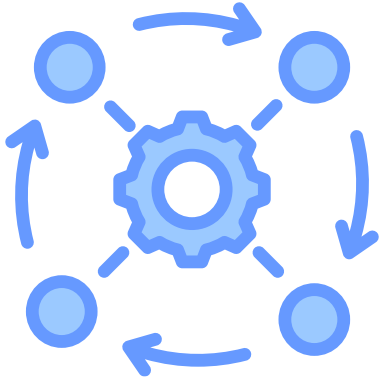
Model weights or **parameters**

The internal settings that control how a model makes its predictions; these are adjusted during training



Hyperparameters

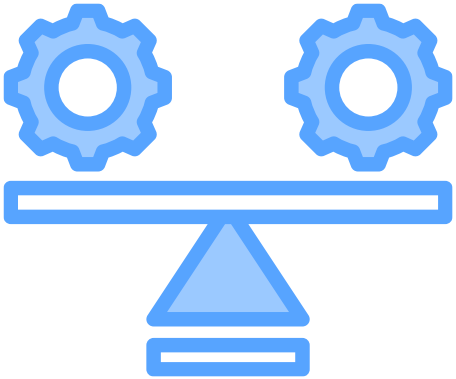
The external settings that are configured before training which affects the training process



Internal and External settings

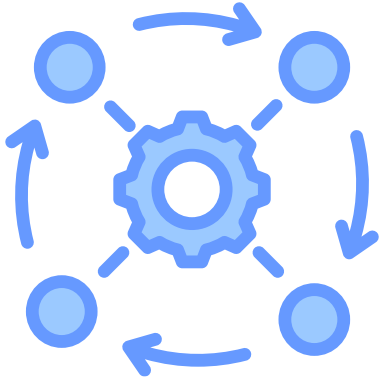
Model weights or **parameters**

The internal settings that control how a model makes its predictions; these are adjusted during training



Hyperparameters

The external settings that are configured before training which affects the training process



"Hyperparameter Optimization" is the process of experimenting with these external settings to achieve the best training results. In some ways, it's a fancy term for trial and error!

The traditional machine learning approach

The traditional machine learning approach

Feature Engineering

A domain expert identifies relevant factors from raw data that could influence the model's predictions

The traditional machine learning approach

Feature Engineering

A domain expert identifies relevant factors from raw data that could influence the model's predictions

Model selection and training

Choosing an appropriate algorithm, and using the training data to set the model parameters so that it can predict unseen data

The traditional machine learning approach

Feature Engineering

A domain expert identifies relevant factors from raw data that could influence the model's predictions

Model selection and training

Choosing an appropriate algorithm, and using the training data to set the model parameters so that it can predict unseen data

Model evaluation

Testing the trained model on unseen data to assess its performance

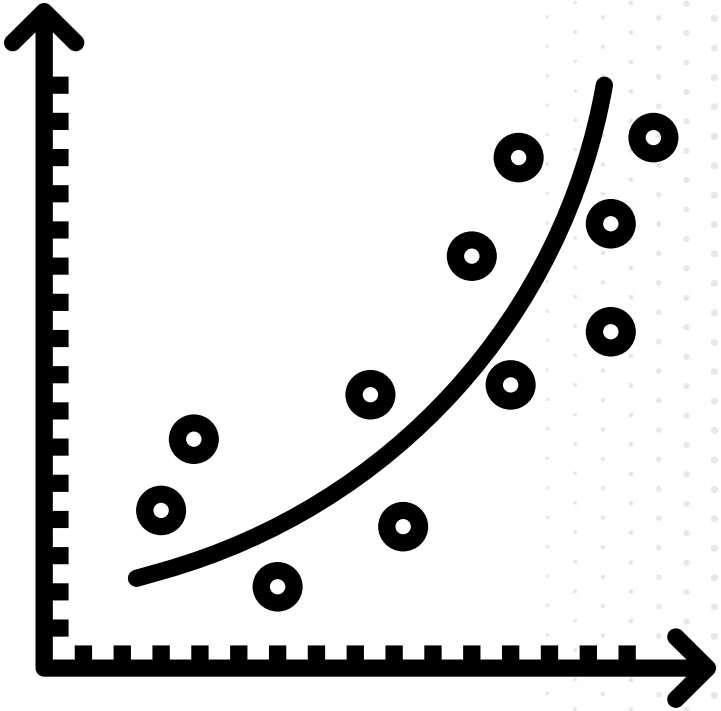
Introducing Linear Regression

- The workhorse of Traditional ML
- Pick features, then see whether your output can be expressed as a linear combination
- The classic formula is:
  $\hat{y} = w_0 + w_1x_1 + w_2x_2 + \ldots$

  where $\hat{y}$ is the predicted value, to be compared with y, the actual "ground truth"

  $w_0, w_1, w_2, \ldots$ are the model parameters / weights

  $x_1, x_2, \ldots$ are the features for a datapoint



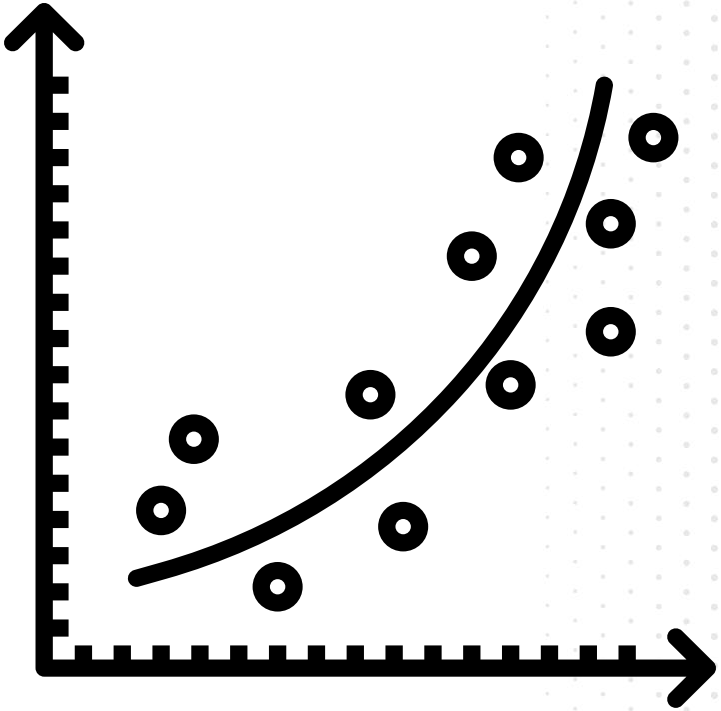
Introducing Linear Regression

- The workhorse of Traditional ML
- Pick features, then see whether your output can be expressed as a linear combination
- The classic formula is:
  $\hat{y} = w_0 + w_1x_1 + w_2x_2 + \ldots$

  where $\hat{y}$ is the predicted value, to be compared with y, the actual "ground truth"

  $w_0, w_1, w_2, \ldots$ are the model parameters / weights

  $x_1, x_2, \ldots$ are the features for a datapoint



Pearson

"Logistic Regression" is a close cousin used for Classification rather than Regression

Tools of the trade

Please give emojis when I mention ones you use:

NumPy

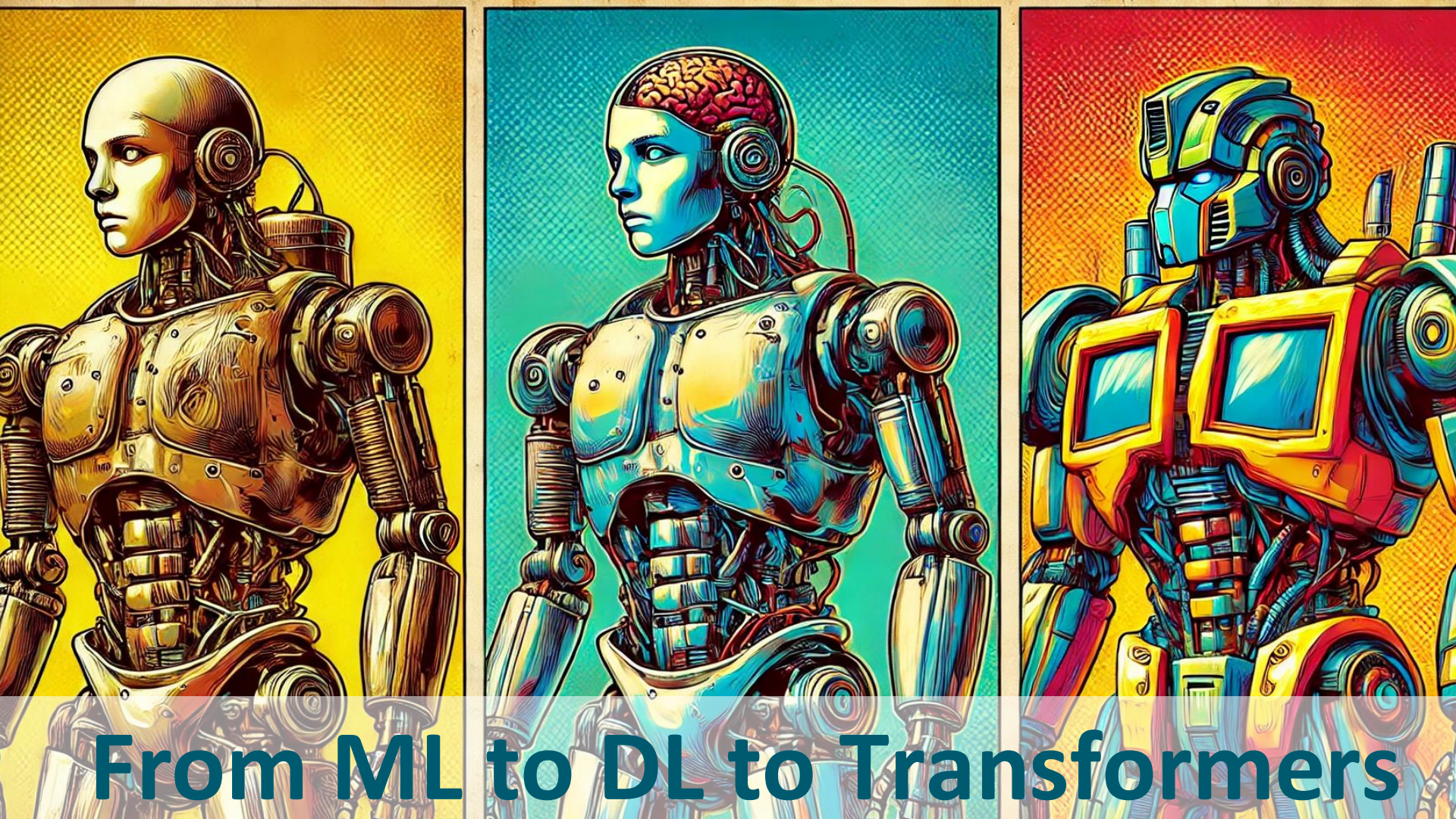
SciPy

pandas

matplotlib

scikit learn

To the lab!

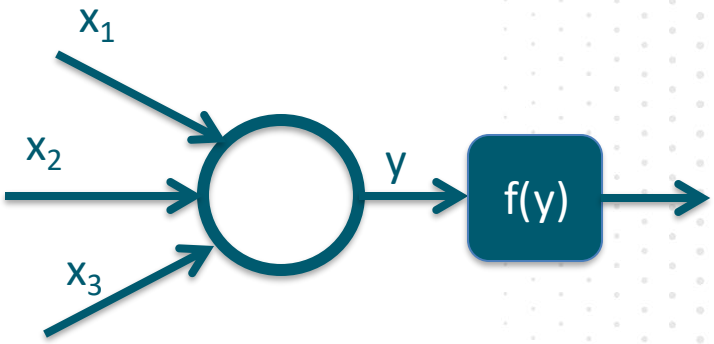


From ML to DL to Transformers

Unveiling the Neuron

- The fundamental building block of Deep Neural Networks
- Loosely inspired by a human neuron
- In many ways very similar to Linear Regression – but with one crucial difference:

  The output is passed through a non-linear "activation function"

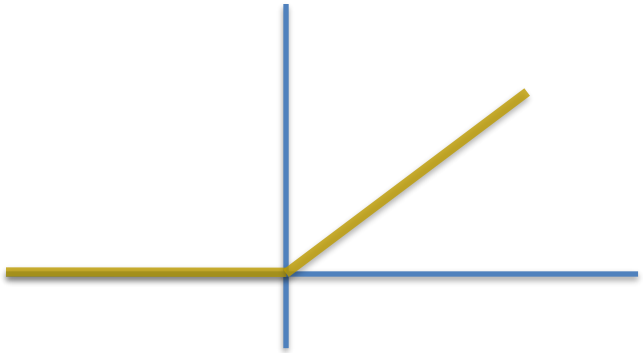


$y = w_1x_1 + w_2x_2 + w_3x_3 + b$

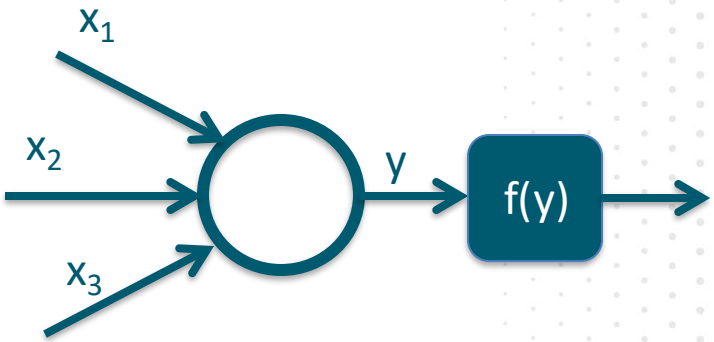
Neuron output = f(y)

Unveiling the Neuron

- The activation function can be as simple as max(y, 0) which is called ReLU



- If it weren't for the activation function you could connect as many neurons as you want, and you'd still just have linear regression

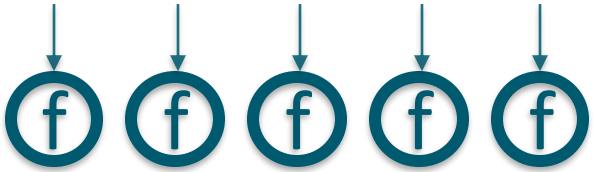


$y = w_1x_1 + w_2x_2 + w_3x_3 + b$

Neuron output = f(y)

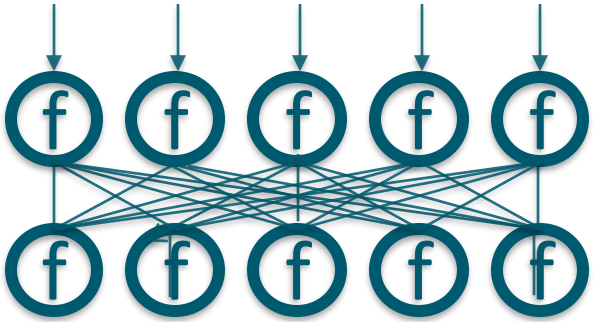
Deep Neural Networks: Stacking Layers of Neurons

- Powerful things start to happen when you connect the outputs of many Neurons – called a Layer of Neurons – to the input of another Layer:



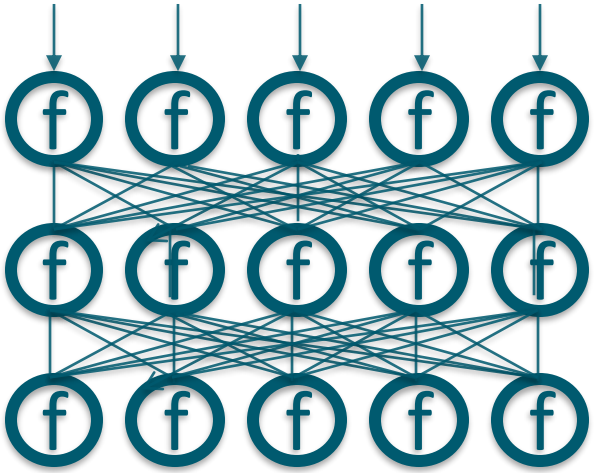
Deep Neural Networks: Stacking Layers of Neurons

- Powerful things start to happen when you connect the outputs of many Neurons – called a Layer of Neurons – to the input of another Layer:



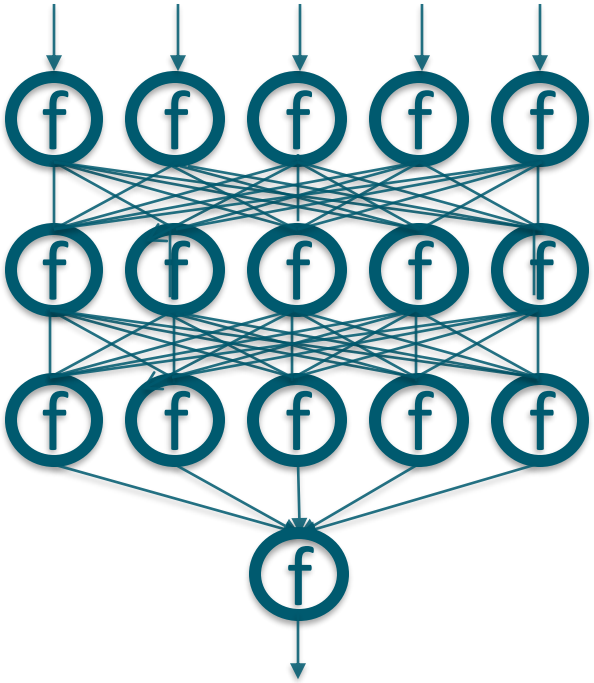
Deep Neural Networks: Stacking Layers of Neurons

- Powerful things start to happen when you connect the outputs of many Neurons – called a Layer of Neurons – to the input of another Layer:



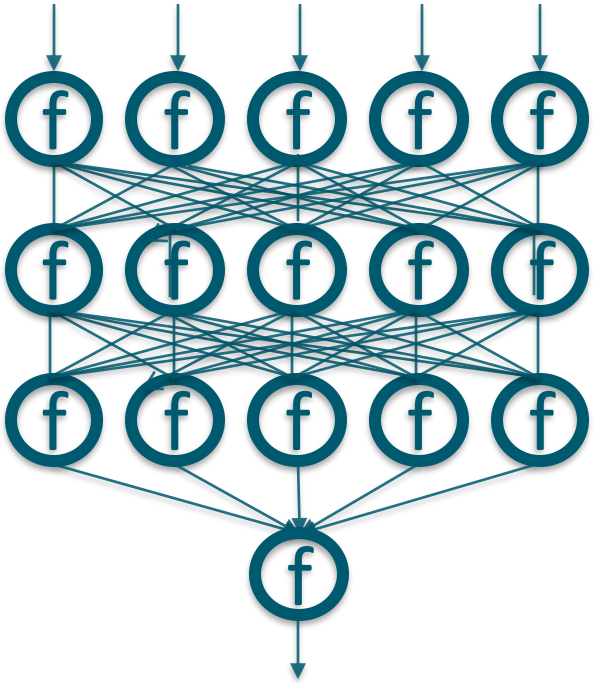
Deep Neural Networks: Stacking Layers of Neurons

- Powerful things start to happen when you connect the outputs of many Neurons – called a Layer of Neurons – to the input of another Layer:



Deep Neural Networks: Stacking Layers of Neurons

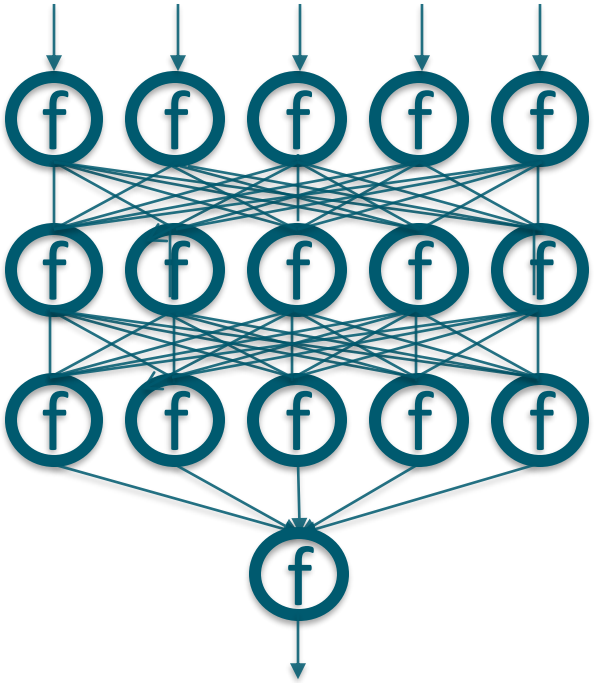
- Powerful things start to happen when you connect the outputs of many Neurons – called a Layer of Neurons – to the input of another Layer:



With enough neurons, you can vary the weights to approximate any function you want

Deep Neural Networks: Stacking Layers of Neurons

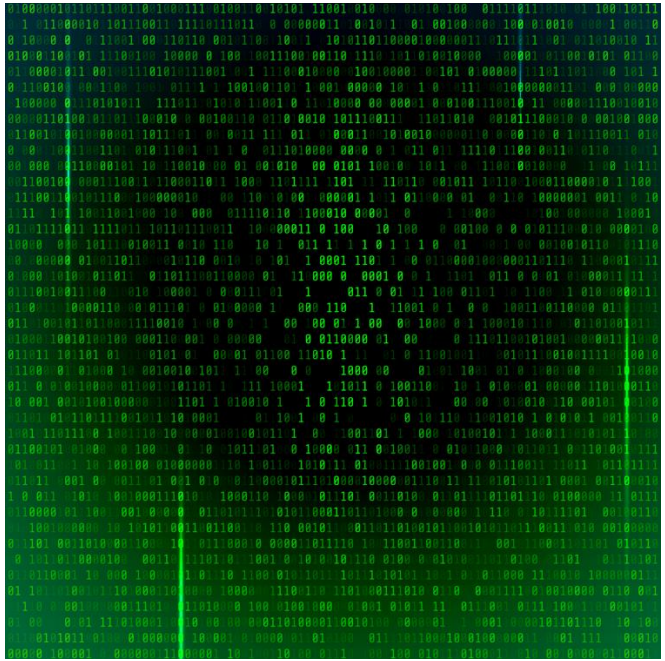
- Powerful things start to happen when you connect the outputs of many Neurons – called a Layer of Neurons – to the input of another Layer:



With enough neurons, you can vary the weights to approximate any function you want

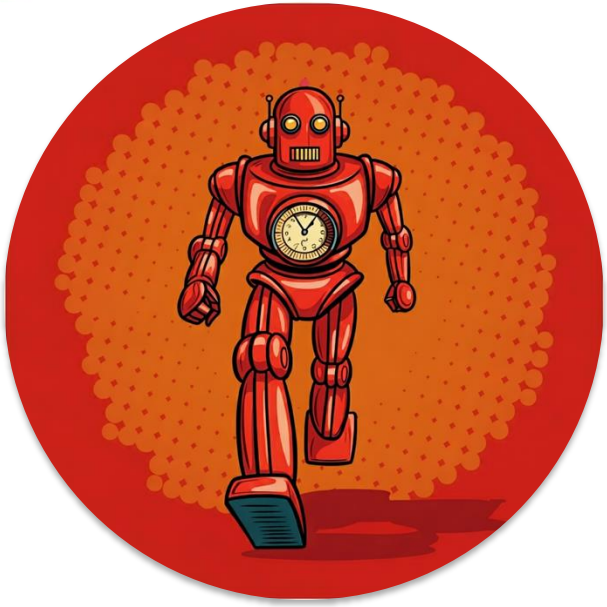
Training a Deep Neural Network involves tweaking the weights to achieve a desired outcome

Enter the Matrix



- Calculating the output of a layer involves multiplying all the weights in by all the inputs
- If the weights are written as a Matrix, this is one big Matrix multiplication
- So, much of Neural Network processing comes down to tons of matrix calcs
  - *Graphics cards are really good at this*

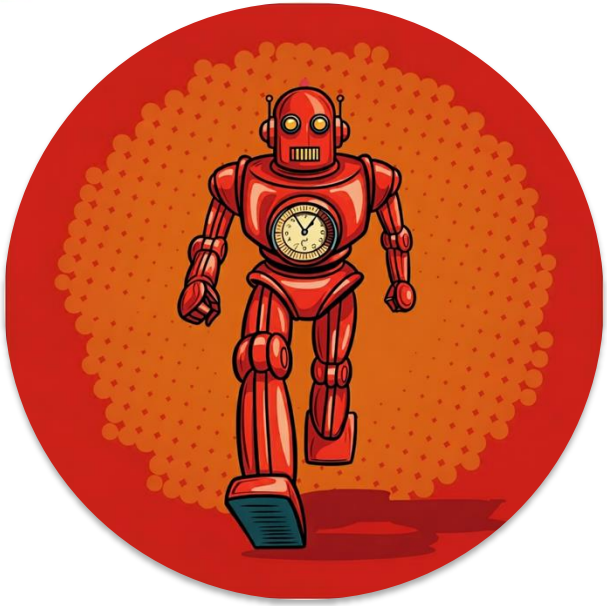
Forward Pass vs Backprop



The Forward Pass

Given inputs, calculate the output(s).
Used during inference.

Forward Pass vs Backprop



The Forward Pass

Given inputs, calculate the output(s).
Used during inference.

The Backward Pass

Use a technique called “Back Propagation” or “Backprop” to determine how the output is affected by a small change in weights.
Used during training (with the forward pass)

The Four Steps in Training

Forward pass

Predict the output given inputs

The Four Steps in Training

Forward pass

Predict the output given inputs

Loss calculation

How different was the prediction to the ground truth

The Four Steps in Training

Forward pass

Predict the output given inputs

Loss calculation

How different was the prediction to the ground truth

Backward pass

How should we tweak parameters to do better next time (the "gradients")

The Four Steps in Training

Forward pass

Predict the output given inputs

Loss calculation

How different was the prediction to the ground truth

Backward pass

How should we tweak parameters to do better next time (the "gradients")

Optimization

Update parameters a tiny step to do better next time

Architecture

- With more data and more compute, Neural Networks have become increasingly powerful
- There is another factor: ingenious techniques for designing Neural Networks
- In 2017, scientists from Google released a seminal paper inventing a new architecture – The Transformer

The Large Language Model (LLM)

Deep Neural Network trained to understand and generate language

Typically, Transformer Architecture

Takes “tokens” of several characters as inputs

Generates text a token at a time

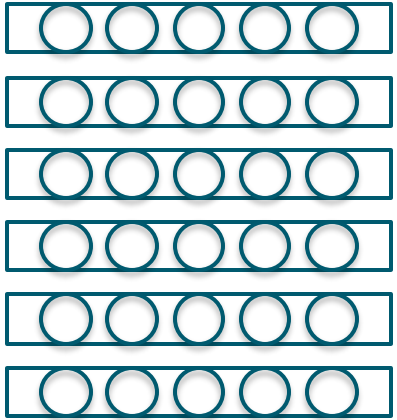
Can only look back across a limited number of tokens – the “context window”

1. Forward Pass

Input Prompt

We are living in a material <???>

LLM



Predicted Next Token

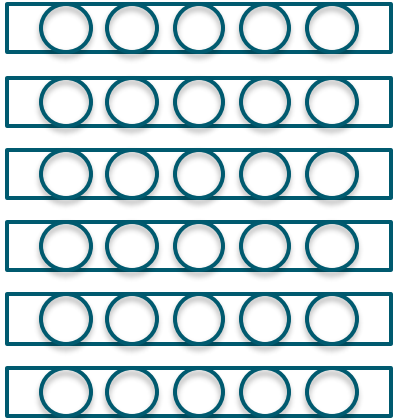
We are living in a material **place**

2. Loss Calculation

Input Prompt

We are living in a material <???>

LLM



Predicted Next Token

We are living in a material **place**

Actual Next Token

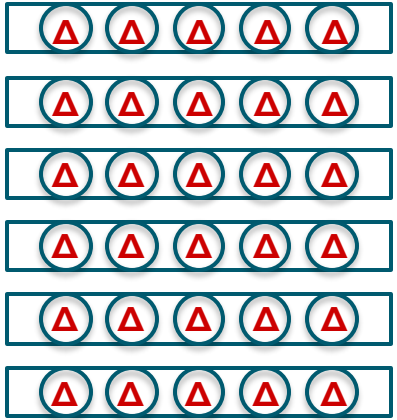
We are living in a material **world**

3. Backward Pass

Input Prompt

We are living in a material <???>

LLM



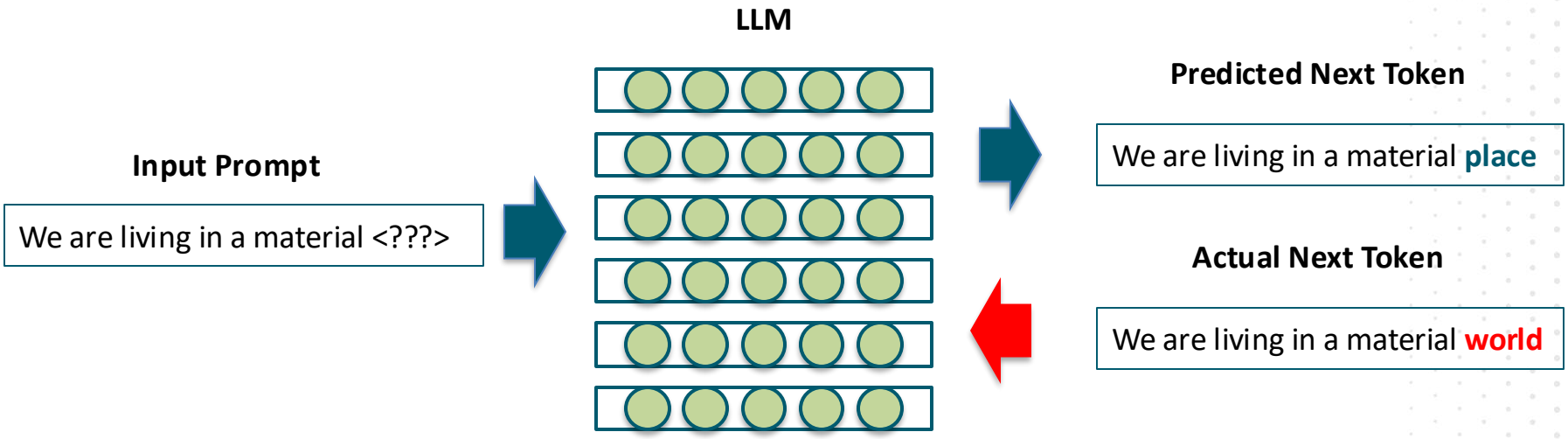
Predicted Next Token

We are living in a material **place**

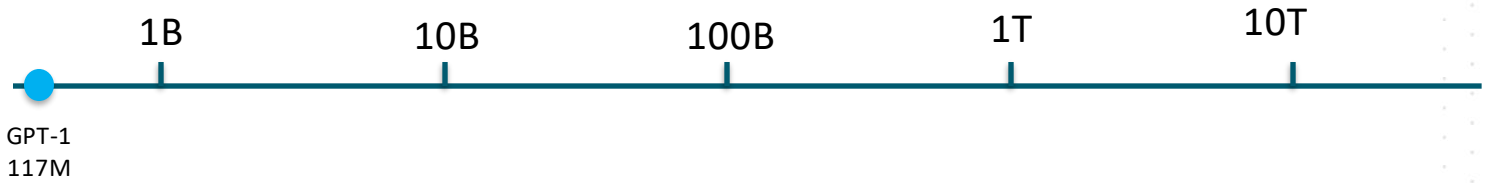
Actual Next Token

We are living in a material **world**

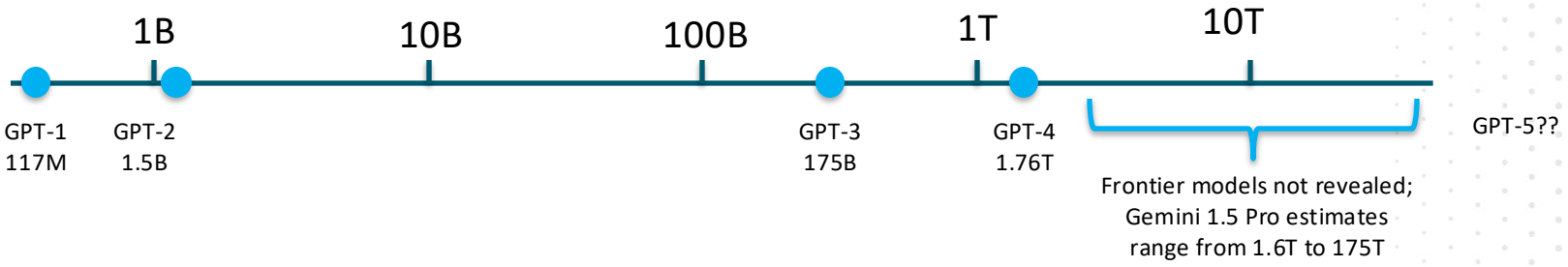
4. Optimization



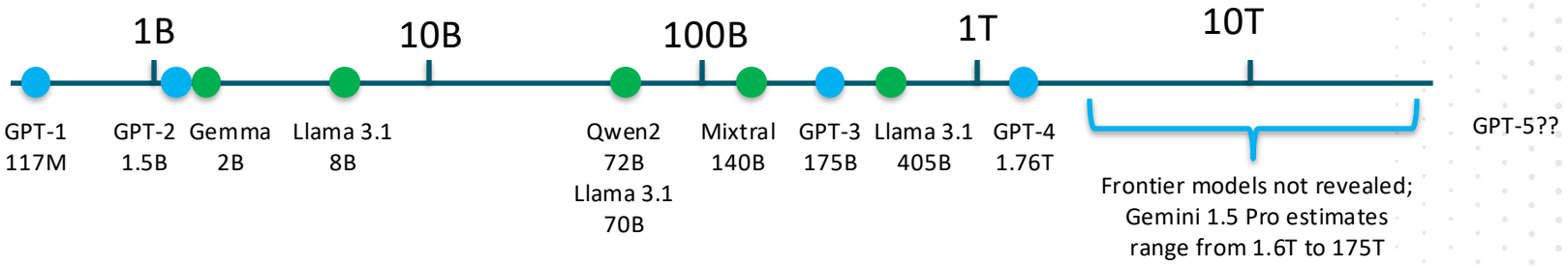
Number of parameters in LLMs (log scale)



Number of parameters in LLMs (log scale)



Number of parameters in LLMs (log scale)



A Fundamental Difference between ML and DL Activities

Machine Learning:

Features are manually selected by Data Scientists

As part of Feature Engineering, Data Scientists handpick tens or hundreds of features.

The model learns from data how to weigh up the features.

Data Scientists typically need to be domain experts.

A Fundamental Difference between ML and DL Activities

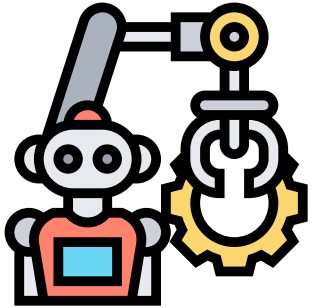
Machine Learning:

Features are manually selected by Data Scientists

As part of Feature Engineering, Data Scientists handpick tens or hundreds of features.

The model learns from data how to weigh up the features.

Data Scientists typically need to be domain experts.



Deep Learning:

The model adapts without explicit feature engineering

There can be millions or even billions of weights.

Data Scientists do not necessarily need to be domain experts.

Do you agree with this statement?

By 2028 we will have **Artificial Super Intelligence** in which AI models will **outperform** humans across a wide range of cognitive tasks

Back to the lab!

Becoming a Data Scientist

Tools of the trade

Please give emojis when I mention ones you use:

PyTorch

Hugging Face

LangChain

gradio

Weights & Biases

Google colab

Four Powerful Techniques for optimizing LLM performance

Four Powerful Techniques for optimizing LLM performance

Fine-Tuning

Four Powerful Techniques for optimizing LLM performance

Fine-Tuning

Multi-shot Prompting

Pearson

Four Powerful Techniques for optimizing LLM performance

Fine-Tuning

Multi-shot Prompting

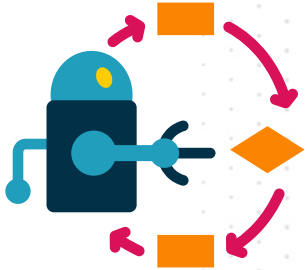
RAG

Four Powerful Techniques for optimizing LLM performance

Fine-Tuning

Multi-shot Prompting

RAG



Agentic Workflows

Four Powerful Techniques for optimizing LLM performance

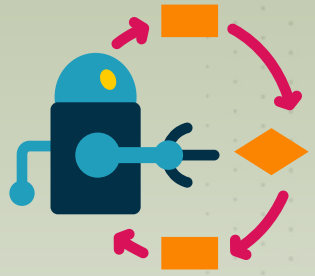
TRAINING TIME

Fine-Tuning

INFERENCE TIME

Multi-shot Prompting

RAG

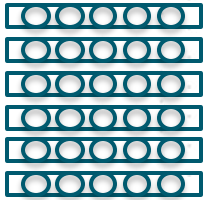


Agentic Workflows

The small idea behind RAG

Chat

Code



LLM

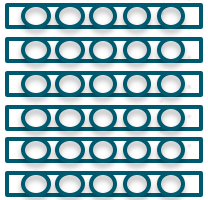
Knowledge Base

The small idea behind RAG

Chat

Question

Code

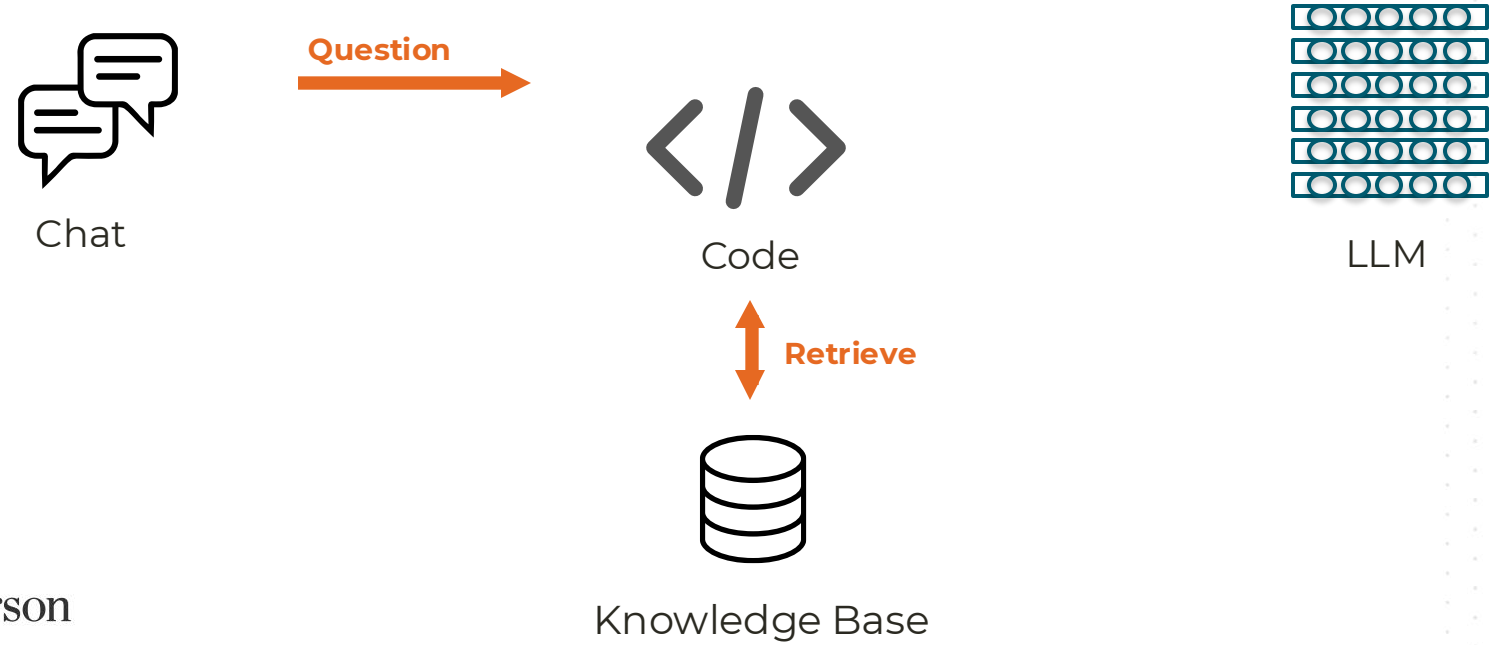


LLM

Knowledge Base

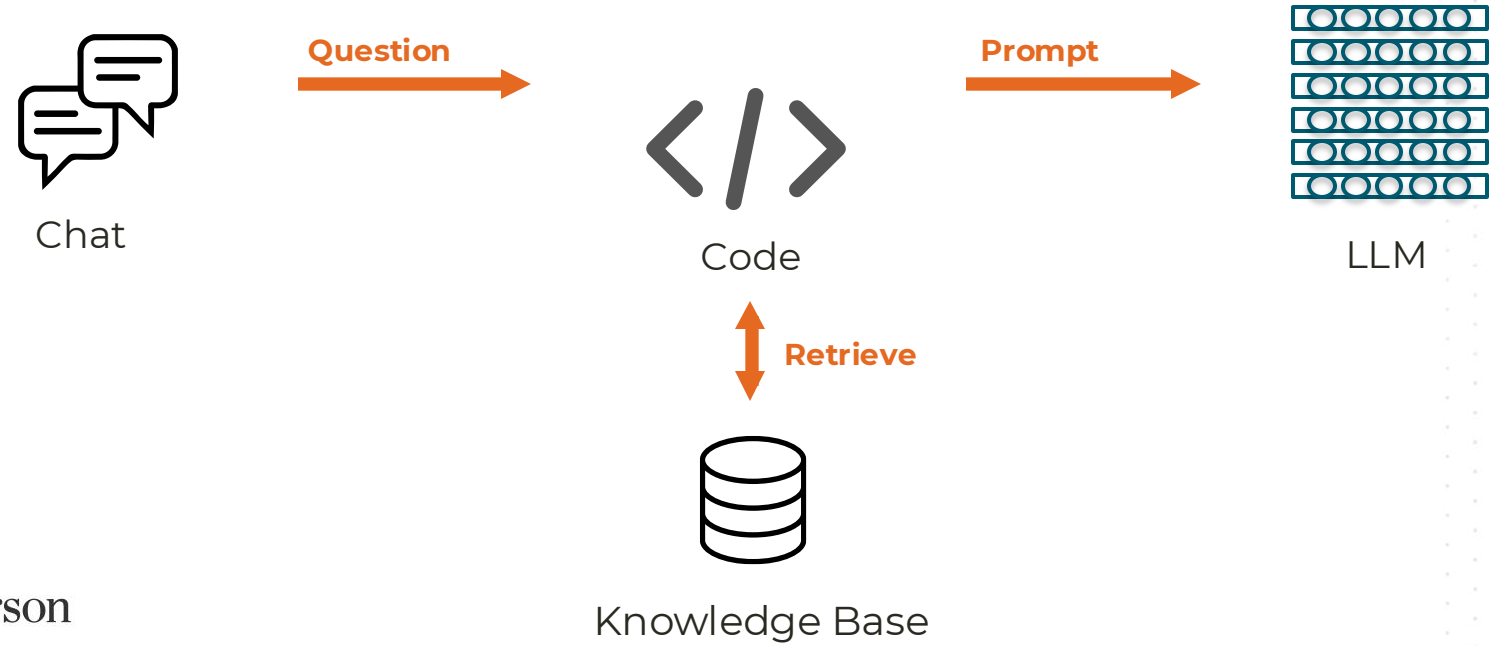
Pearson

The small idea behind RAG



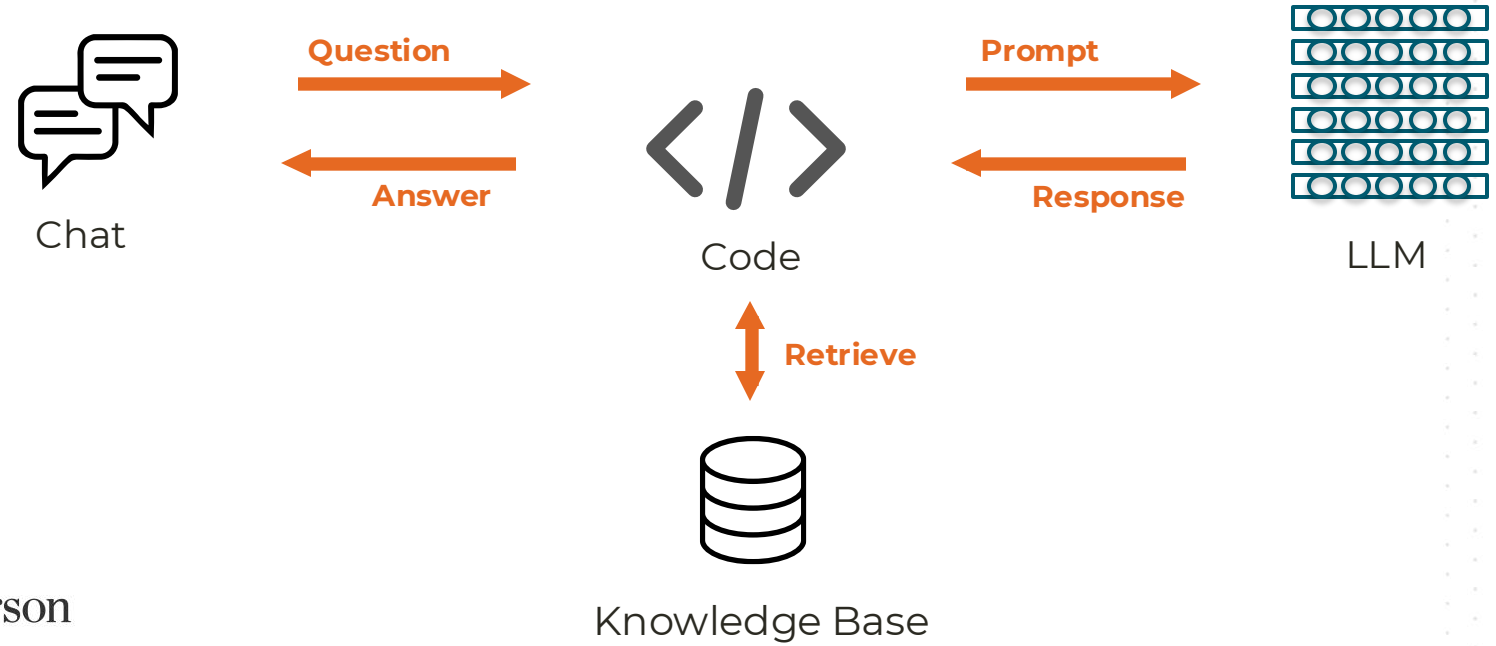
Pearson

The small idea behind RAG



Pearson

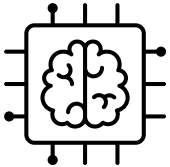
The small idea behind RAG



Pearson

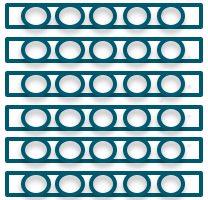
The big idea behind RAG

Encoding LLM



Chat

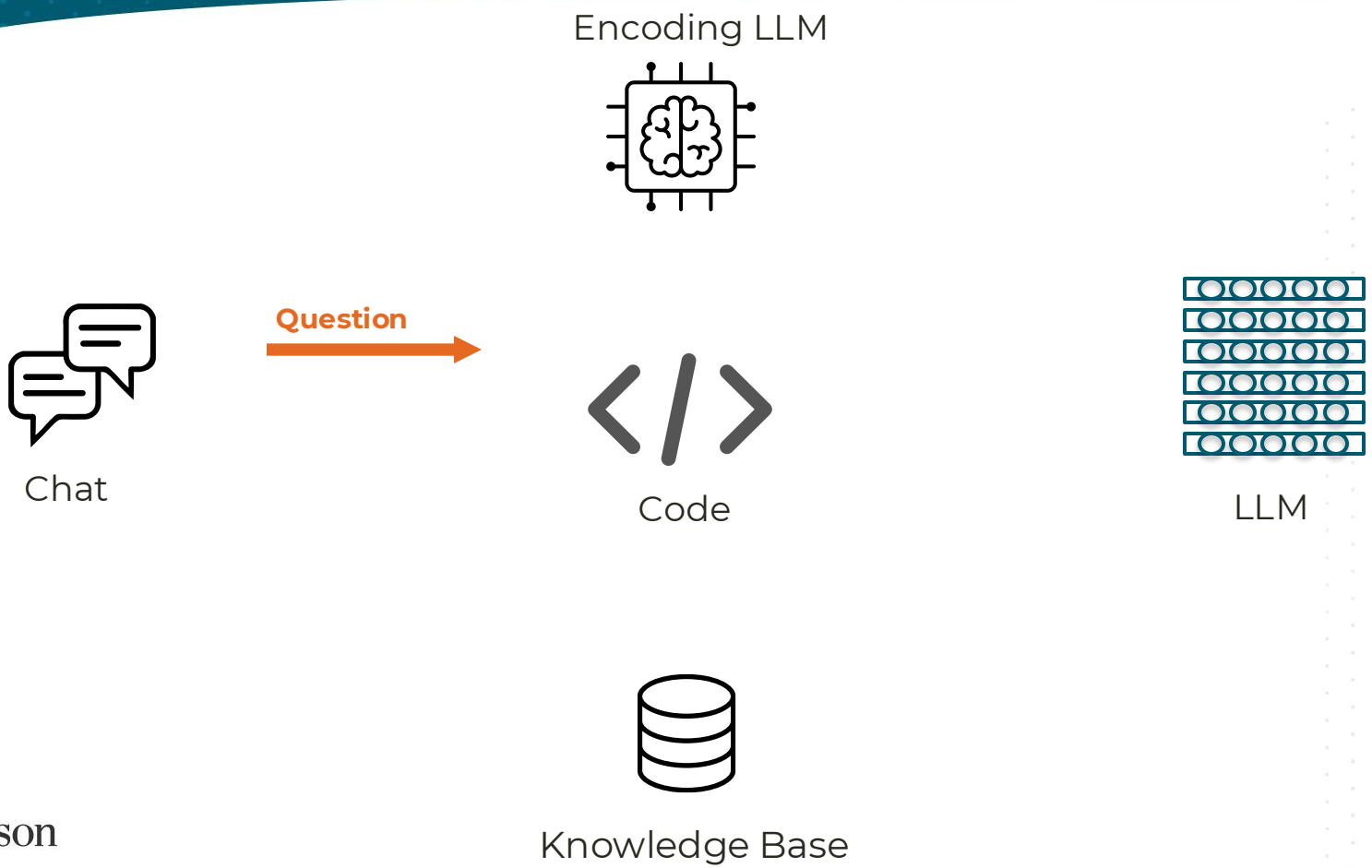
Code



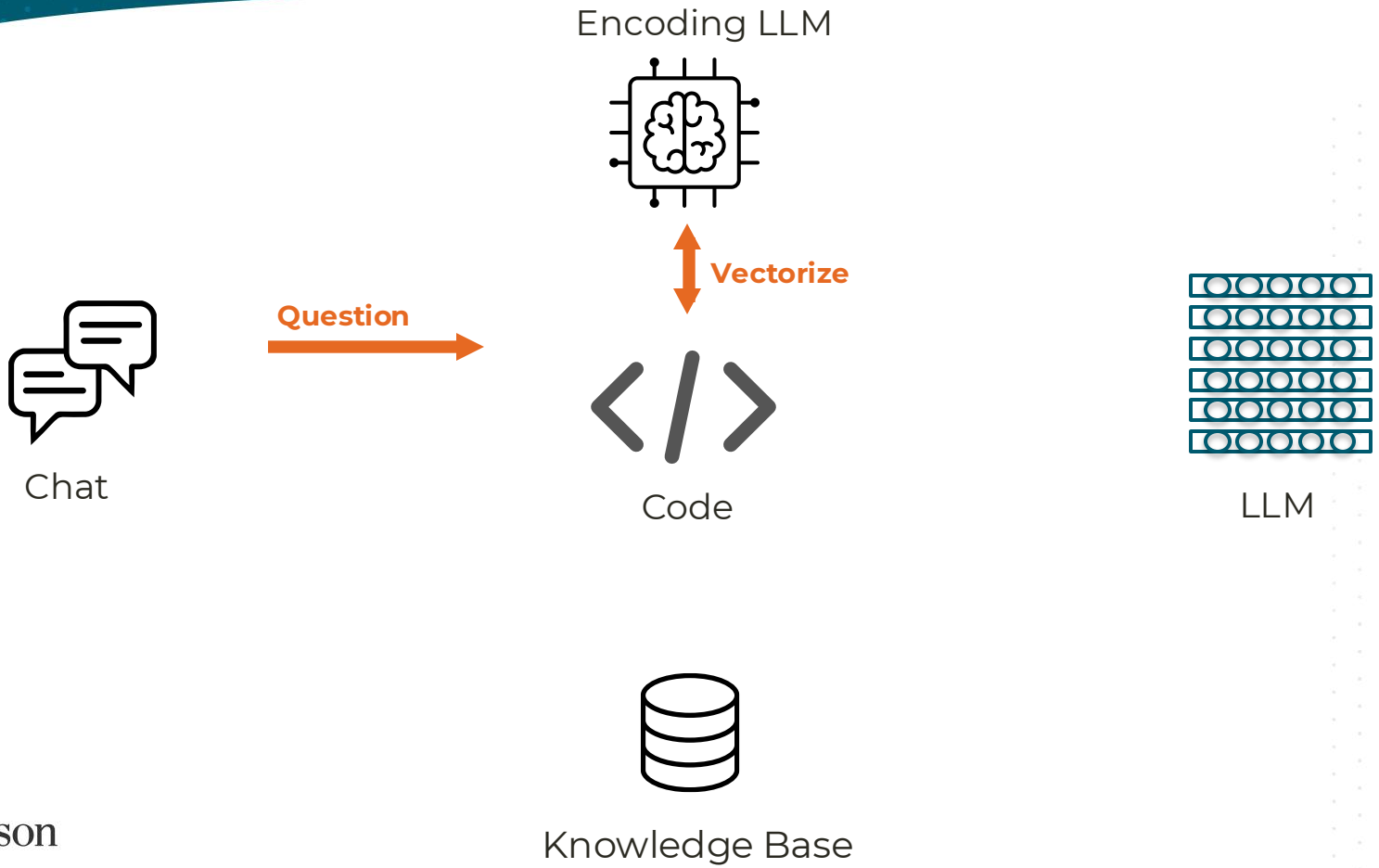
LLM

Knowledge Base

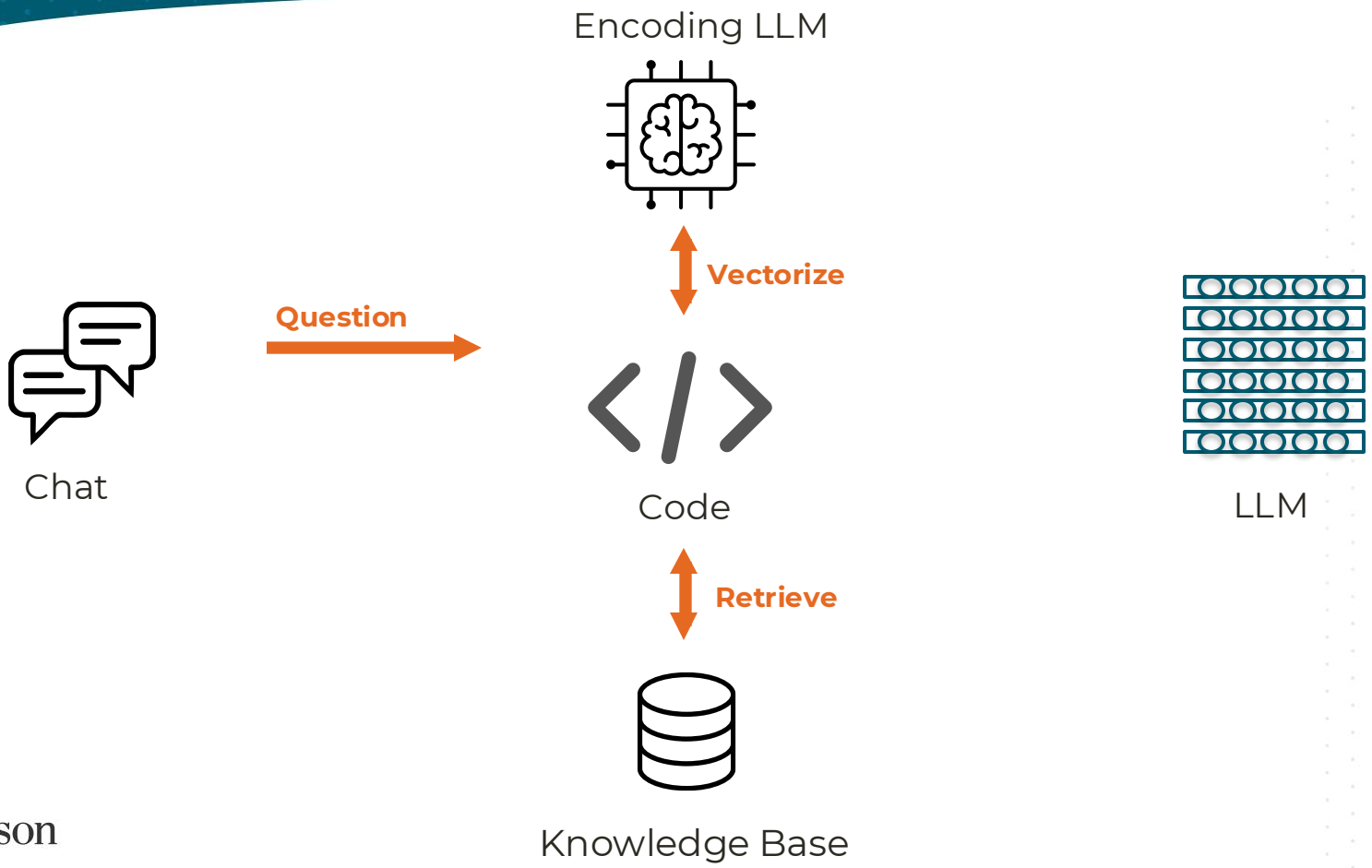
The big idea behind RAG



The big idea behind RAG

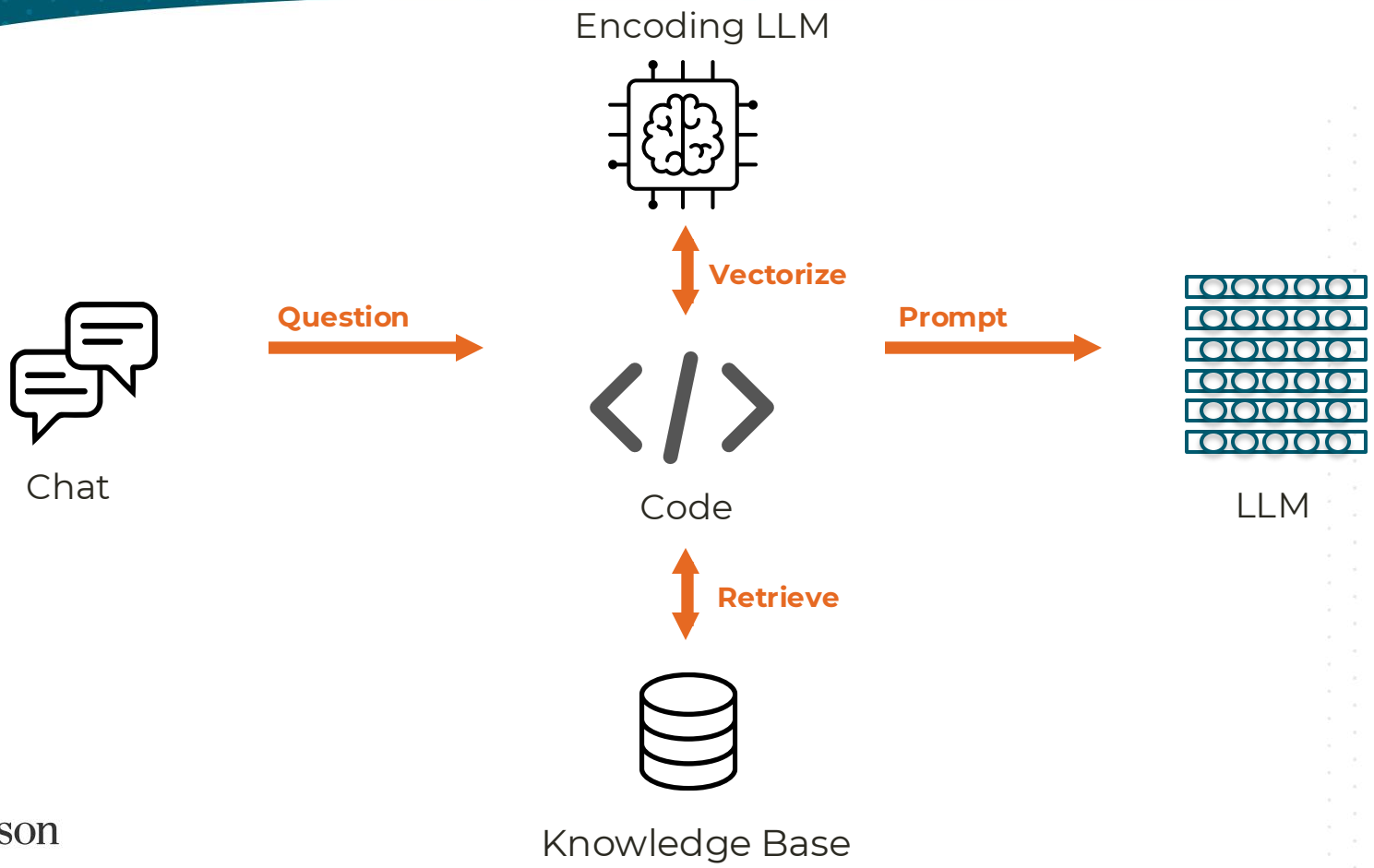


The big idea behind RAG



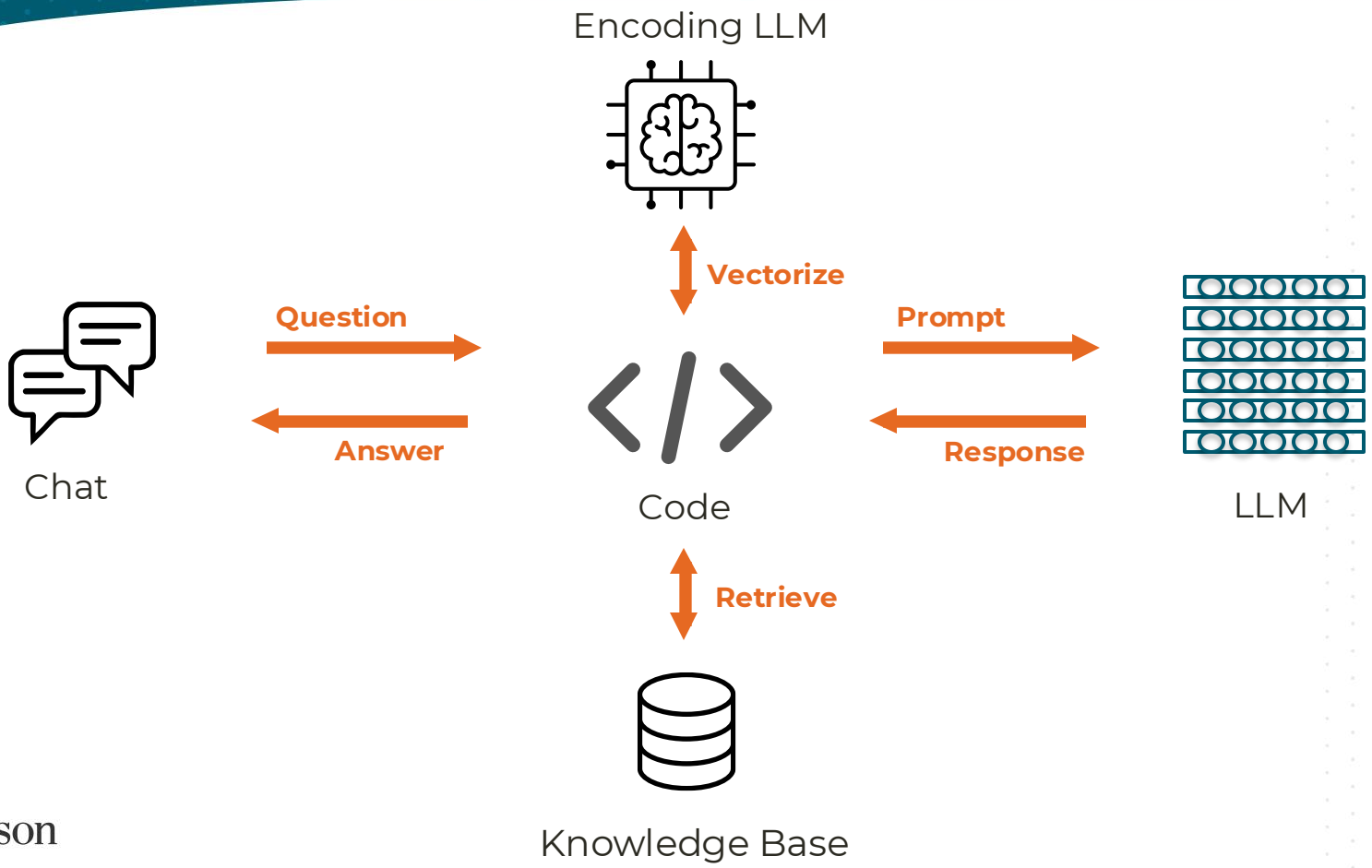
Pearson

The big idea behind RAG



Pearson

The big idea behind RAG



Pearson

QLoRA Fine-Tuning - motivation

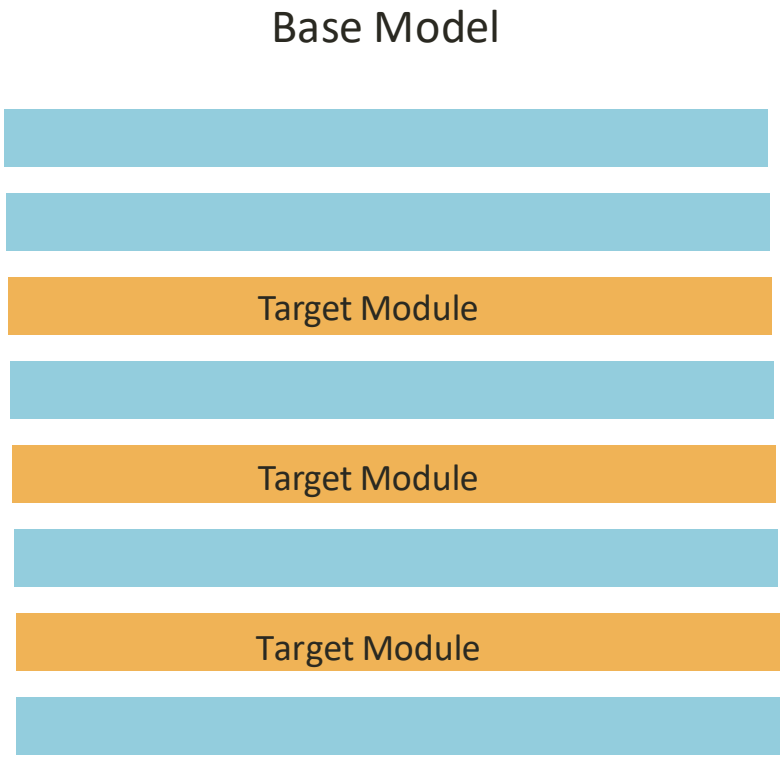
Take an existing “base” model like Llama 3.1, and train it to perform well at a specialized task, using “transfer learning”

But even ‘small’ models like Llama 3.1 8B have 8 billion parameters..

Step 1: Freeze the model weights

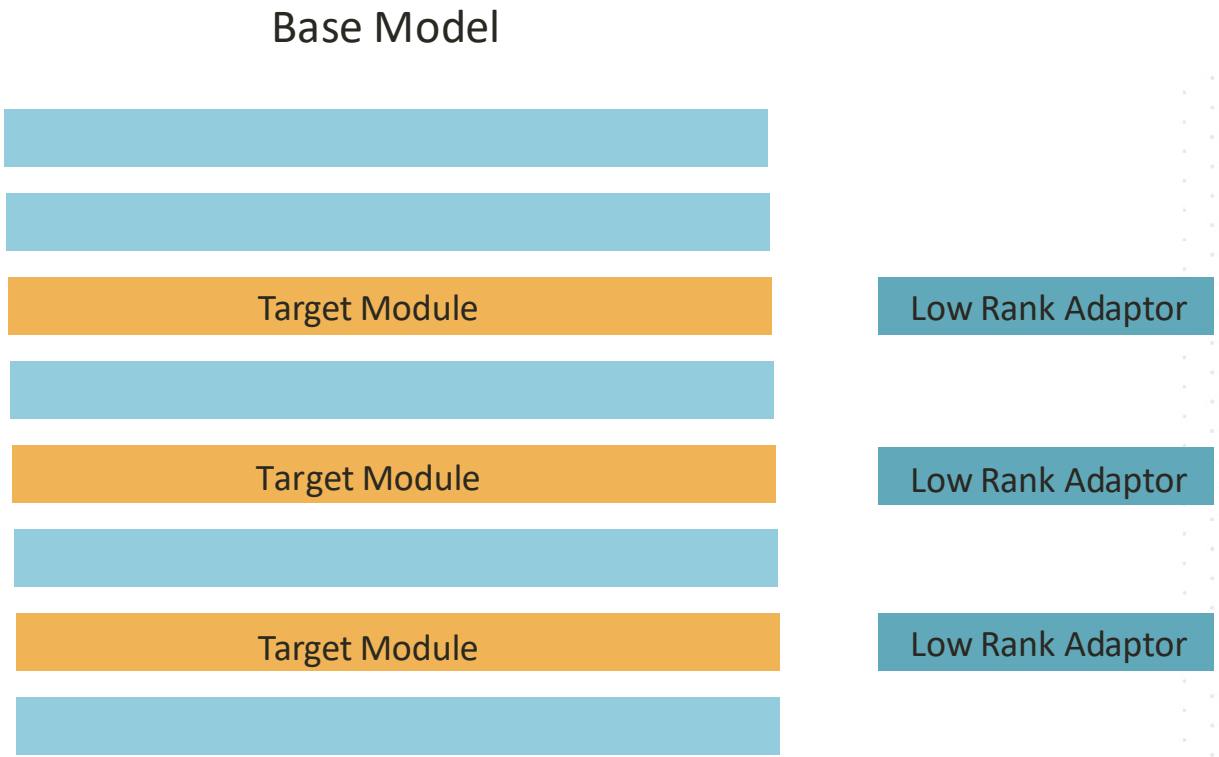
Base Model

Step 2: identify a subset of layers called “Target Modules”

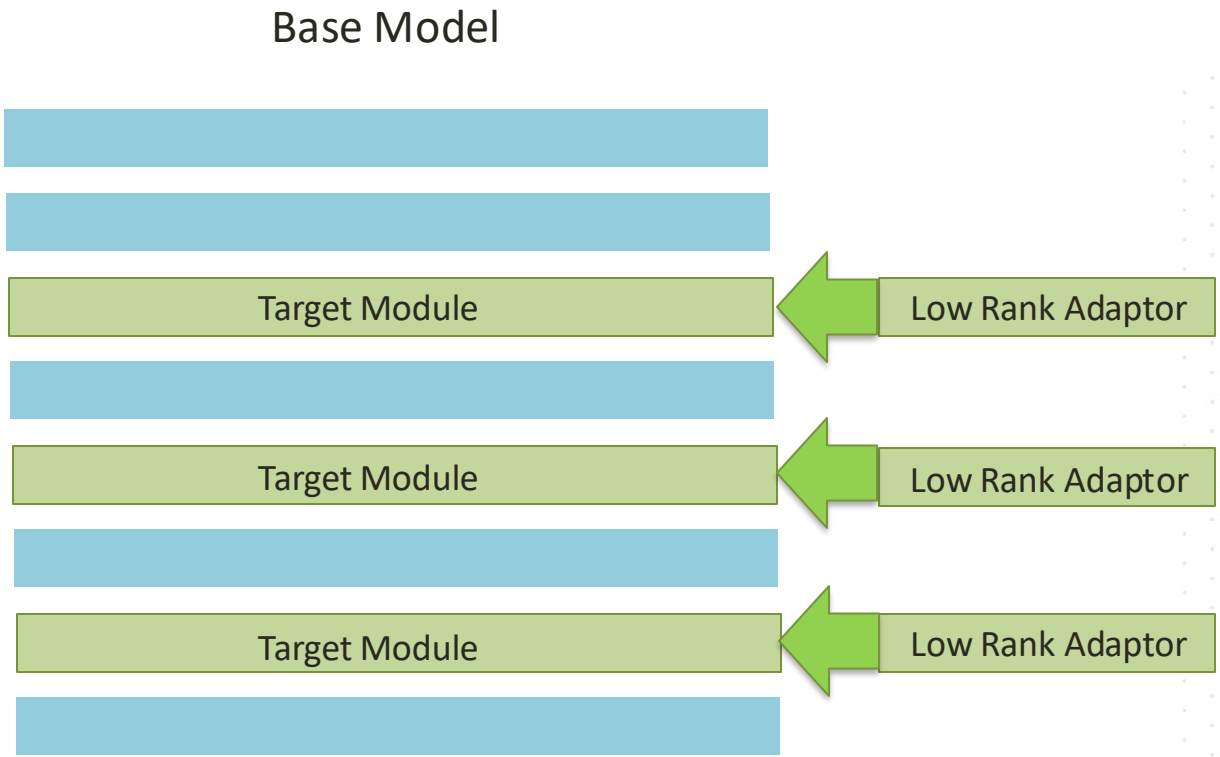


Pearson

Step 3: New "adaptor" matrices with lower dimensions

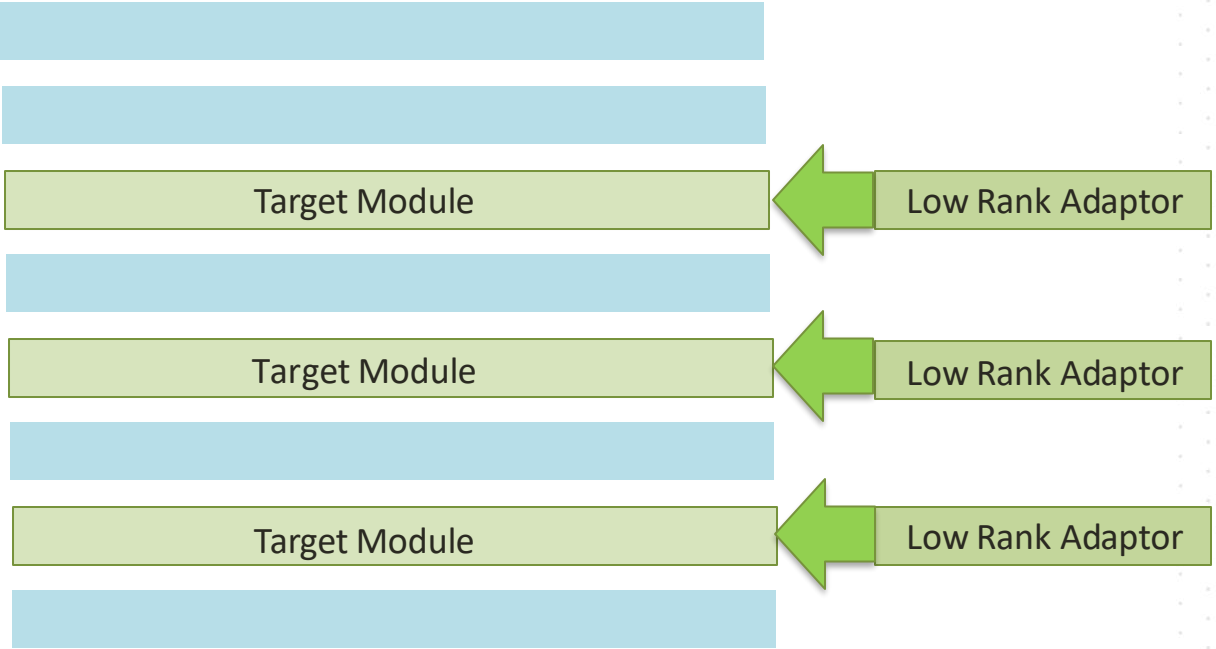


Step 4: Train these and apply them to the target modules



Step 5: Final trick: “Quantization”

Reduce the precision of the Base Model
down to 8 bits or even 4 bits



QLoRA = Quantized Low Rank Adaptation

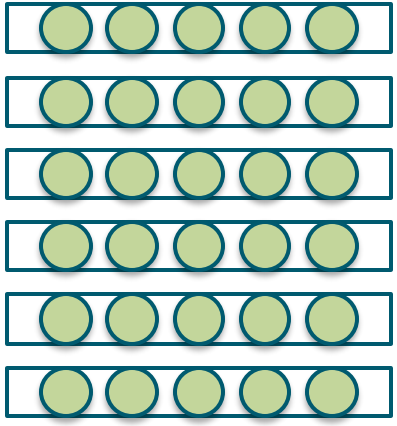
Pearson

Remember this slide on LLM training?

Input Prompt

We are living in a material <???>

LLM



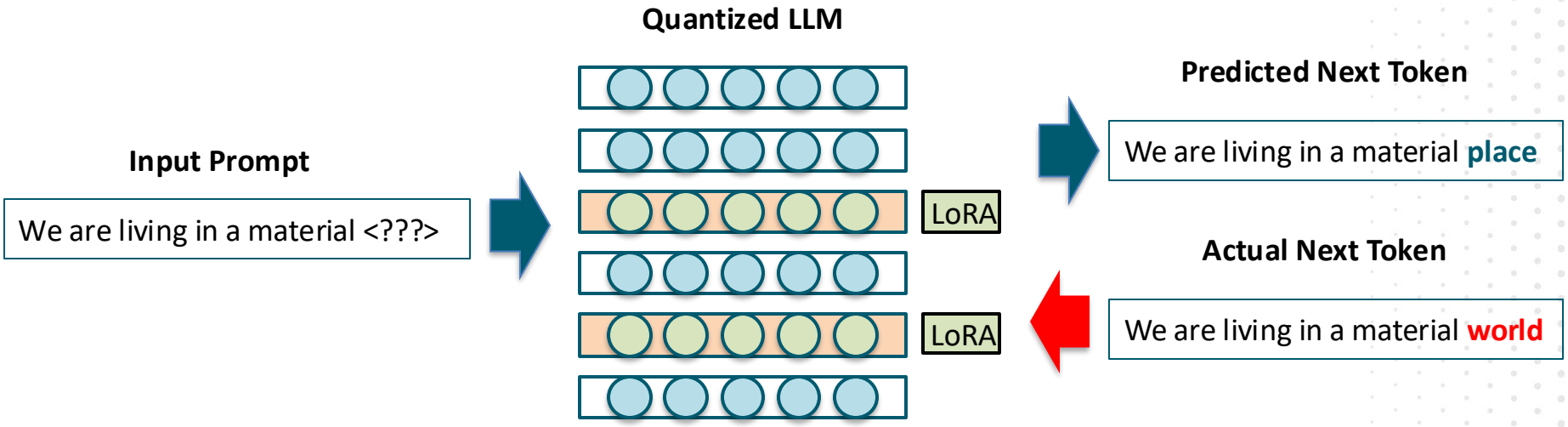
Predicted Next Token

We are living in a material **place**

Actual Next Token

We are living in a material **world**

…this is how it looks under QLoRA



The Hallmarks of an Agentic AI solution

Breaking a larger problem into smaller steps

Using Tools / Function Calling

An Environment where Agents collaborate

A Planning Agent that coordinates activities

Autonomy & Memory - existing beyond a chat with a human

Again to the lab!

Career Path into AI Data Science

Career Paths into AI Data Science

Ambiguity

Four key roles

Data Analyst

Data Engineer

Data Scientist

ML Engineer

Data Analyst

Responsibilities

- Generate actionable insights and reports
- Query databases, interpret data, provide analysis
- Understand Data Quality, quantity and completeness
- Generate data visualizations

Skills

- SQL
- Tools like Excel, Tableau, PowerBI
- Often: Python, R

Pearson

Data Engineer

Responsibilities

- Collect, process and store massive volumes of data
- Design and build databases and data pipelines
- Extract, Transform and Load data at scale

Skills

- ETL
- Distributed data processing and streaming frameworks like Spark, Hadoop, Kafka
- Databases and Cloud services

Pearson

Data Scientist

Responsibilities

- Solve business problems by learning from data
- R&D to invent new models
- Train models to learn from data

Skills

- Machine learning and deep learning theory
- Advanced python scripting and R
- Jupyter Notebooks, Numpy, Scipy, Pandas, sk-learn, Matplotlib
- Pytorch, Tensorflow, HuggingFace Transformers

Pearson

Machine Learning Engineer

Responsibilities

- Productionize Machine Learning Models
- Partner with Data Science to deploy and scale models
- Combines software engineering with Data Science

Skills

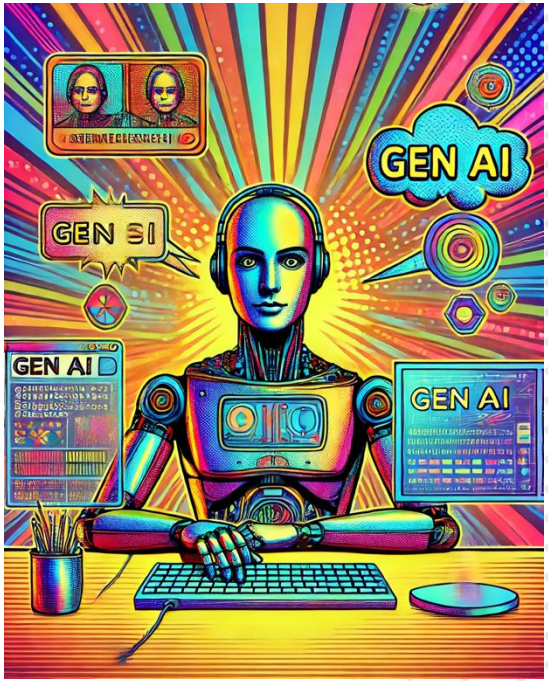
- Broad understanding of Data Science
- Software engineering / Devops: CICD, Cloud Deployments, Docker, Kubernetes, data pipeline frameworks
- Advanced Python

Pearson

But wait… there’s more!

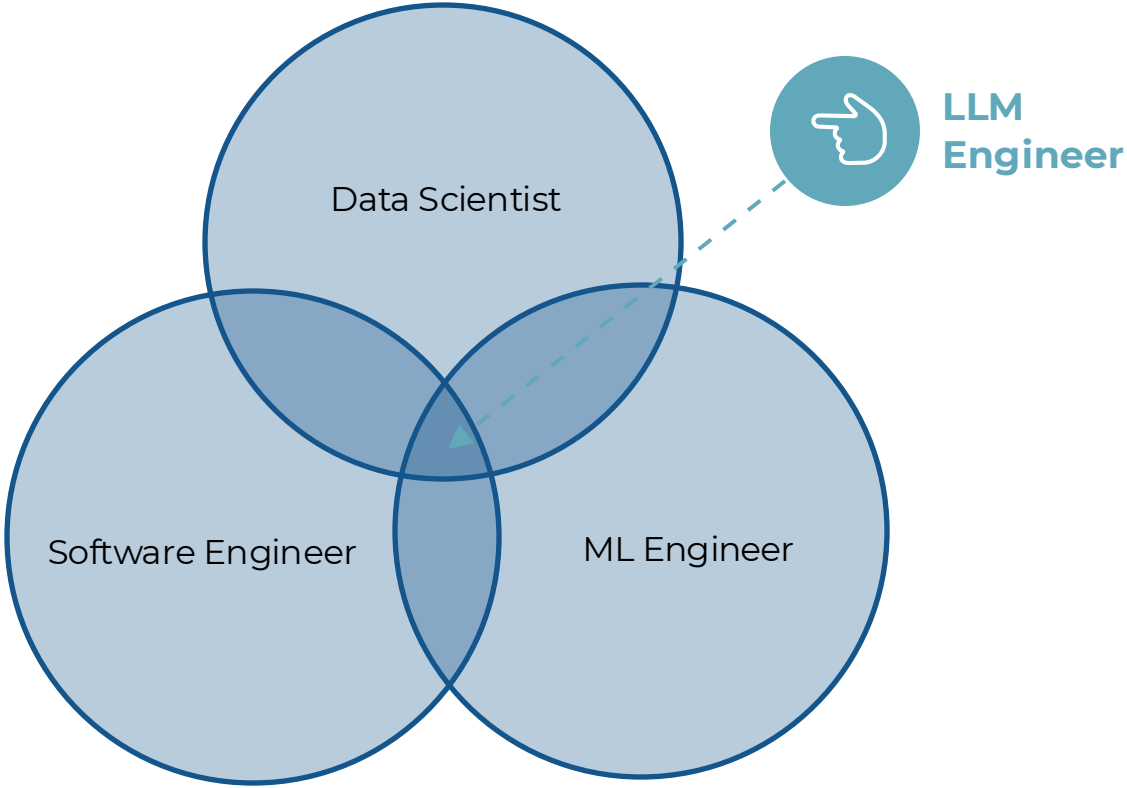
Data Architect

MLOps

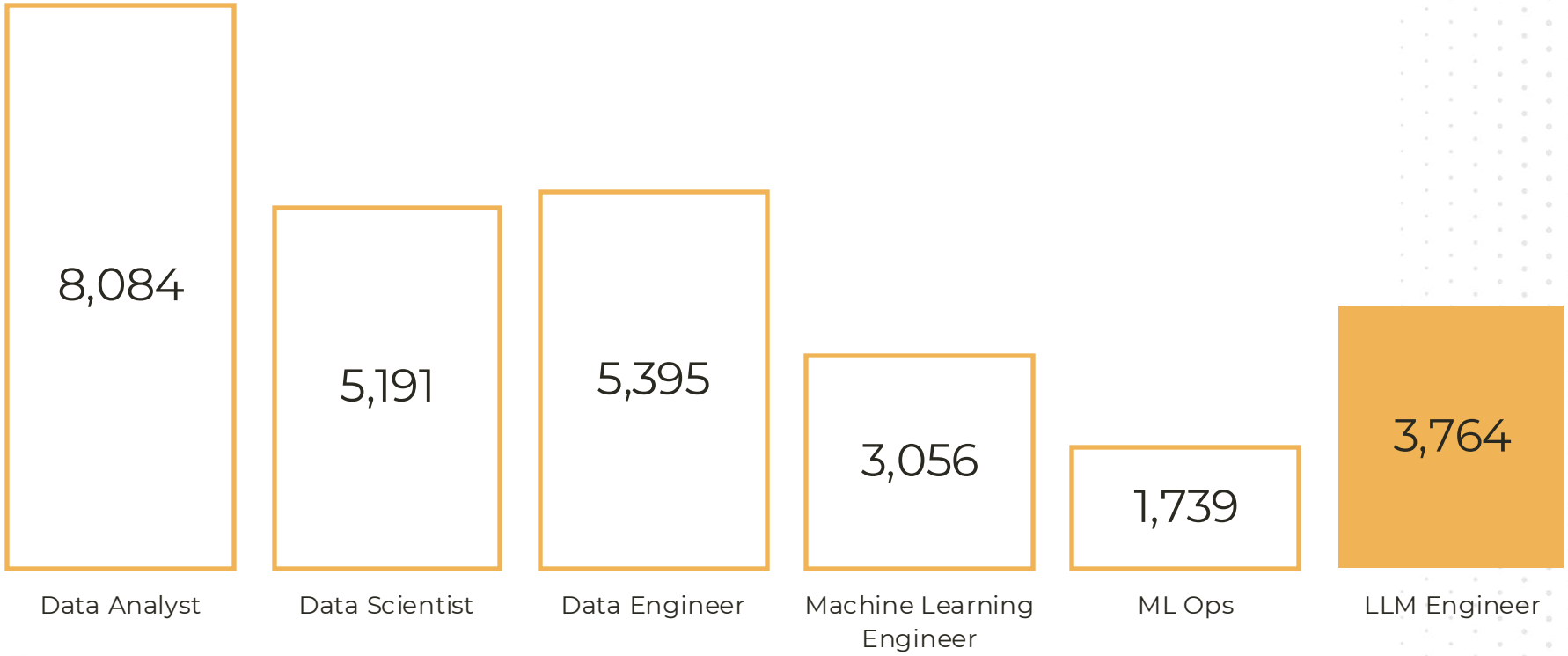


LLM Engineer

LLM Engineer is an evolving hybrid role with Gen AI focus



US Job Postings on LinkedIn (Dec 2024)



Pearson

Traditional Career Path

1

Data Science Masters or PhD

2

Data Scientist

3

Senior Data Scientist

Another Common Career Path

1

Any role involving Data
with specialist domain knowledge

2

Data Analysis
in the same field

3

Data Scientist

Emerging Hybrid Career Paths

1

Software Engineer

2

Software Engineer working with LLM projects

or, ML Engineer

3

LLM Engineer

Right Place at the Right Time

Final extra example projects for those who stayed to the end!

Fine-tuning an LLM on your own Text Message history to create a simulation of yourself!

https://edwarddonner.com/2024/01/02/fine-tuning-an-llm-on-240k-text-messages/

See LLMs try to outwit each other:

https://edwarddonner.com/outsmart/

Next steps

1. Clone the GitHub repo and run & tweak the examples
2. Review the resources, reach out with questions
3. Attend related Live Events

Next steps

1. Clone the GitHub repo and run & tweak the examples
2. Review the resources, reach out with questions
3. Attend related Live Events

…and decide whether a career in AI Data Science is for you!!

THANK YOU!!

Stay in touch:
https://www.linkedin.com/in/eddonner/
ed@edwarddonner.com

Please reach out if I can help. Or for advice on pursuing a career in AI if you choose that path!

You'll be receiving a feedback survey – all feedback is massively appreciated and is instrumental in helping me improve.

Pearson