

What is This?

A Machine Learning Model for Ants?

How to shrink deep learning models, and why you would want to

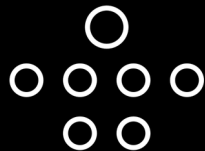
ELIAS TROMMER / 37C3 / DAY 3

Lifecycle of a deep learning model

Training

- Architecture: Describes Parameter Interactions
- Parameters: a few MB to GB

Random
Parameters

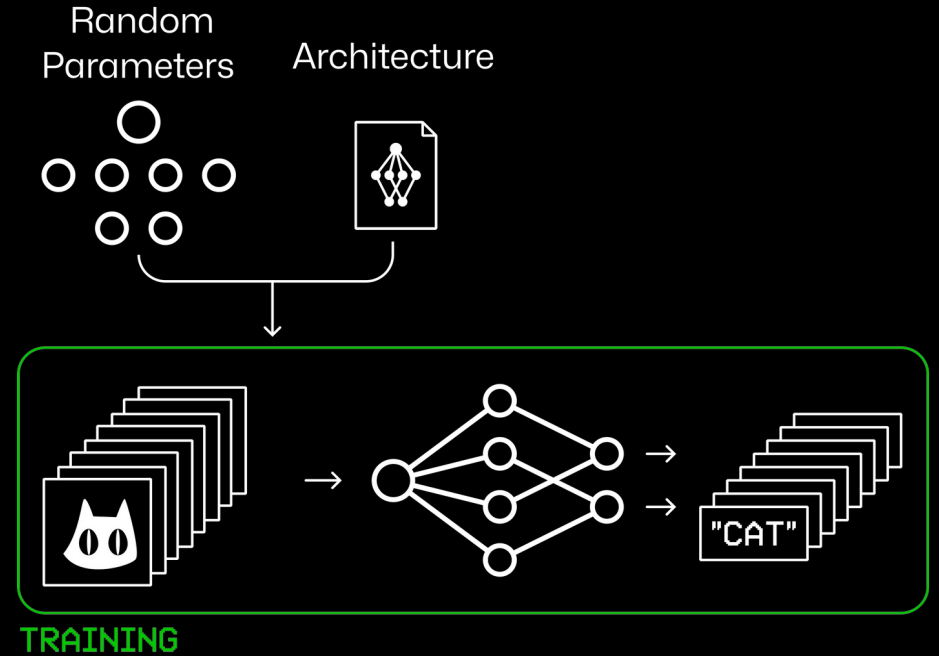


Architecture



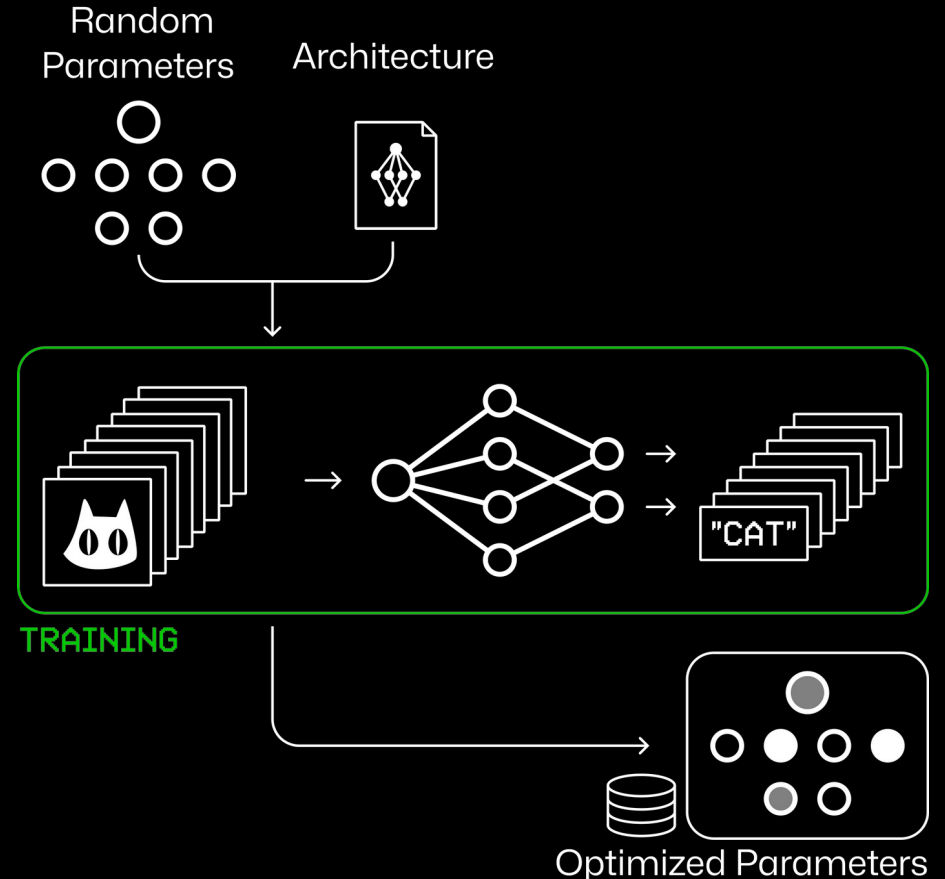
Training

- Architecture: Describes Parameter Interactions
- Parameters: a few MB to GB
- Use input and expected output (“labels”) to learn prominent features from training data



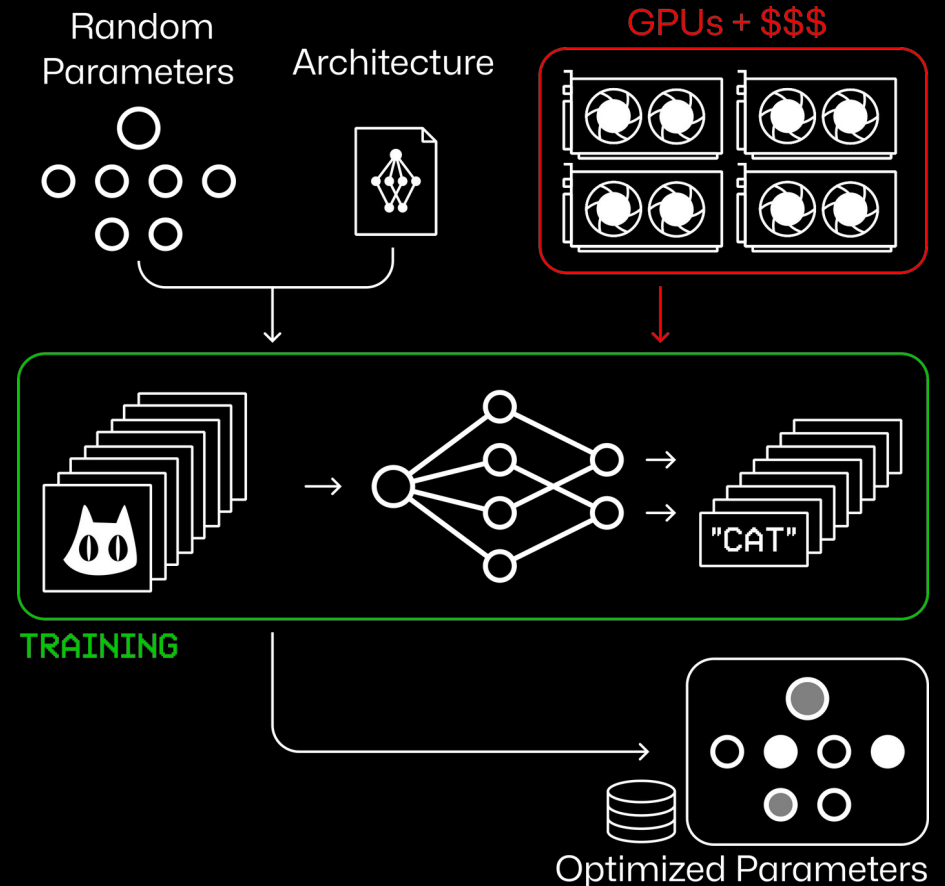
Training

- Architecture: Describes Parameter Interactions
- Parameters: a few MB to GB
- Use input and expected output (“labels”) to learn prominent features from training data
- Knowledge about training data is contained in the optimized parameters



Training

- Architecture: Describes Parameter Interactions
- Parameters: a few MB to GB
- Use input and expected output (“labels”) to learn prominent features from training data
- Knowledge about training data is contained in the optimized parameters
- Don't forget to bring a fleet of high-end GPUs
- Run once/rarely





You

show me an image of a computer scientist in the process of trying to understand the inner workings of a state-of-the-art artificial neural network

JA

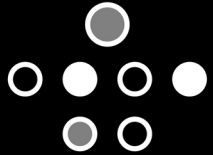
You

show me an image of a computer scientist in the process of trying to understand the inner workings of a state-of-the-art artificial neural network



Inference

Optimized
Parameters

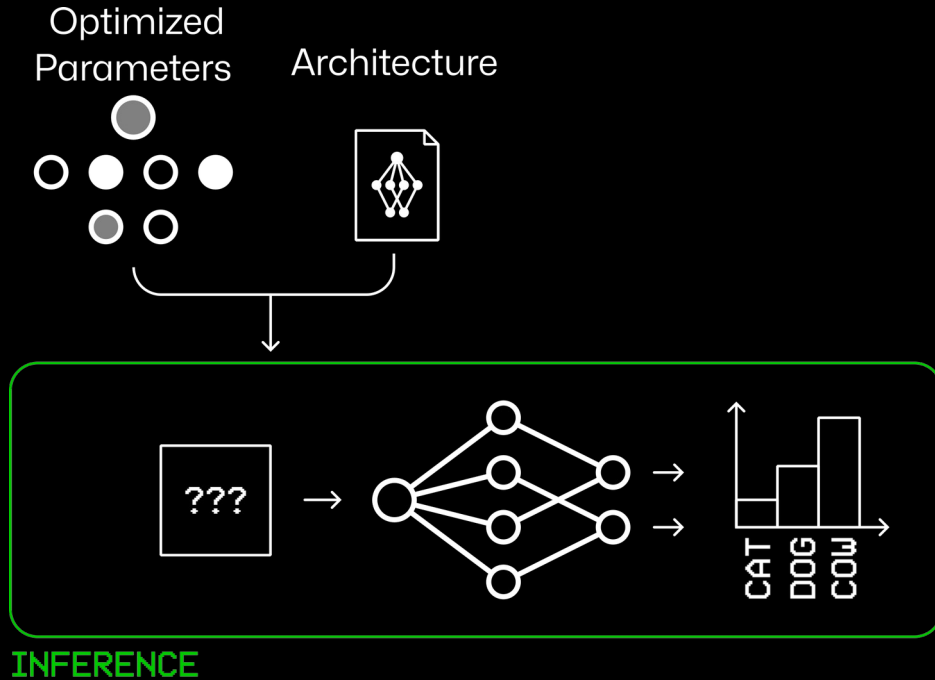


Architecture



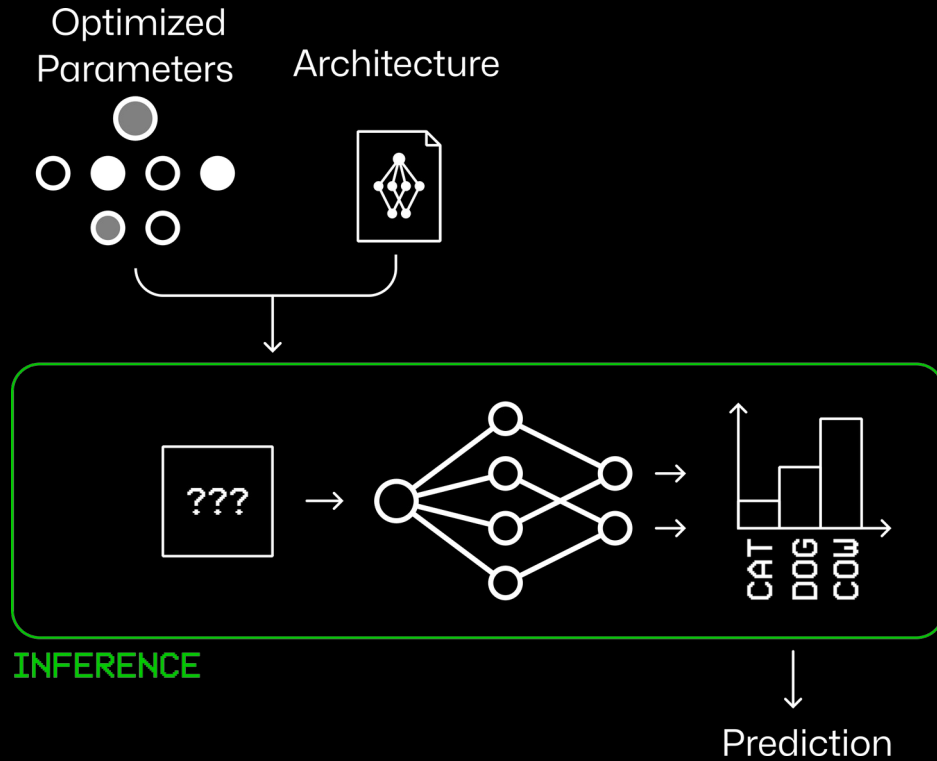
- Take optimized parameters

Inference



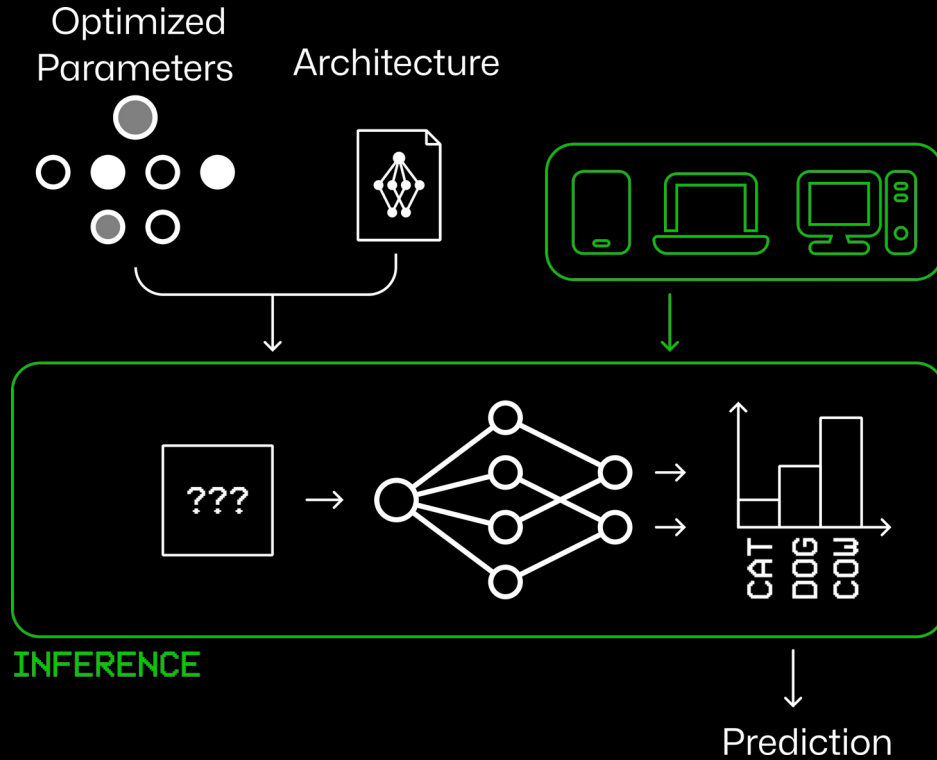
- Take optimized parameters
- Run them on unseen input

Inference



- Take optimized parameters
- Run them on unseen input
- Obtain a prediction

Inference



- Take optimized parameters
- Run them on unseen input
- Obtain a prediction
- Lightweight
- Runs millions of times

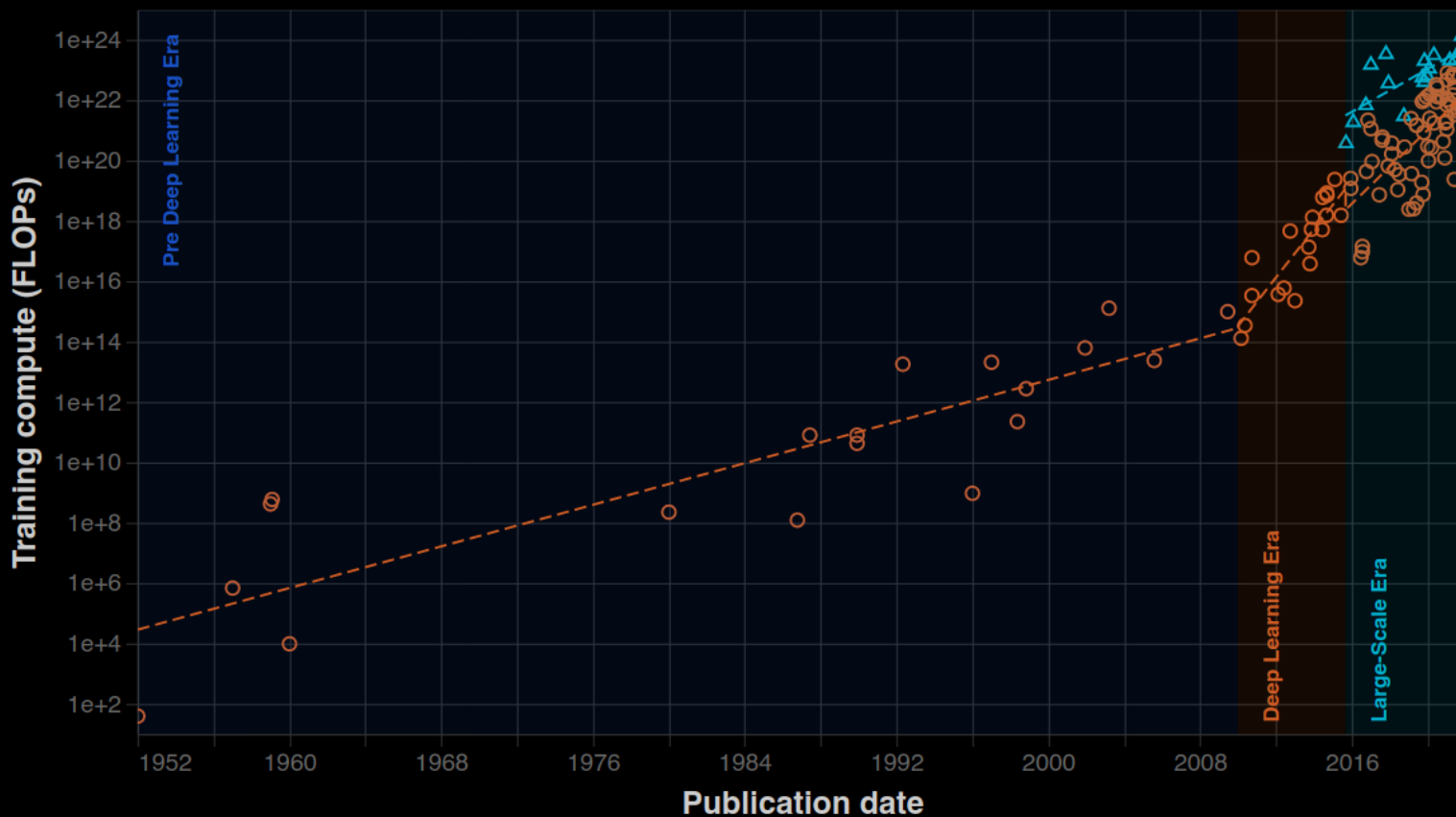
Machine Learning's Energy Problem

How fast are ML models growing?

- **Explosion** of model complexity and required compute through deep learning
- From doubling every 22 months before ca. 2010 to **doubling every 5 months**
- Why? More data + bigger models + more training = Better performance

Training compute (FLOPs) of milestone Machine Learning systems over time

n = 121



Sevilla et al., Compute Trends Across Three Eras of Machine Learning (2022)

The Saarland – but for energy



- Proposed reference for energy:
1 standard wind turbine
- **5 MW** rated power output
- **2,500 full load hours**
(2021, onshore, northern Germany) [1]
- **12.5 GWh** annual output (= 1 🏠) could supply **5000** 2-person households

[1] Statista, Anzahl der Wind-Volllaststunden nach typischen Standorten für Windenergieanlagen in Deutschland im Jahr 2021

Energy Cost of Training

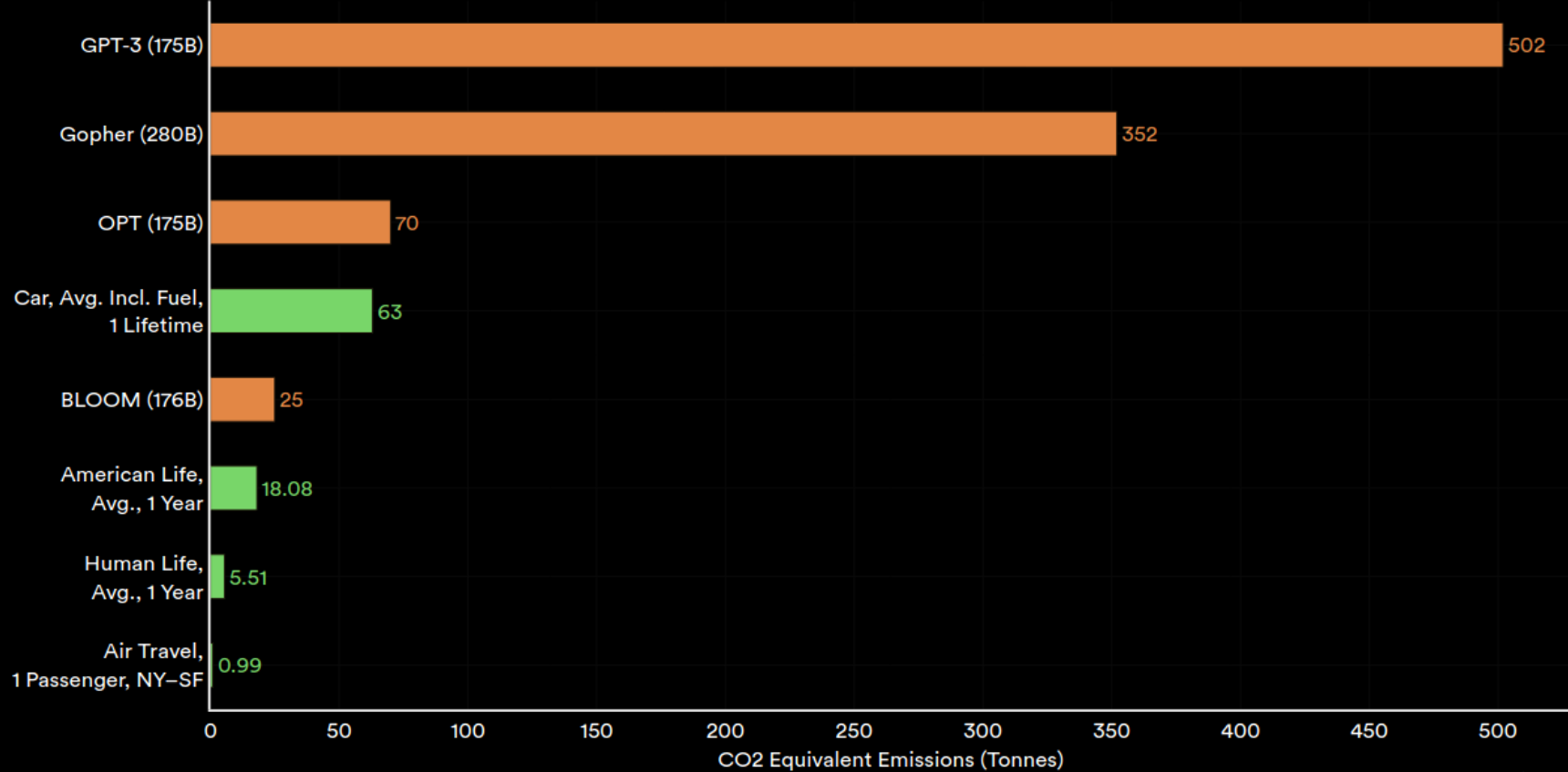
- Between **400MWh** and **1,300MWh** [1] for training a ~200B Parameter LLM
- **0.032** to **0.1** 🌊 (1.5 to 5.5 weeks of wind turbine energy production)
- Computer Vision: ca. **9MWh** for a large model [2] (6h of wind turbine output)

[1] Luccioni et al., Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model (2022)

[2] ViT-H14/JFT, 60k training hours on TPUv3 @ 110W per core + 33% estimated system overhead

CO2 Equivalent Emissions (Tonnes) by Selected Machine Learning Models and Real Life Examples, 2022

Source: Luccioni et al., 2022; Strubell et al., 2019 | Chart: 2023 AI Index Report



Inference Cost

- Low for most applications (Image Detection, Speech Recognition, ...)
- State-of-the art LLMs require significant infrastructure & energy, even for inference
- Guess-timate of ChatGPT's daily energy consumption for inference¹:
 - Estimated **28,936 A100 GPUs** [1] (400W TDP each, 50% utilization)
 - 33% overhead for RAM, CPUs, etc. [2]
 - **185 MWh** daily (0.015 🗣️ / 5.5 days of operation at average annual output)

¹ without embodied emissions of equipment, etc.

[1] Patel, D. & Ahmad, A., The Inference Cost Of Search Disruption - Large Language Model Cost Analysis (2023)

[2] Luccioni et al., Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model (2022)

Where does all the energy go?



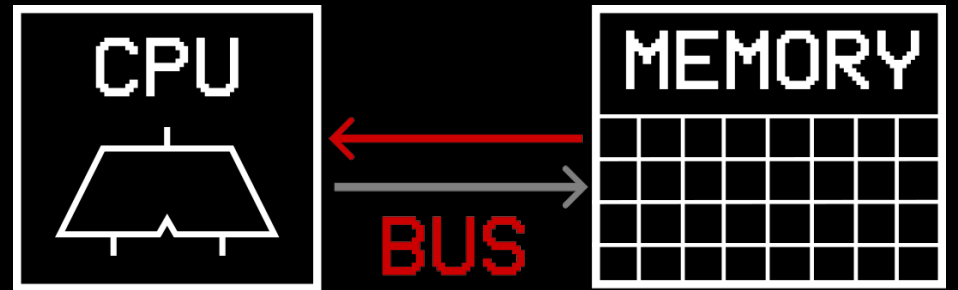
Where does all the energy go?



John von Neumann (1903-1957)

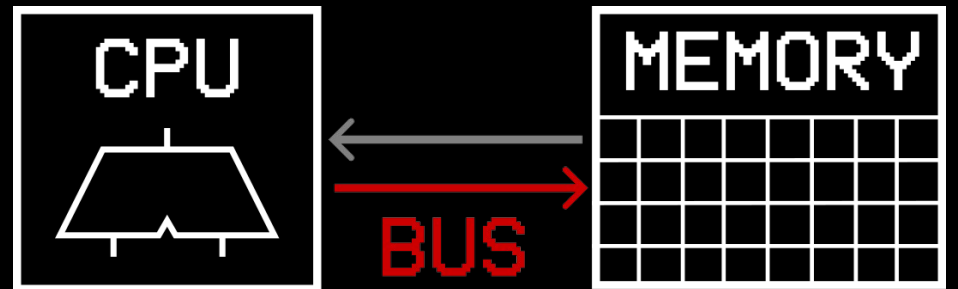
Von Neumann Architecture

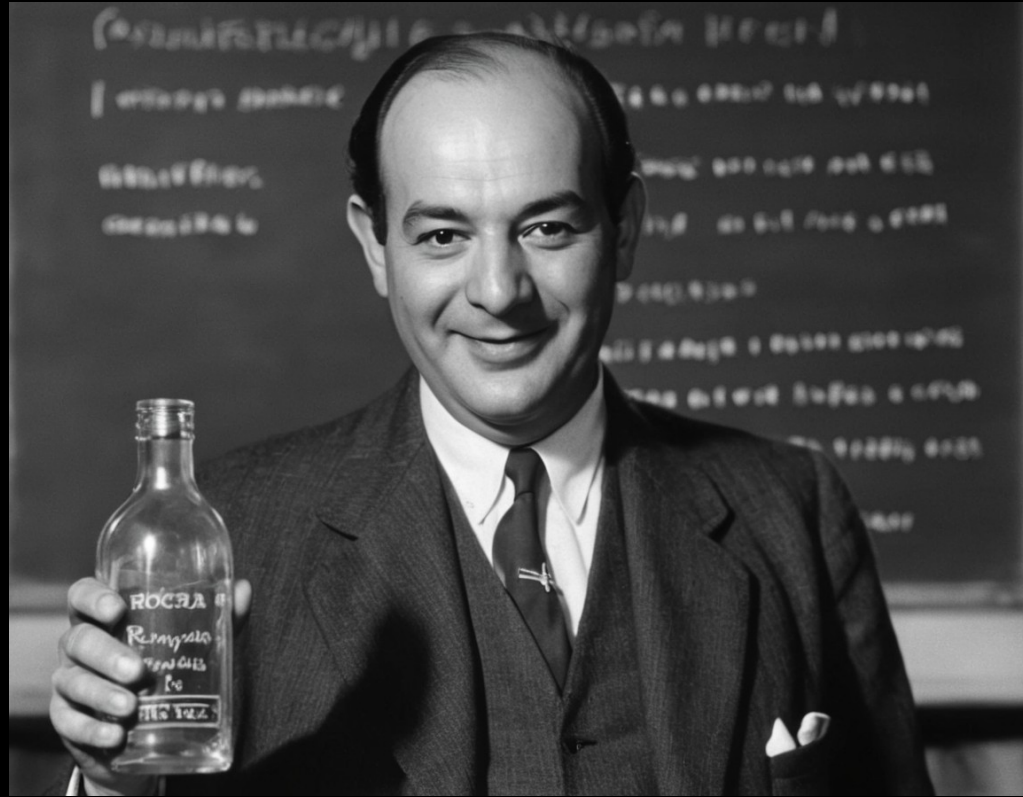
- No distinction between code and data
- Computation and memory are **separate**
- Data must travel through the **bus**
- “Von Neumann Bottleneck”



Von Neumann Architecture

- No distinction between code and data
- Computation and memory are **separate**
- Data must travel through the **bus**
- “Von Neumann Bottleneck”





The Energy Cost of the Von Neumann bottleneck

Operation		Picojoules per Operation		
		45 nm	7 nm	45 / 7
+	Int 8	0.03	0.007	4.3
	Int 32	0.1	0.03	3.3
	BFloat 16	--	0.11	--
	IEEE FP 16	0.4	0.16	2.5
	IEEE FP 32	0.9	0.38	2.4
×	Int 8	0.2	0.07	2.9
	Int 32	3.1	1.48	2.1
	BFloat 16	--	0.21	--
	IEEE FP 16	1.1	0.34	3.2
	IEEE FP 32	3.7	1.31	2.8
SRAM	8 KB SRAM	10	7.5	1.3
	32 KB SRAM	20	8.5	2.4
	1 MB SRAM ¹	100	14	7.1
GeoMean ¹		--	--	2.6
DRAM		Circa 45 nm	Circa 7 nm	
	DDR3/4	1300	1300	1.0
	HBM2	--	250-450	--
	GDDR6	--	350-480	--

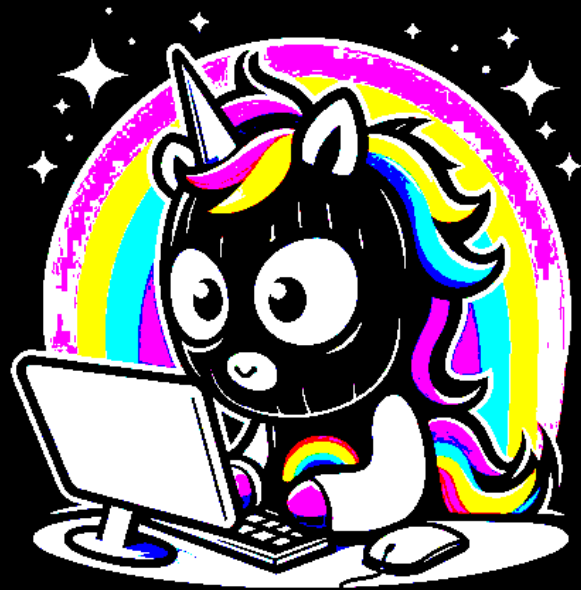
Table 2. Energy per Operation: 45 nm [16] vs 7 nm. Memory is pJ per 64-bit access.

- Moving data around is energy-intensive
- Neural Networks move **a lot** of data around



Shrinking networks for inference

Quantization



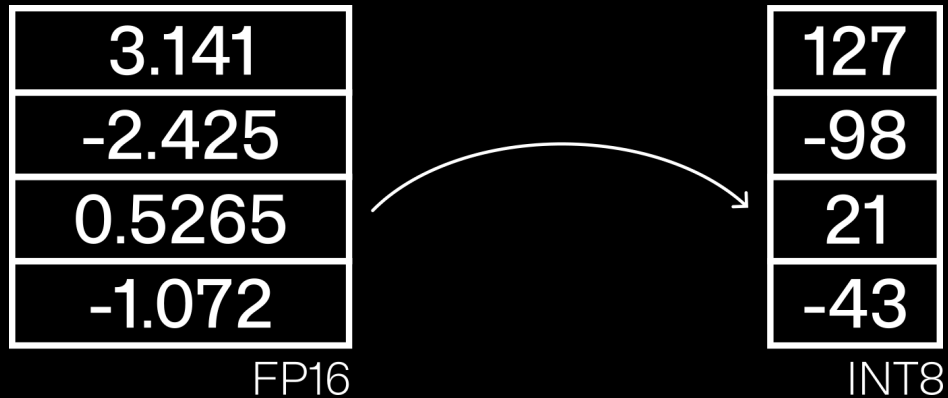
Quantization

3.141
-2.425
0.5265
-1.072

FP16

- Training done in 16-bit floating-point

Quantization



- Training done in 16-bit floating-point
- Convert parameters to 8-bit (or less) Integer for inference
- Slight loss in performance, but:
 - Half the memory footprint
 - Easier to build hardware for
- Changes maths, but not too much

Pruning




Pruning

112	-55	-11	-95
4	62	17	-15
74	32	-33	5
-3	72	-87	103

- Parameters are dense and highly regular
- Likely a lot of redundancy

Pruning



112	-55	-11	-95
4	62	17	-15
74	32	-33	5
-3	72	-87	103

112	-55	0	-95
0	62	0	0
74	0	0	0
0	72	-87	103

- Parameters are dense and highly regular
- Likely a lot of redundancy
- Pruning removes least important elements (based on some heuristic)
- Up to ~50% of elements can typically be removed without noticeable accuracy loss

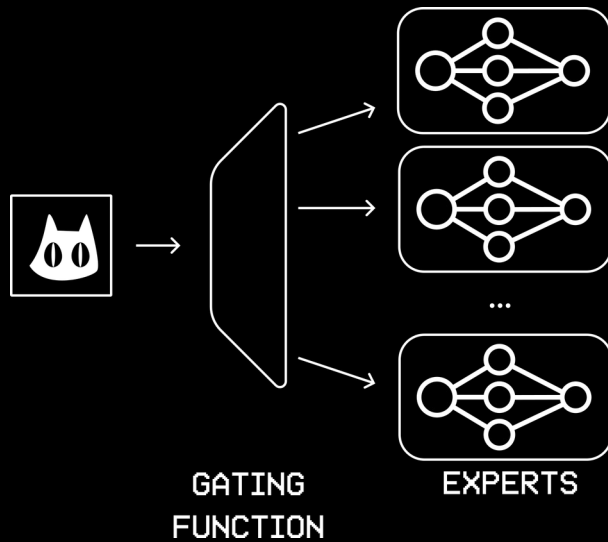
Mixture of Experts



Mario Casciano, Wikimedia, CC BY-SA 3.0

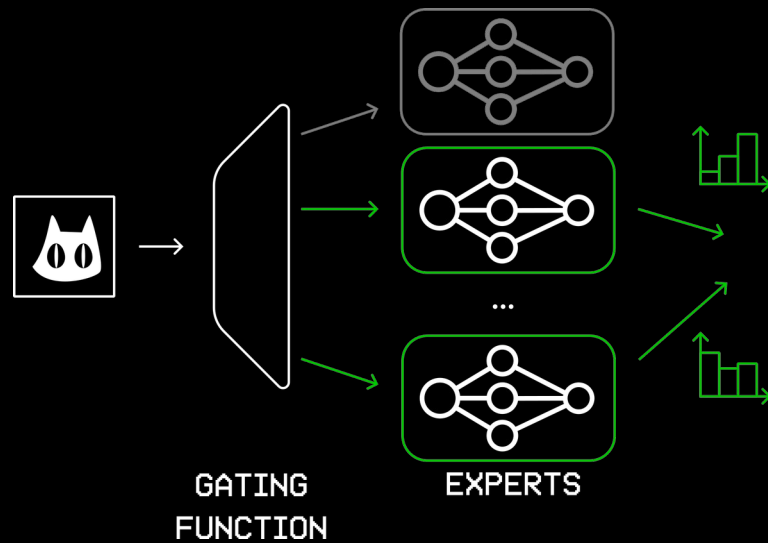
Mixture of Experts

- Gating function and several sub-networks (“experts”) are optimized during training



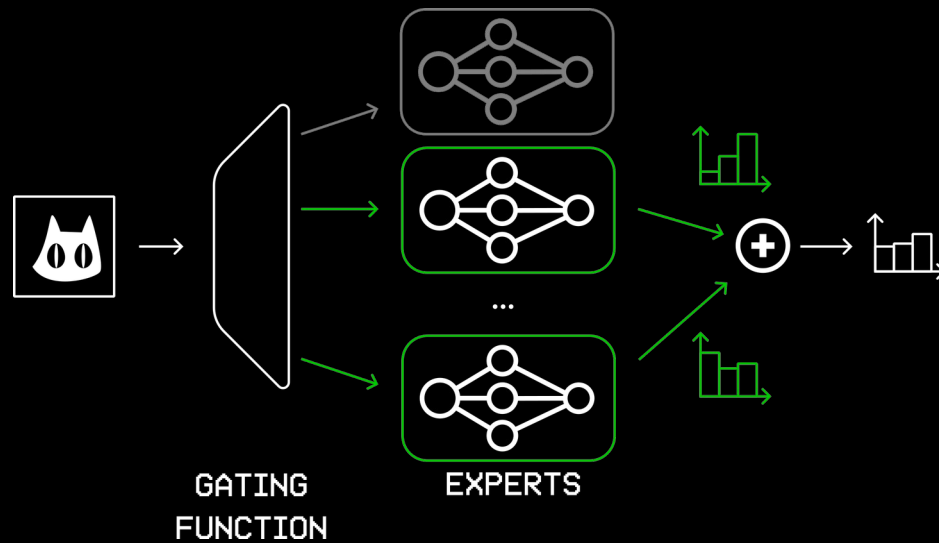
Mixture of Experts

- Gating function and several sub-networks (“experts”) are optimized during training
- Gating function forwards input to only some of the experts at runtime



Mixture of Experts

- Gating function and several sub-networks (“experts”) are optimized during training
- Gating function forwards input to only some of the experts at runtime
- Results are combined



Knowledge Distillation

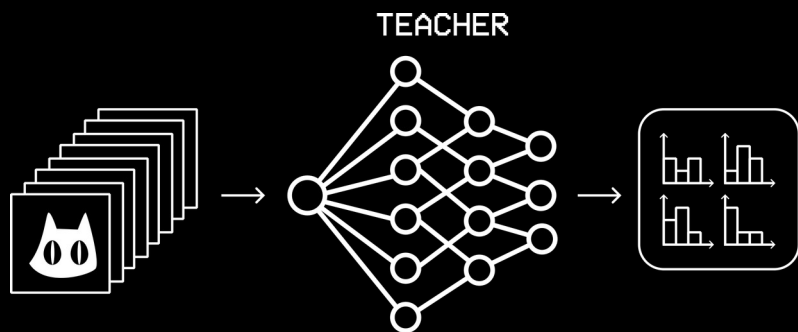


**Training a model
on data**



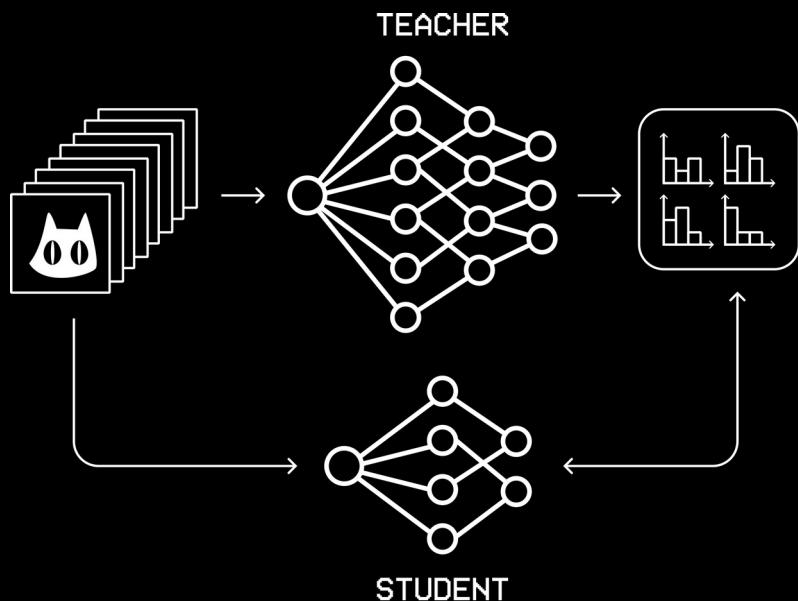
**Training a model
on another model**

Knowledge Distillation



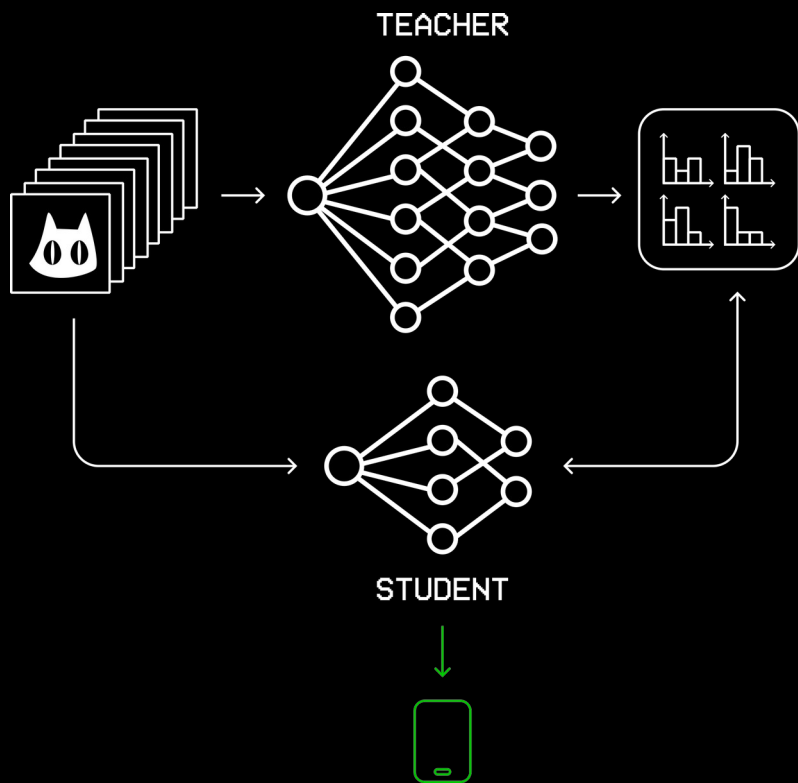
- Train a large **teacher** model first

Knowledge Distillation



- Train a large **teacher** model first
- Then train a smaller **student** model on the **probability outputs** of the teacher
- “Hidden knowledge” of teacher helps student learn better than direct training

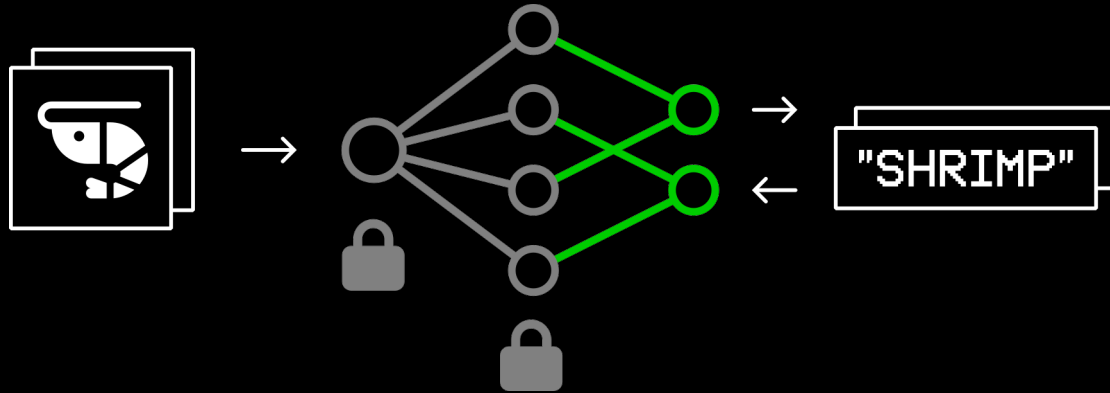
Knowledge Distillation



- Train a large **teacher** model first
- Then train a smaller **student** model on the **probability outputs** of the teacher
- “Hidden knowledge” of teacher helps student learn better than direct training

Adapting existing models

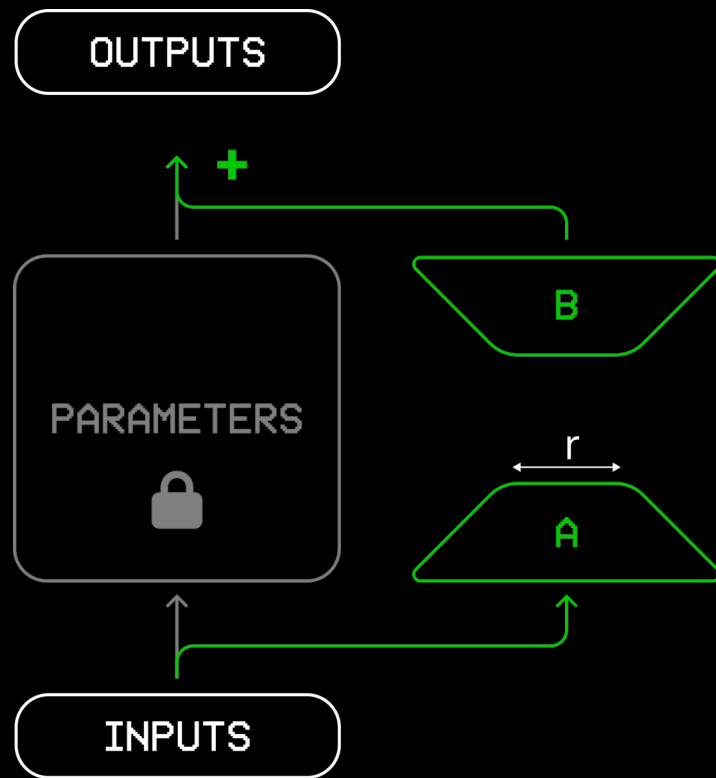
Transfer Learning



- Adapt a pre-trained for a new task with a **small** dataset
- Freeze feature detection (colors, edges, primitive shapes)
- Only train the classifier (part that generates the prediction)
- Mostly useful for simpler tasks

Low-Rank Adaptation

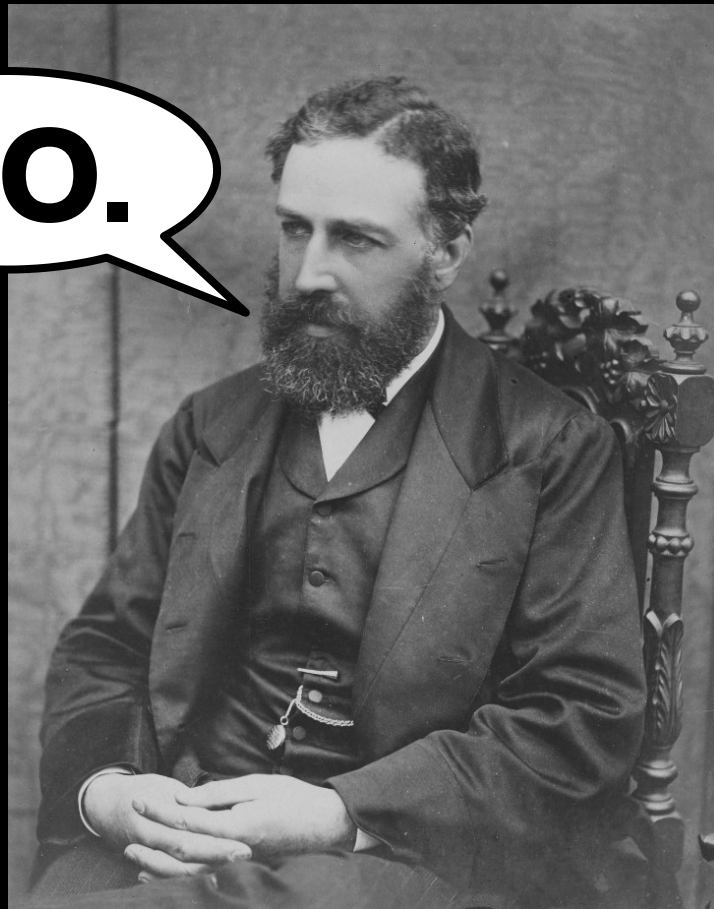
- Freeze original parameters frozen
- Additionally transform input through **A** and **B**
- **A** and **B** only have a small number of learnable parameters
- Can be merged with original parameters



**Higher efficiency reduces energy
consumption**

...right?

NO.



William Stanley Jevons
(1835 - 1882)

Summary

- Energy consumption is **already an issue**, especially with LLMs
- **10× scaling every 2 years** means that this will likely get worse in the near future
- Running down-scaled models **locally** is almost always an option
- **Training** cutting-edge models from scratch is out of reach for everyone but the largest corporations. **Adapting** models is not.