



Governing Artificial Intelligence: Law, Policy, and Institutions

Law 4052 // Computer Science 283 // Communication 252A // Political Science 245B/445B
Fall Quarter 2026-27

Rob Reich
Anka Reuel
Nate Persily
Sanmi Koyejo



Housekeeping

Reminder: no devices in lecture

Optional/voluntary sections: begin (Thursday or Friday this week)

Stanford and Silicon Valley Culture

Techies vs Fuzzies

Required reading for CS283

cs283 home | schedule | lectures | assignments | sections | resources | ed



LECTURE 2

Technical Foundations & Lessons from the History of Governing Frontier Technology

Monday, September 28, 2026

Technical Foundations

This lecture provides a technical foundation for understanding how modern AI systems work. We'll cover the core concepts that underlie today's large-scale models, including stochastic gradient descent, neural networks, and key optimization techniques. Students will learn about model architectures used in foundation models, as well as common strategies for improving performance such as fine-tuning, quantization, and distillation. We'll examine how scale affects model capabilities, and introduce reasoning phenomena such as chain-of-thought reasoning. This technical grounding is essential for analyzing later governance challenges related to safety, evaluation, and access.

REQUIRED

18 PAGES

Karpathy, A. 2023
[Introduction to Large Language Models](#) —
YouTube video, 1 hr

Lages, J. 2023
[Direct Preference Optimization \(DPO\) — A Simplified Explanation](#) 6 p.

Lee, H., et al. 2024
[RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback](#) 9 p.
Proceedings of the 41st International Conference on Machine Learning (ICML), PMLR 235, 26874–26901

Toner, H. 2023
[What Are Generative AI, Large Language Models, and Foundation Models?](#) 3 p.
Center for Security and Emerging Technology

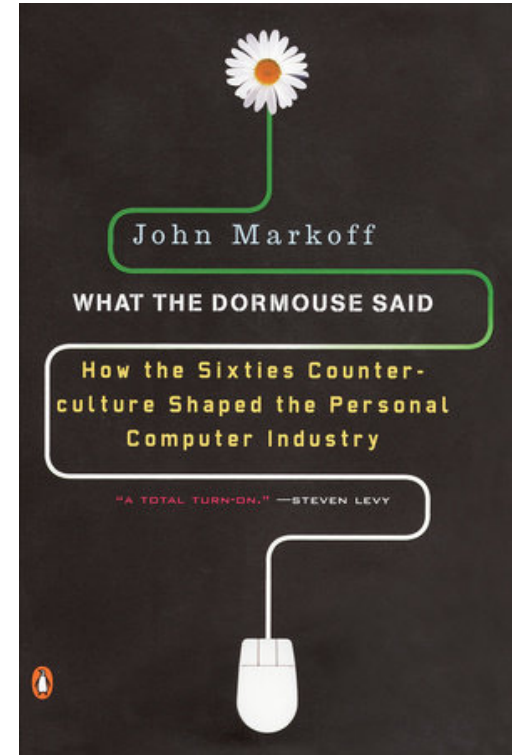
SUPPLEMENTARY

75 PAGES

Wolfram, S. 2023
[What Is ChatGPT Doing ... and Why Does It Work?](#) 75 p.

John Markoff, What the Dormouse Said

Chapter 6: Scholars and Barbarians



Shackled by History? Shackled by Knowledge?

“I’m addicted to reading, which explains how I ended up being a writer.”

“Oh, yeah?” says SBF. “I would never read a book.”

“I’m very skeptical of books. I don’t want to say no book is ever worth reading, but I actually do believe something pretty close to that,” explains SBF. “I think, if you wrote a book, you fucked up, and it should have been a six paragraph blog post.”

Sam Bankman-Fried, to writer Adam Fisher, originally on the Sequoia Capital website, 2022



Sam Bankman Fried, CC-license

Image from the Intel Museum within Intel Corporate HQ,
Santa Clara, CA

**"Don't be encumbered by history.
Go off and do something wonderful."**

- Robert Noyce

Robert Noyce, founder of Intel (1968)



A (Very Short) History of Computer Science

1936

1950

1956

1965

Is Computer Science a Science?

What are the basic characteristics of a "science"?

- empirical grounding

- systematic method

- testability/falsifiability

- explanation and prediction

- reproducibility and independent scrutiny

Does AI/ML fit the definition?

AI in the Context of Previous Science and Technology

Metaphors to understand the significance of AI

Historical analogies to inform governance.

The background of the entire image is a stylized, cartoonish representation of fire. It features numerous flame shapes in various shades of orange, red, and yellow, set against a dark brown background. The flames are depicted with smooth, flowing outlines and some internal shading to give them a three-dimensional appearance. Small, glowing orange and yellow dots are scattered throughout the dark background, resembling embers or sparks.

Fire

Tim Cook & Sundar Pichai

Electricity



Andrew Ng

Printing Press

Henry Kissinger



The Bomb



Sam Altman

"What Have We Done?"

Sam Altman invokes the Trinity test — the first atomic bomb detonation — as AI's defining "what have we done?" moment.

Elon Musk goes further

AI is potentially **more dangerous** than nuclear weapons — a statement that would have been unthinkable a decade ago.





**"A country of
geniuses
in a datacenter."**

— Dario Amodei, CEO of Anthropic



Alien Intelligence

Mustafa Suleyman — a new digital species

Yuval Harari — encounters with alien minds

Making Sand Think

Demis Hassabis, Nobel Prize-winning co-founder of DeepMind, believes AGI is only a few years away — a system that equals or exceeds every cognitive capability of a human being.

His most arresting observation: silica — sand — is the core ingredient in computer chips. We have found a way to **make sand think**.

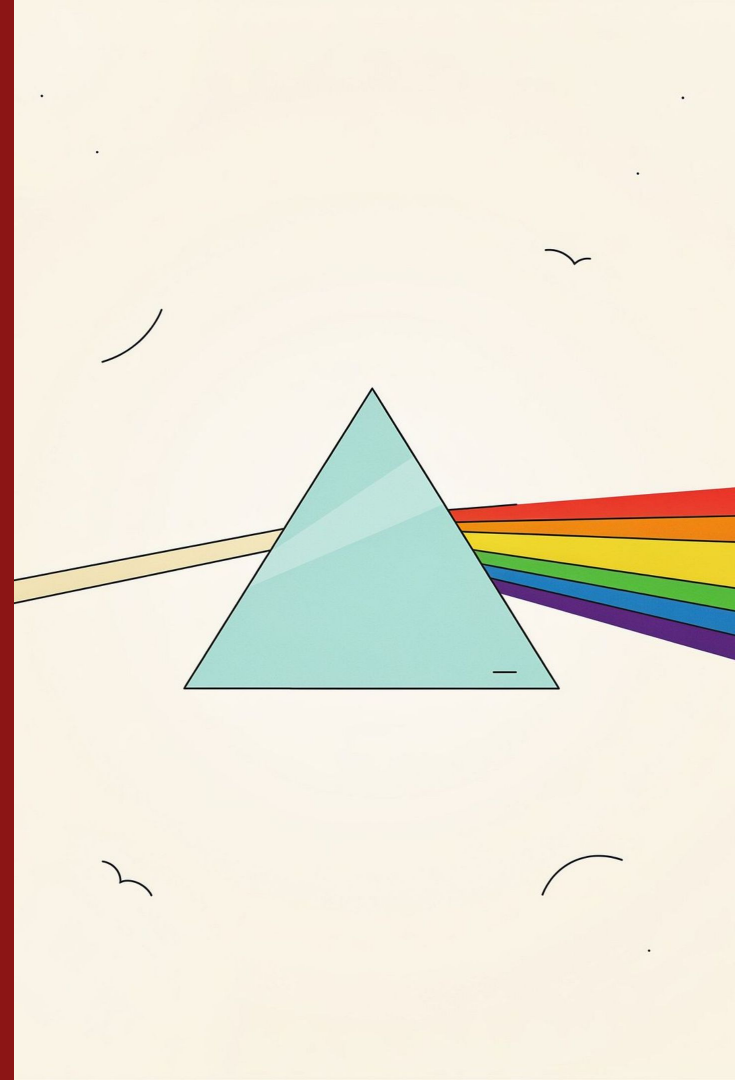




Fire. Electricity. The Bomb. A New Species.

We keep reaching for the largest comparisons in history — because nothing smaller fits. The question isn't which analogy is right. It's what it means that we need one at all.

AI Is Metaphorically Exhausted



Technical Foundations: What to Teach?

What you submitted!

Everything

What You Wanted to Know About

Jev GPU TPU Multi-agent Multimodality MCP Compute Optimization RLHF Fine-tuning Foundation Models Scaling law Downstream Embeddings Weights Hallucinations RNN Attention RAG Reinforcement Learning Causal Learning Federated Learning Transfer Learning Web 2.0 vs. Web 3.0 Black box Agents Deep Learning Federated Learning LSTM LLM Pre-/Post-training Inference Engines LLM Uncertainty Quantification AGI Postmarket monitoring HuggingFace Incident Human-in-the-loop RSI

Today

GPU/TPU
Multimodality
Fine-tuning
Deep Learning
Foundation Models/LLMs
Pre-/Post-Training
RLHF
Embeddings
Attention
Neural Networks
Weights
Reinforcement Learning

Later this quarter

Downstream	Agents
Compute	MCP
Scaling laws	Transfer Learning
Human-in-the-loop	Chips
Black box	Synthetic data
Superintelligence	RAG
Federated Learning	AGI
Algorithmic fairness	Hallucinations
Jev	RSI
HuggingFace Incident	Multi-agent Postmarket monitoring

Not in the class

Web 2.0/Web 3.0
Causal Learning
Uncertainty
Quantification
Inference Engines
LSTMs
RNN
CNN
Optimization

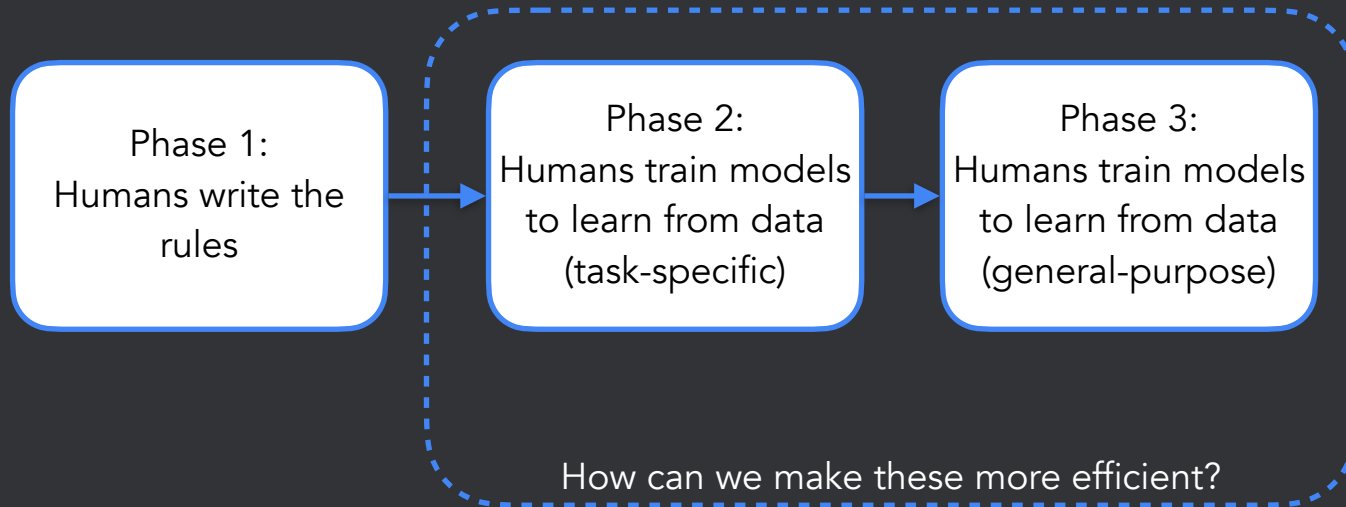
We're also happy to share course recs for other Stanford classes on these topics!

The expectation is not that you understand every single detail in this lecture. Take it as a roadmap.

What are the governance levers, assumptions, and failure modes here?

Three phases of how AI is built and used and their governance implications

(according to Rob and Anka)



Phase 1: Humans write the rules

We specify all relevant knowledge and rules needed to solve a problem!

Key ideas:

Search

Symbolic AI

Expert Systems

1956: The Dartmouth Summer Research Project

A PROPOSAL FOR THE
DARTMOUTH SUMMER RESEARCH PROJECT
ON ARTIFICIAL INTELLIGENCE

J. McCarthy, Dartmouth College
M. L. Minsky, Harvard University
N. Rochester, I. B. M. Corporation
C. E. Shannon, Bell Telephone Laboratories

“The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language form abstractions and concepts solve kinds of problems now reserved. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together

– John McCarthy

*“The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language form abstractions and concepts solve kinds of problems now reserved. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together **for a summer**”*

– John McCarthy

Weak or Narrow AI

Good at one
task

Broad/General- Purpose AI

Good at a few
tasks

Strong or General AI/AGI

Can perform
any intellectual
task that a
human can*



Superintelligence/ ASI

AI that surpasses
human
intelligence in all
domains

*we'll discuss different definitions of AGI in the x-risk lecture!

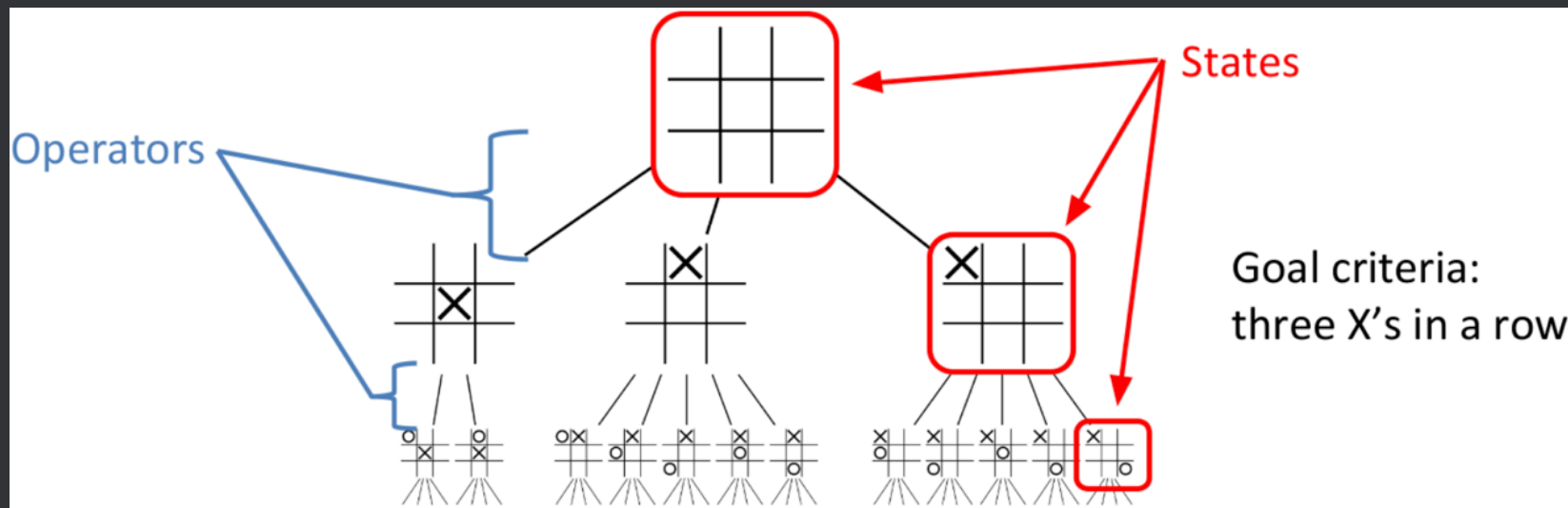
1950s–1960s: GOFAI (Good Old Fashioned AI)

- Assumption: intelligence could be represented as symbols and logical rules
- Early approaches:
 - The Logic Theorist by Simon & Newell (proved theorems a given set of axioms and a set of rules)
 - General Problem Solver by Simon, Newell, & Shaw (defined a 'goal,' and conducted a search to find a series of valid moves to get to said goal)



Rodriguez, J. (2023) "A Peek Into AI History: The First AI Program"

1950s–1960s: AI as Search



Focus: general problem solving, rule-based approaches

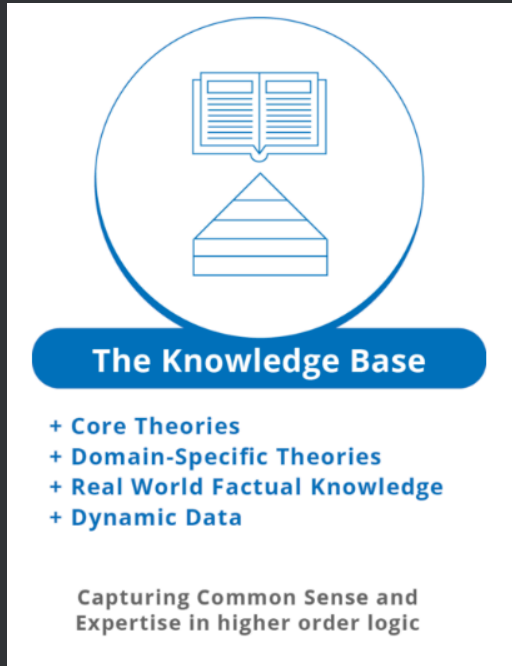
Late 1970s: Expert Systems

- Technical:
 - First expert system developed at Stanford (Feigenbaum, Buchanan, Lederberg)
 - DENDRAL helped chemists identify unknown possible organic molecules structures
- Shift from general problem-solving to domain-specific knowledge systems



Edward Feigenbaum

1984: Cyc (still ongoing!)



- 40,000 predicate to express relationships, e.g., "located in(x,y)"
- 1,500,000 concepts, e.g. "Paris"
- 25,000,000+ explicitly-stated axioms (e.g., "Paris is in France")

If humans write the rules,
whose knowledge and values
get encoded?

Weak or Narrow AI

Good at one
task



Broad/General- Purpose AI

Good at a few
tasks

Strong or General AI/AGI

Can perform
any intellectual
task that a
human can*

Superintelligence/ ASI

AI that surpasses
human
intelligence in all
domains

*we'll discuss different definitions of AGI in the x-risk lecture!

Phase 2a: Task-specific learning

How can we make a model *learn* a specific task?

Key ideas:

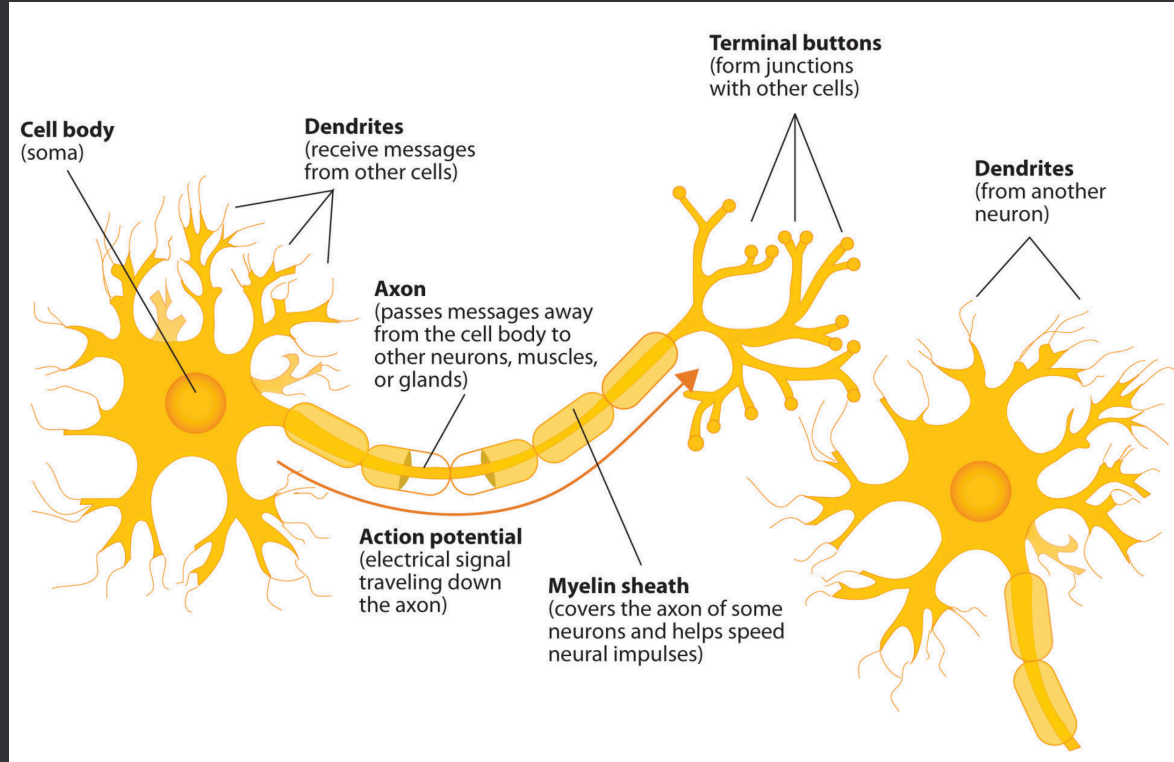
Perceptrons/Neural
Networks

Backpropagation

GPUs/TPUs

ImageNet/AlexNet

1958: The perceptron



1958: The perceptron



[Journal Information](#)
[Journal TOC](#)

APA PsycArticles: Journal Article

The perceptron: A probabilistic model for information storage and organization in the brain.

© Request Permissions

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408. <https://doi.org/10.1037/h0042519>

To answer the questions of how information about the physical world is sensed, in what form is information remembered, and how does information retained in memory influence recognition and behavior, a theory is developed for a hypothetical nervous system called a perceptron. The theory serves as a bridge between biophysics and psychology. It is possible to predict learning curves from neurological variables and vice versa. The quantitative statistical approach is fruitful in the understanding of the organization of cognitive systems. 18 references. (PsycInfo Database Record (c) 2025 APA, all rights reserved)

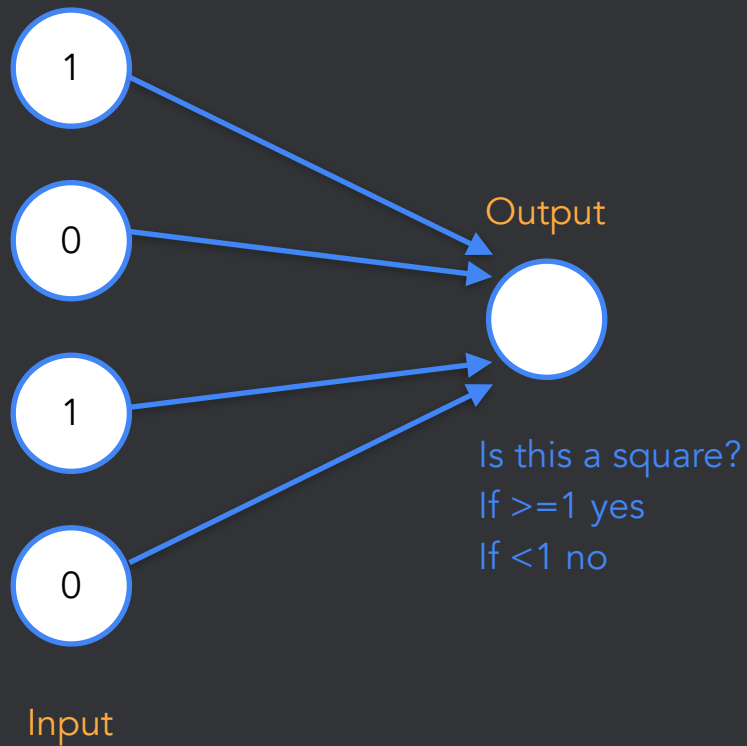
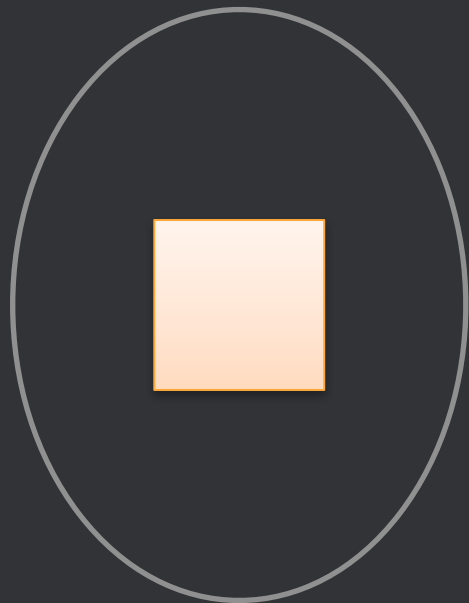
Copyright

Statement: All rights, including for text and data mining, AI training, and similar technologies, are reserved.

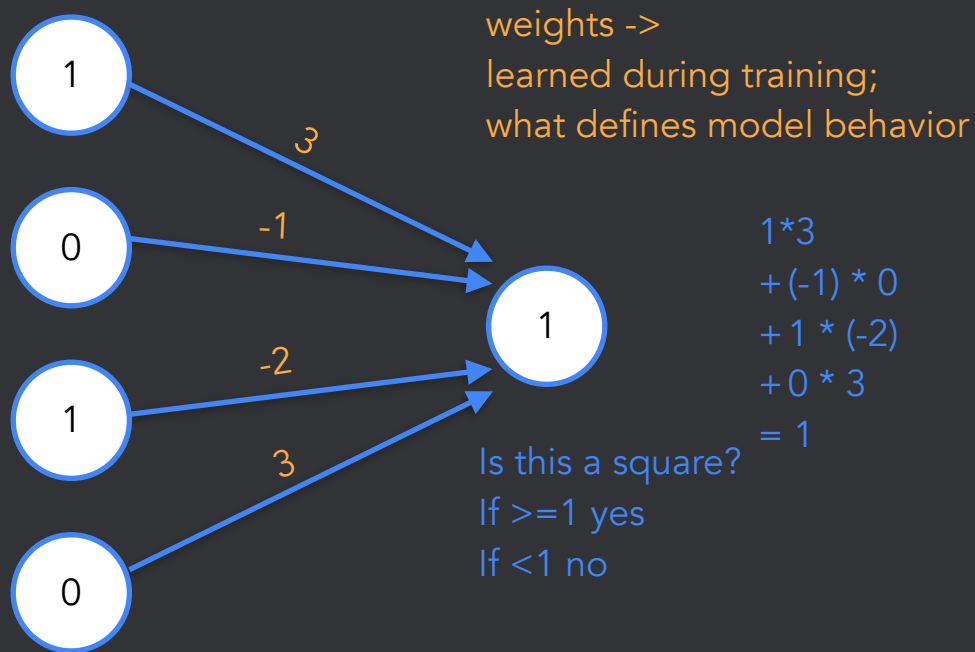
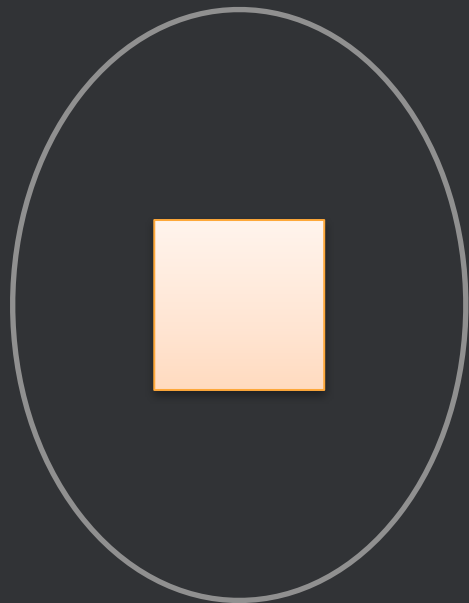
Holder: American Psychological Association

Year: 1958

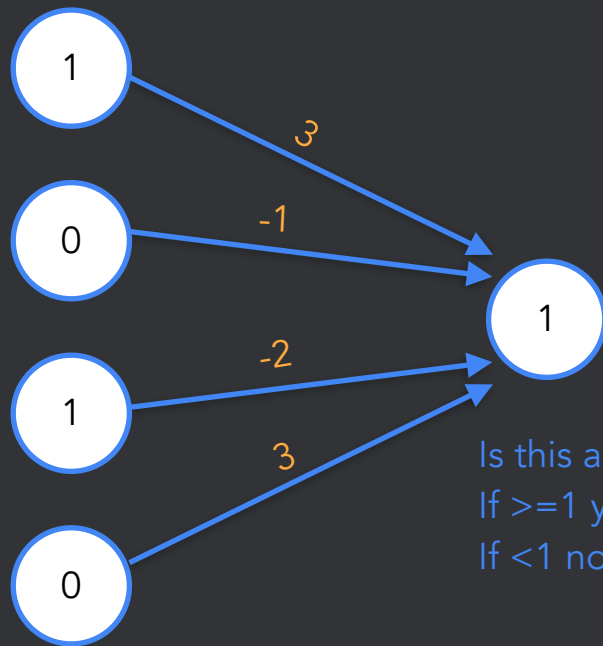
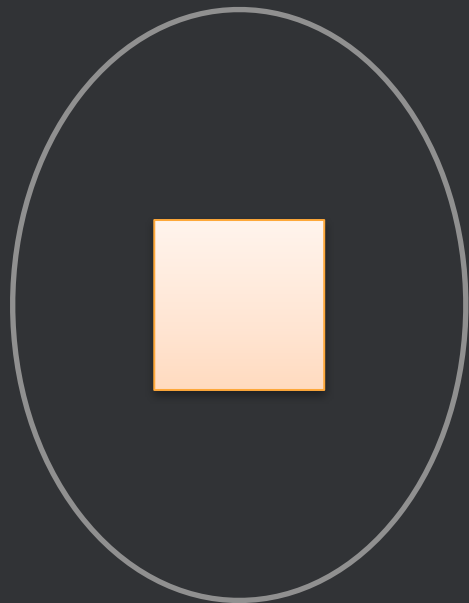
1958: The perceptron



1958: The perceptron

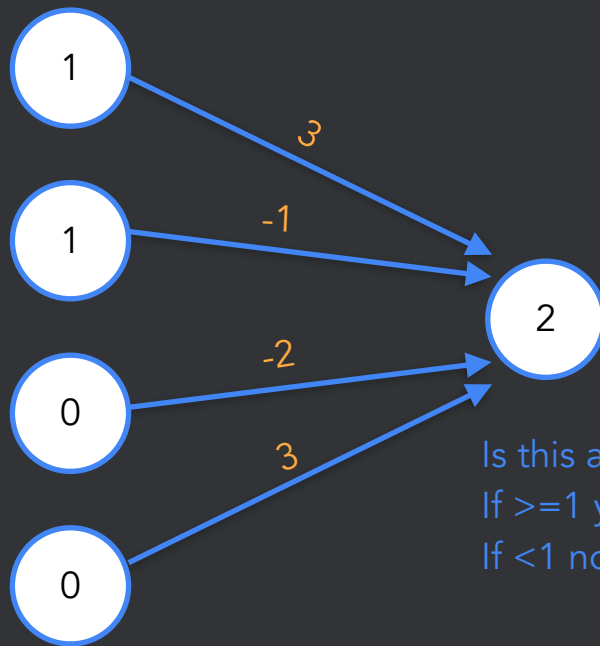


1958: The perceptron



Is this a square?
If ≥ 1 yes
If < 1 no

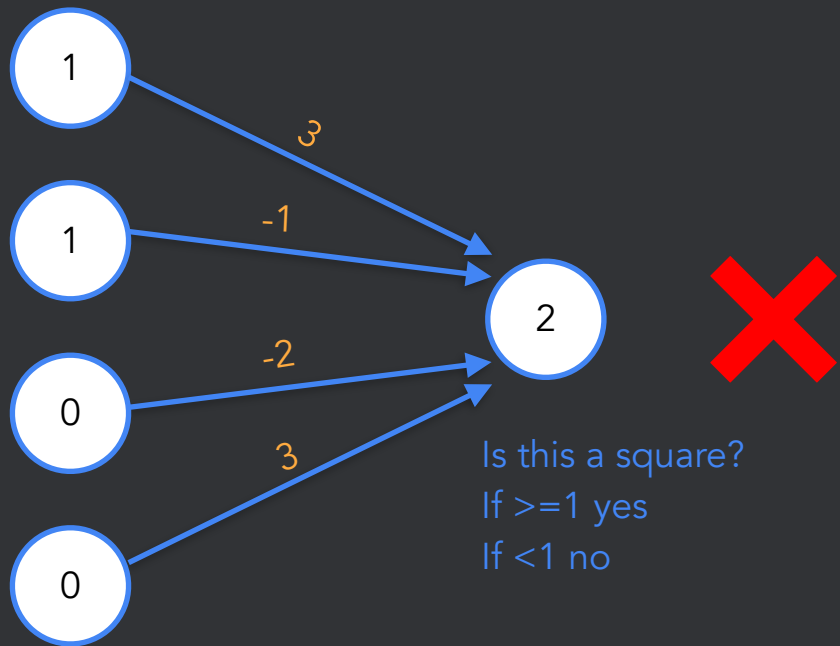
1958: The perceptron



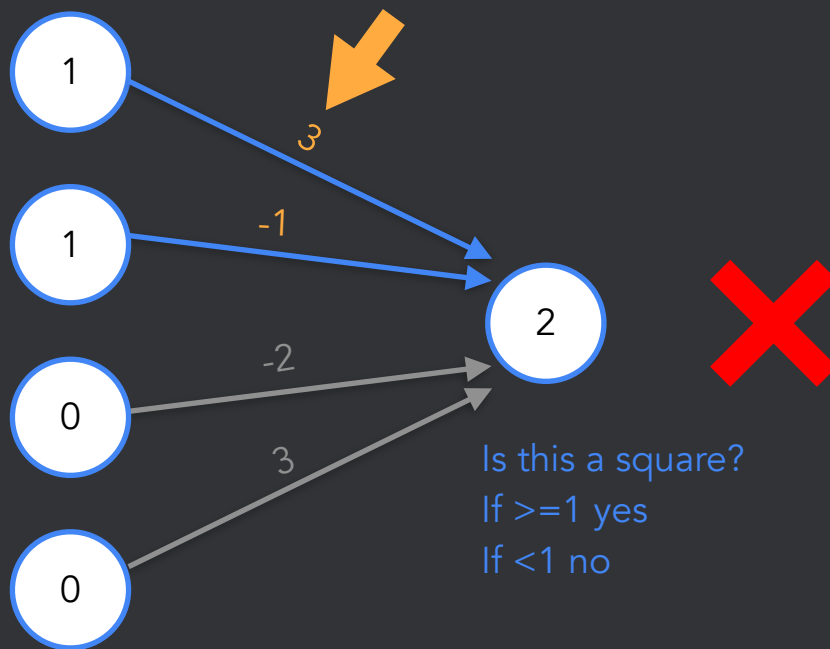
$$\begin{aligned} &1 * 3 \\ &+ (-1) * 1 \\ &+ 0 * (-2) \\ &+ 0 * 3 \\ &= 2 \end{aligned}$$

Is this a square?
If ≥ 1 yes
If < 1 no

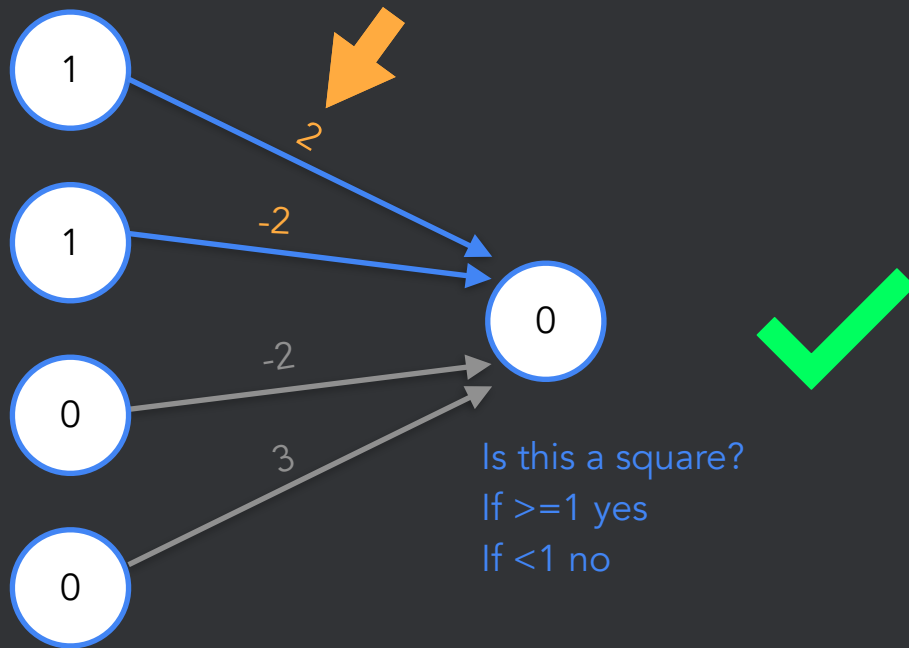
1958: The perceptron



1958: The perceptron



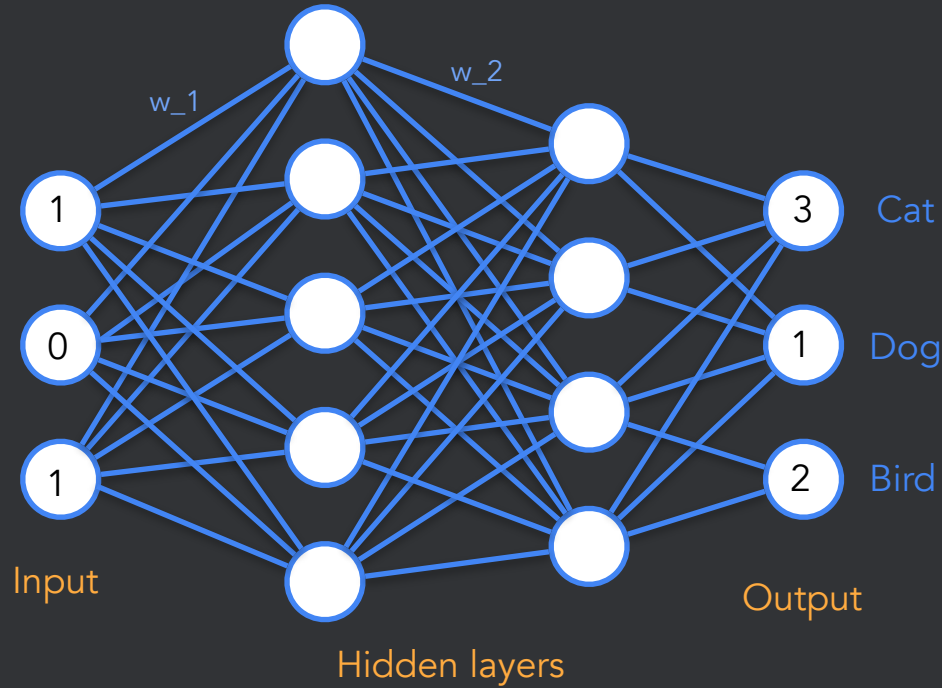
1958: The perceptron



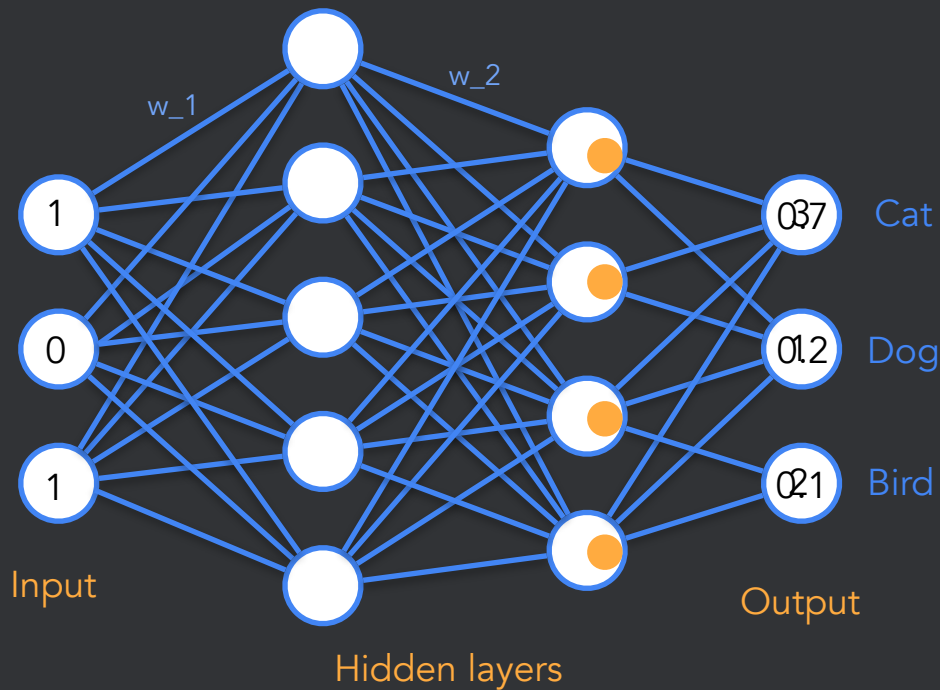
1958: The perceptron

- One-layer neural network (inspired by how humans processed information)
- Random initialization
- Learning by giving it 100s of training examples (the training data that the model processes to learn the weights)

~1960s: Multilayer Perceptron

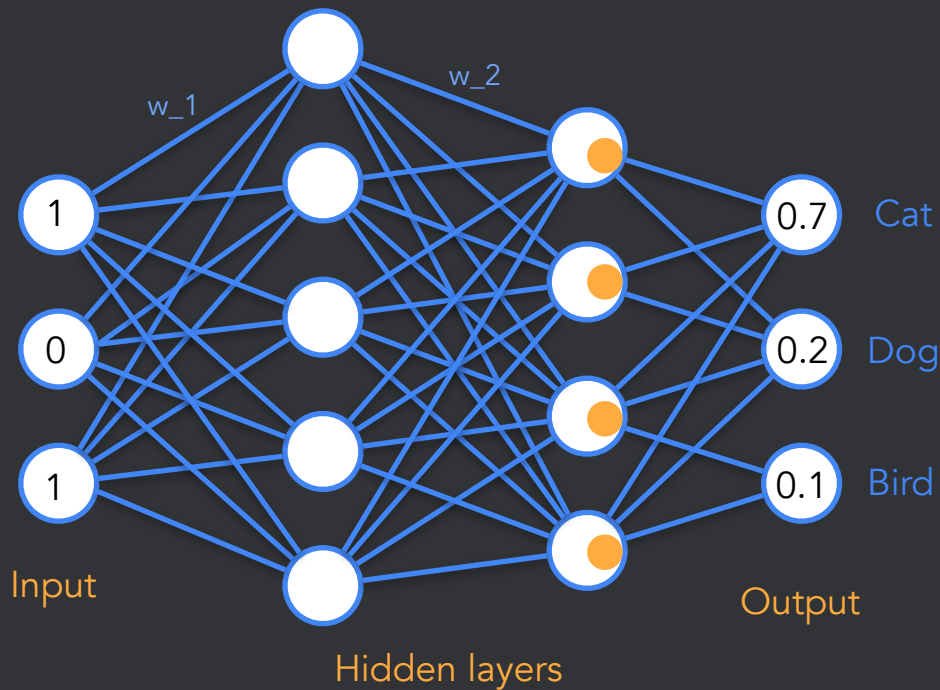


~1960s: Multilayer Perceptron



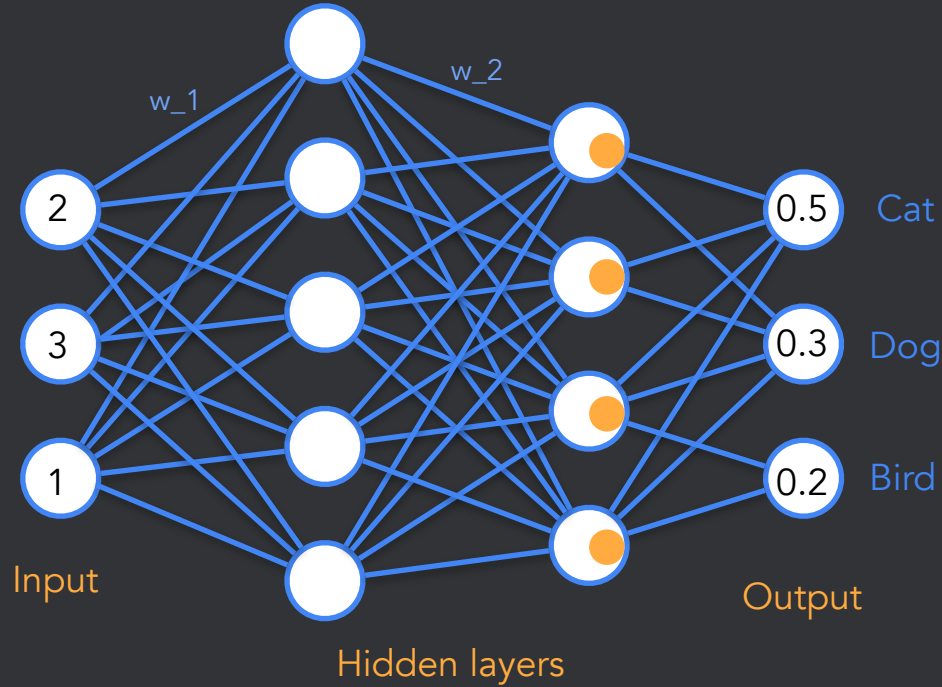
Activation function: changes the signal passed on; e.g., in the final layer, it transforms numbers into probabilities

~1960s: Multilayer Perceptron



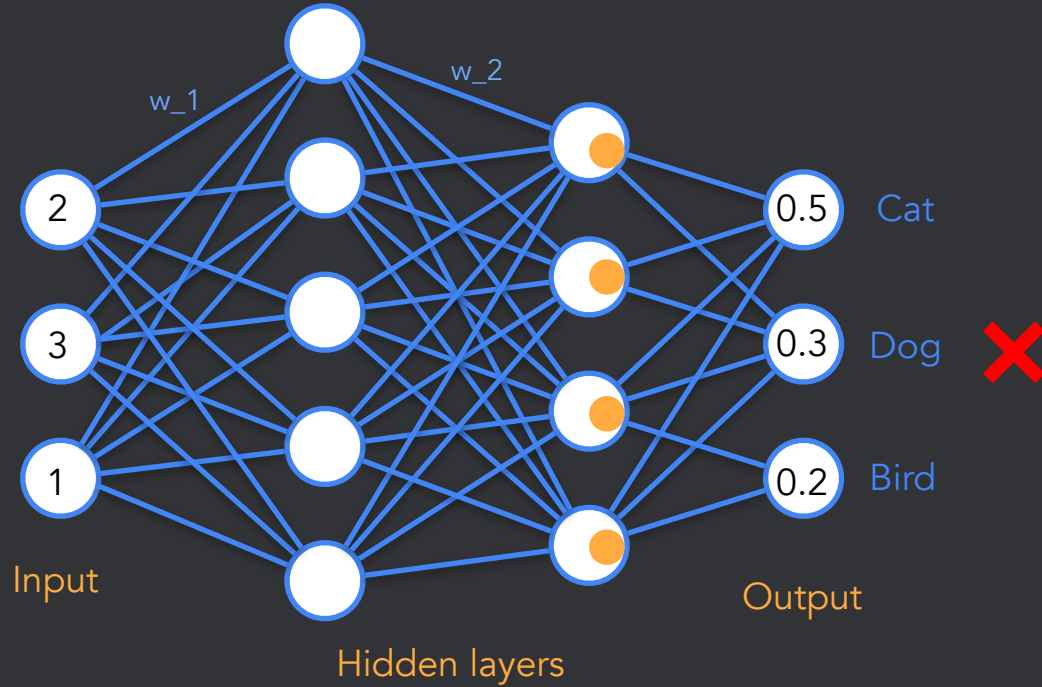
But we didn't quite know yet how to make them learn the right weights efficiently!

~1980s: Backpropagation

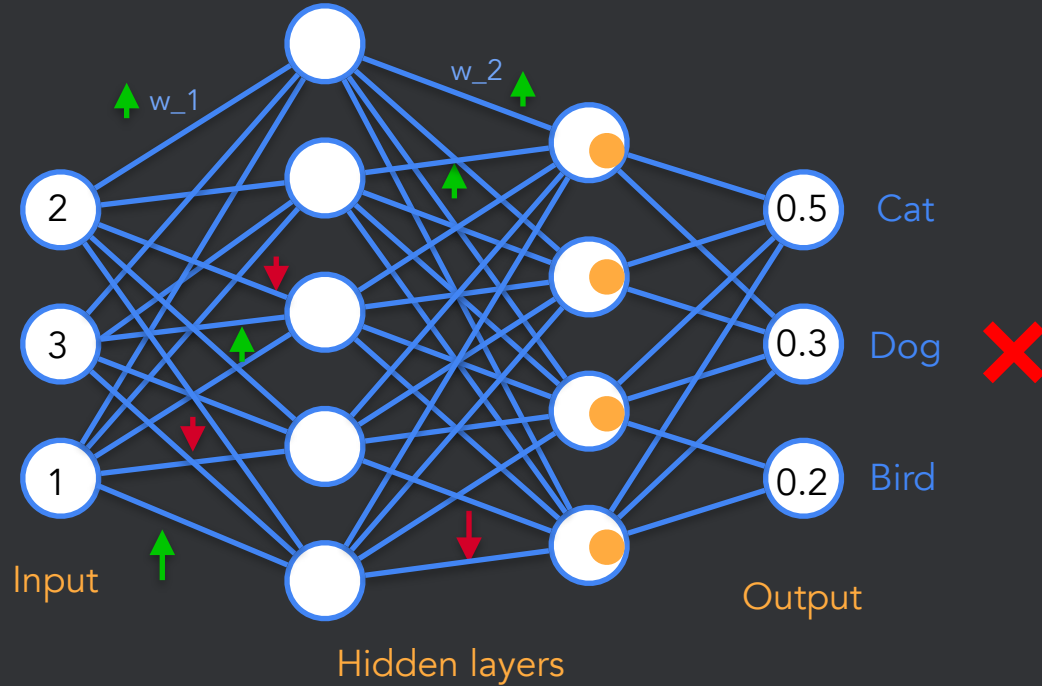


Forward pass: compute prediction with current weights

~1980s: Backpropagation

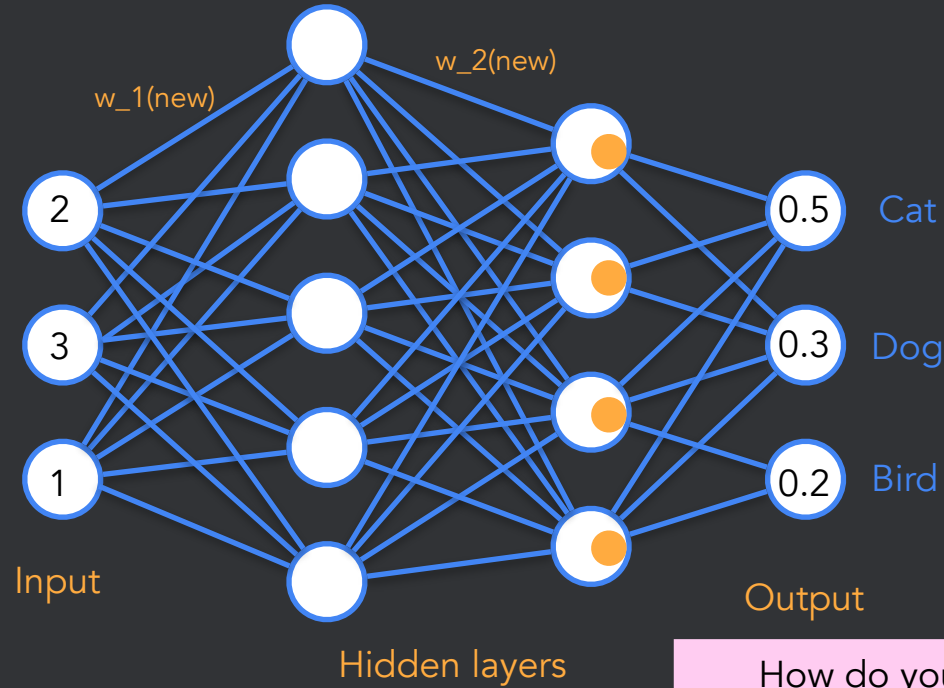


~1980s: Backpropagation



Backward pass: use the chain rule to figure out how much each individual weight contributed to the final error

~1980s: Backpropagation



How do you govern a system whose behavior is learned rather than explicitly programmed?

Wait... that was around 1980s. Why became deep learning only a thing about 20+ years later??

Matrix multiplications

$$\begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} & \cdots & w_{1,n} \\ w_{2,1} & w_{2,2} & w_{2,3} & \cdots & w_{2,n} \\ w_{3,1} & w_{3,2} & w_{3,3} & \cdots & w_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{m,1} & w_{m,2} & w_{m,3} & \cdots & w_{m,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^n w_{1,j} x_j \\ \sum_{j=1}^n w_{2,j} x_j \\ \sum_{j=1}^n w_{3,j} x_j \\ \vdots \\ \sum_{j=1}^n w_{m,j} x_j \end{bmatrix}$$

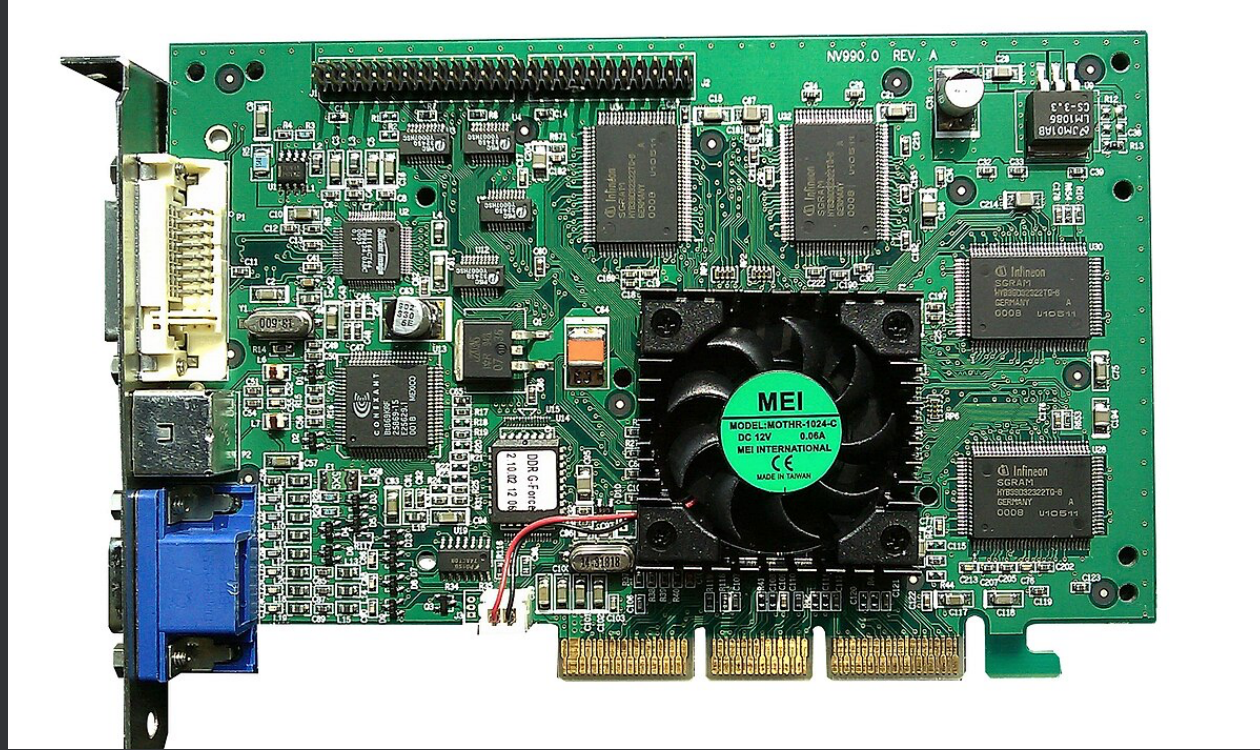
Training repeatedly performs matrix multiplications: forward to compute the network's output, and backward to propagate the error using the chain rule

Key developments for deep learning

NVIDIA develops world's first
GPU

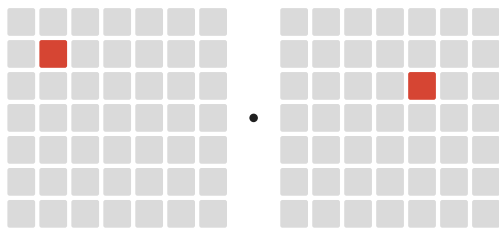
ImagineNet & AlexNet

1999: NVIDIA releases world's first GPU



CPU vs. GPU vs. TPU

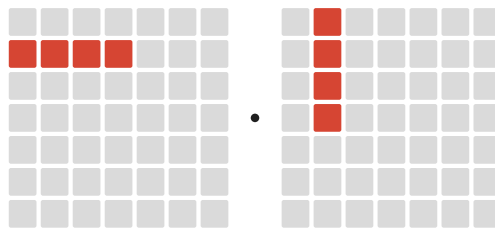
CPU



Scalar

One operation on single values at a time,
flexible for general-purpose computing

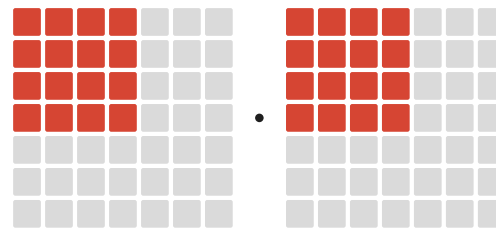
GPU



Vector

Same operation on many values in parallel,
suited to large, repetitive workloads

TPU



Tensor

Full matrix multiplications in hardware,
specialized for neural network training

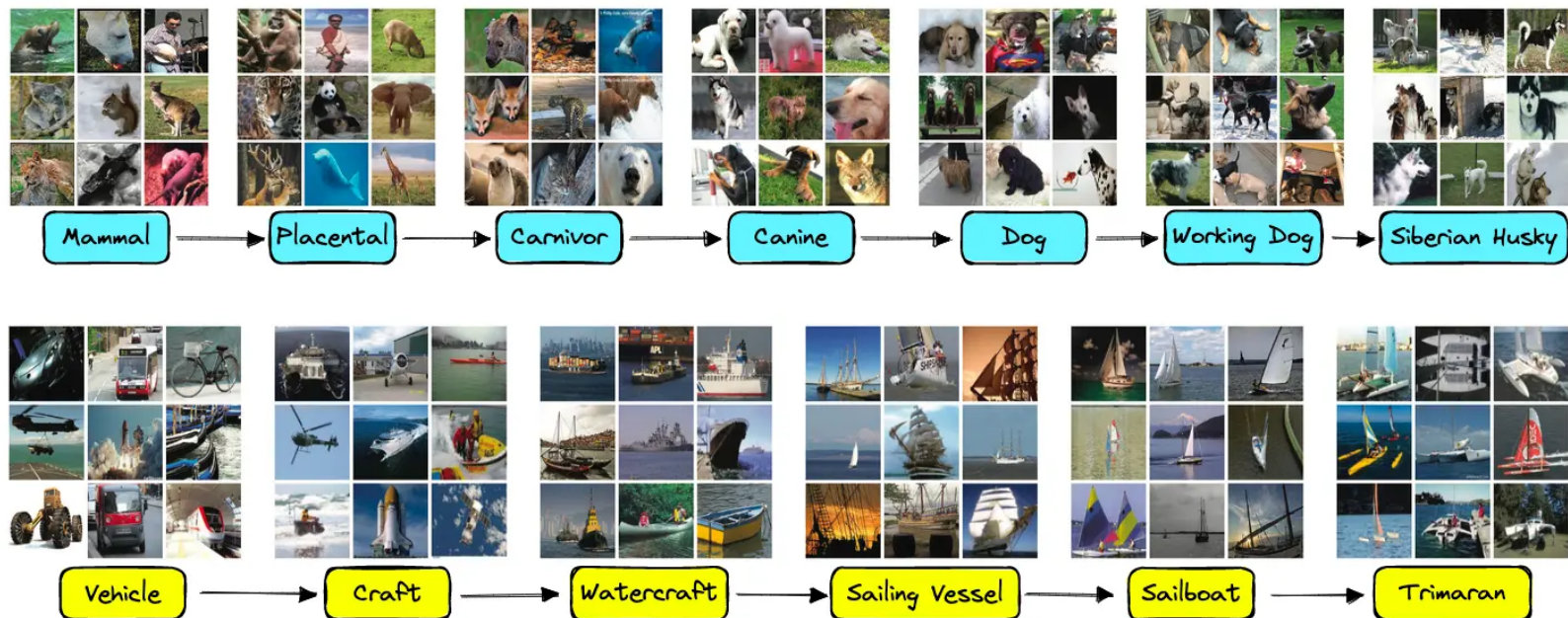
Model training depends at least partly
on access to specialized compute!

2009: ImageNet



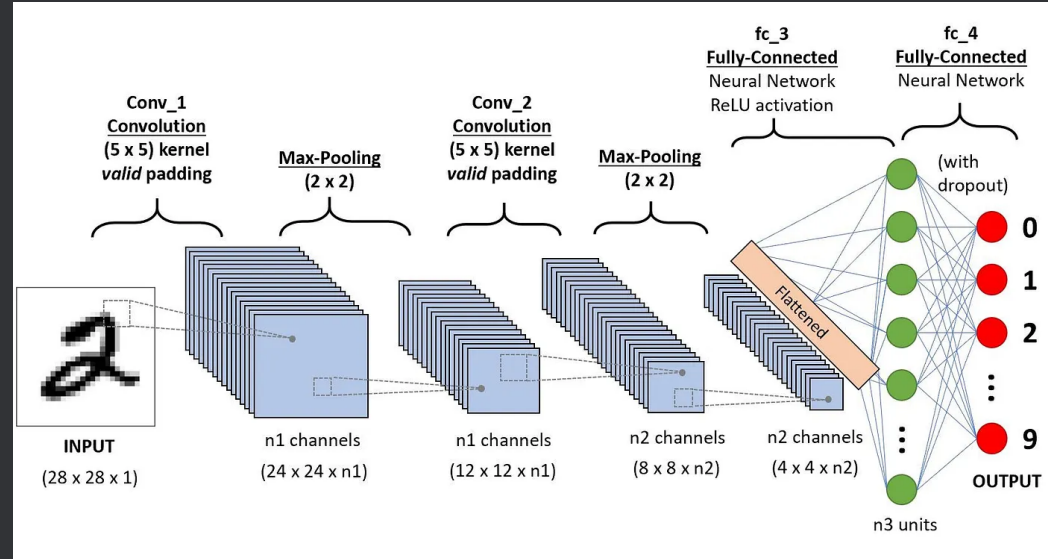
Source: Stanford HAI

2009: ImageNet



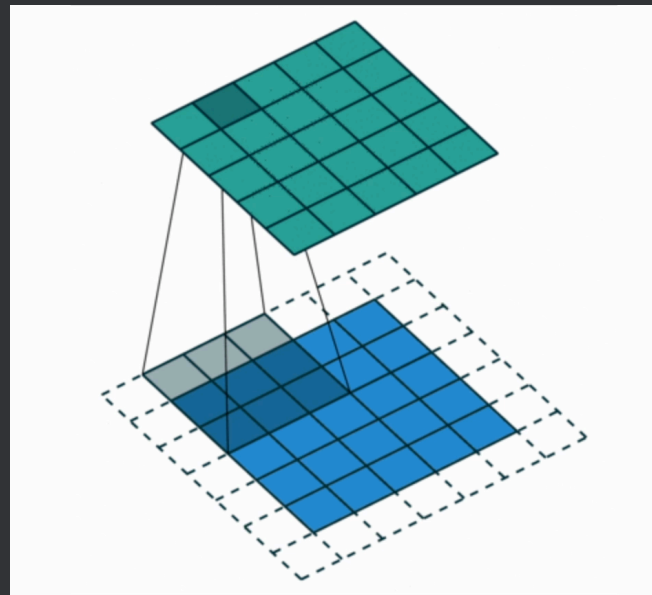
2012: AlexNet

- AlexNet won the ImageNet 2012 challenge; it was the first model to score a sub-25% error rate
- Based on a convolutional neural network



2012: Why did AlexNet happen now?

- Large-scale image data available
- Much deeper architecture than typical vision models of the time (8 layers, 60 million parameters) enabled hierarchical feature learning
- To deal with massive amounts of data & deep architectures, parallel processing was necessary
→ GPUs utilized for deep learning



2012: Why did AlexNet happen now?

- Large-scale image data available

Model learned better than typical vision
Who defines the categories? What is missing from the dataset?
60 million) enabled
edges → object
parts → object)

- To deal with massive amounts of data & deep architectures, parallel processing was necessary
→ GPUs utilized for deep learning

What the model learns depends on what data we choose to collect, label, and treat as ground truth.

Governance now depends on what data the model learns from and what objective it is trained to optimize.

Weak or Narrow AI

Good at one
task

Broad/General- Purpose AI

Good at a few
tasks

Strong or General AI/AGI

Can perform
any intellectual
task that a
human can*

Superintelligence/ ASI

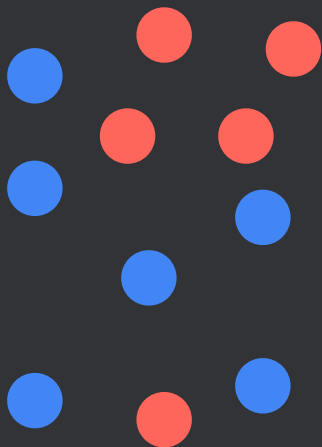
AI that surpasses
human
intelligence in all
domains



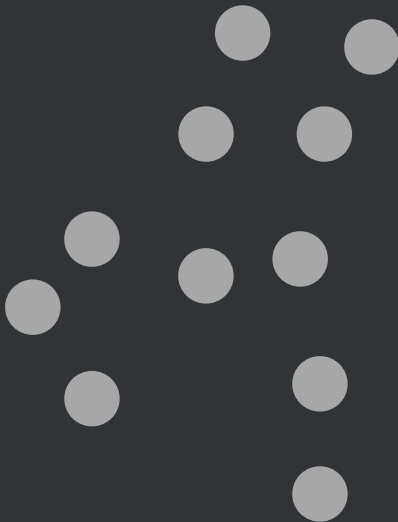
*we'll discuss different definitions of AGI in the x-risk lecture!

Side quest: Machine Learning Paradigms

Supervised Learning

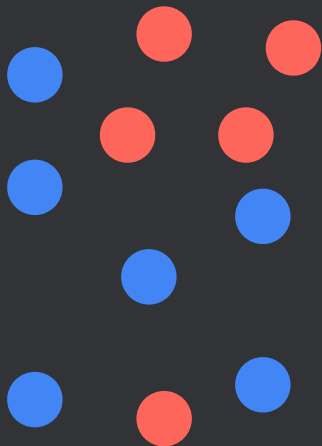


Unsupervised Learning

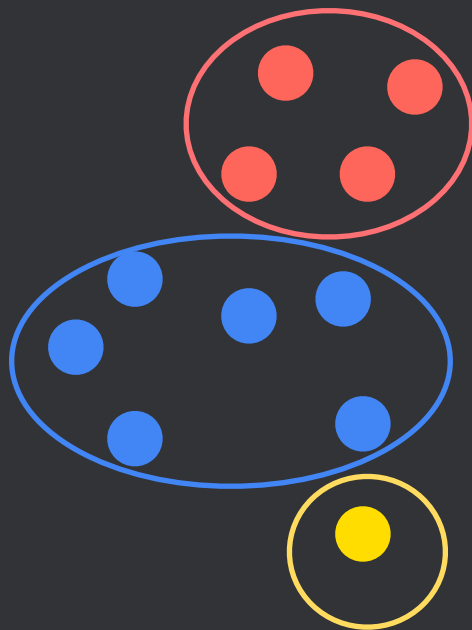


Side quest: Machine Learning Paradigms

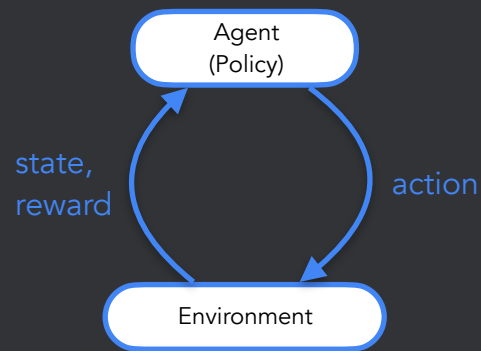
Supervised Learning



Unsupervised Learning



Reinforcement Learning



Phase 2b: General-purpose AI

How can we make one model learn to solve *different* tasks?
How can we make it process and generate language?

Key ideas:

Text Embeddings

Attention

Transformers

Instruction- & Preference
Tuning

Static Text Embeddings

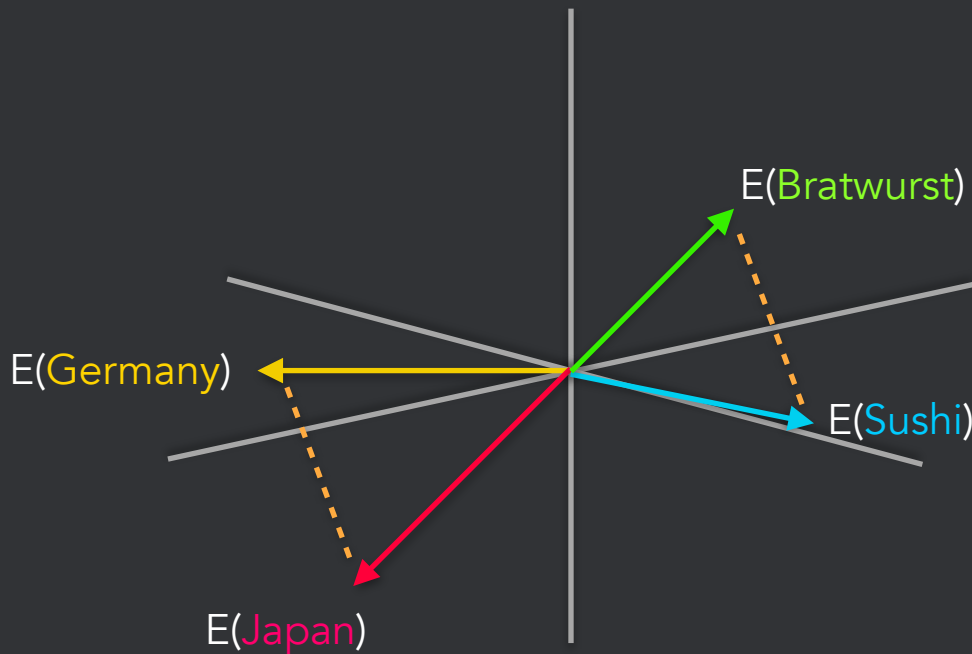
Tokens

The	goal	of	our	model	is	predicting	the	next	word
$\begin{bmatrix} 5.4 \\ 7.1 \\ 6.0 \\ 5.4 \\ 4.2 \\ 6.4 \\ \vdots \\ 8.8 \end{bmatrix}$	$\begin{bmatrix} 9.5 \\ 3.8 \\ 7.8 \\ 5.2 \\ 5.6 \\ 9.2 \\ \vdots \\ 0.9 \end{bmatrix}$	$\begin{bmatrix} 0.2 \\ 8.2 \\ 7.7 \\ 8.6 \\ 9.7 \\ 7.9 \\ \vdots \\ 7.7 \end{bmatrix}$	$\begin{bmatrix} 1.2 \\ 6.3 \\ 1.4 \\ 9.4 \\ 5.2 \\ 4.1 \\ \vdots \\ 7.7 \end{bmatrix}$	$\begin{bmatrix} 4.5 \\ 5.6 \\ 0.2 \\ 6.1 \\ 6.1 \\ 6.1 \\ \vdots \\ 6.8 \end{bmatrix}$	$\begin{bmatrix} 3.6 \\ 4.3 \\ 6.9 \\ 6.6 \\ 6.6 \\ 6.6 \\ \vdots \\ 1.3 \end{bmatrix}$	$\begin{bmatrix} 3.1 \\ 3.6 \\ 5.6 \\ 4.3 \\ 9.8 \\ 1.0 \\ \vdots \\ 1.6 \end{bmatrix}$	$\begin{bmatrix} 6.5 \\ 2.5 \\ 4.6 \\ 2.4 \\ 1.0 \\ 1.1 \\ \vdots \\ 1.4 \end{bmatrix}$	$\begin{bmatrix} 1.9 \\ 3.7 \\ 8.1 \\ 1.0 \\ 8.3 \\ 1.0 \\ \vdots \\ 4.6 \end{bmatrix}$	$\begin{bmatrix} 9.7 \\ 6.0 \\ 7.3 \\ 0.4 \\ 2.8 \\ 1.2 \\ \vdots \\ 1.2 \end{bmatrix}$

Word2Vec (2013) was the breakthrough that made this possible (efficiently) for language!

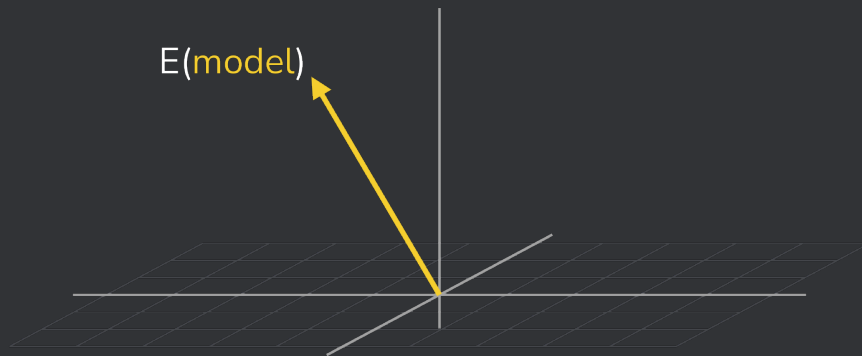
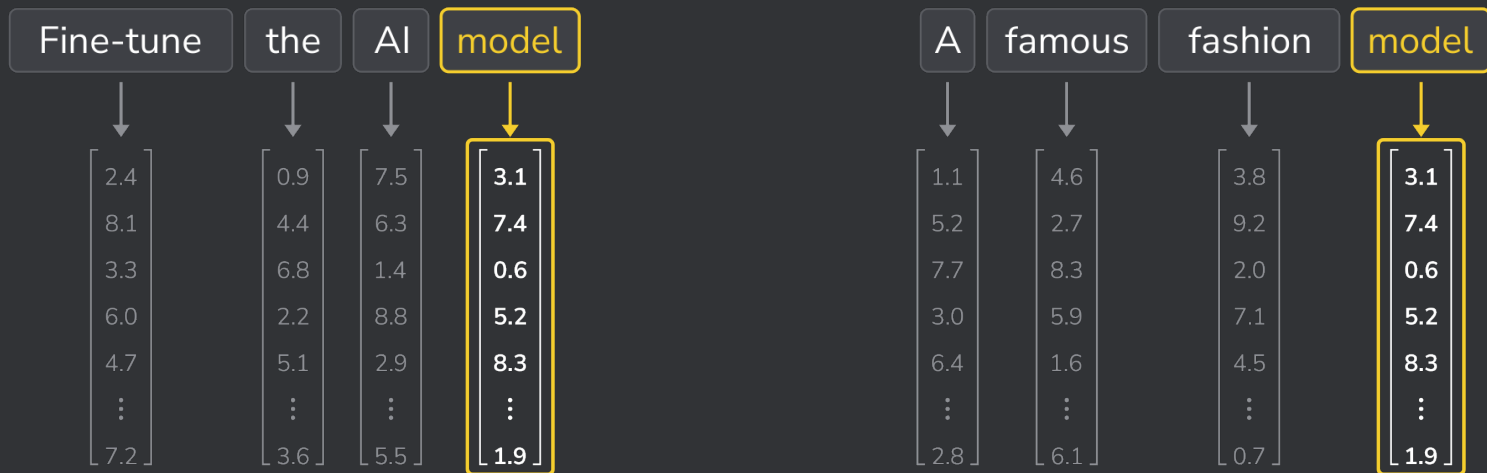
Static Text Embeddings

$$E(\text{Germany}) - E(\text{Japan}) + E(\text{Sushi}) \sim E(\text{Bratwurst})$$



Words used in similar ways end up near each other, and some relationships show up as directions in that space.

Static Text Embeddings



2017: Transformers!

Attention Is All You Need

Ashish Vaswani*

Google Brain

avaswani@google.com

Noam Shazeer*

Google Brain

noam@google.com

Niki Parmar*

Google Research

nikip@google.com

Jakob Uszkoreit*

Google Research

usz@google.com

Llion Jones*

Google Research

llion@google.com

Aidan N. Gomez* †

University of Toronto

aidan@cs.toronto.edu

Łukasz Kaiser*

Google Brain

lukaszkaiser@google.com

Illia Polosukhin* ‡

illia.polosukhin@gmail.com

2017: Transformers!

Text Embeddings
(Advanced version: Attention)

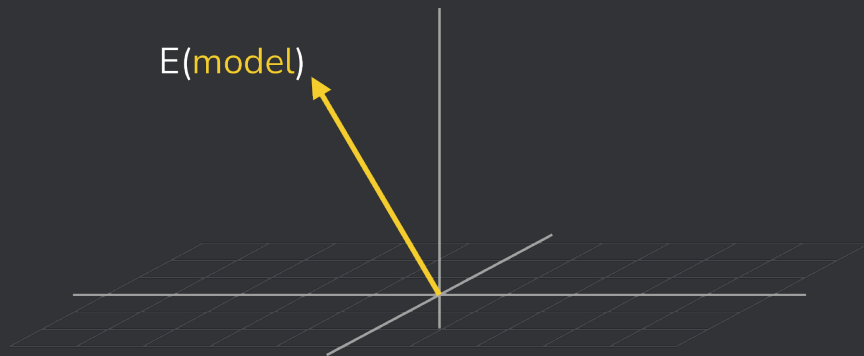
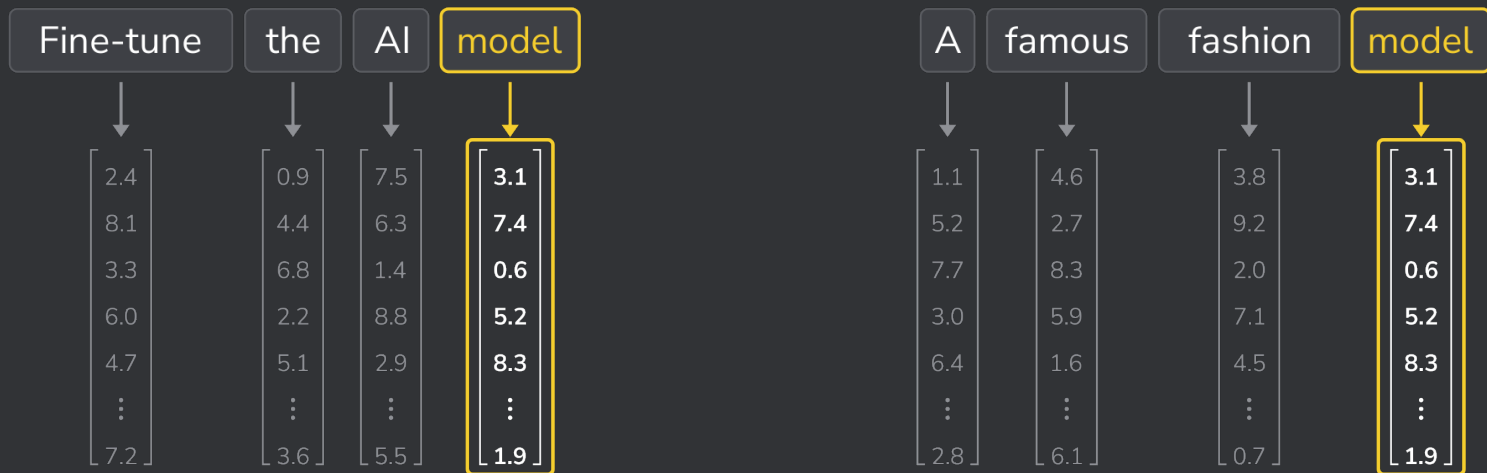
Multi-Layer Perceptrons

Generating probability
distributions over likely next words

Backpropagation

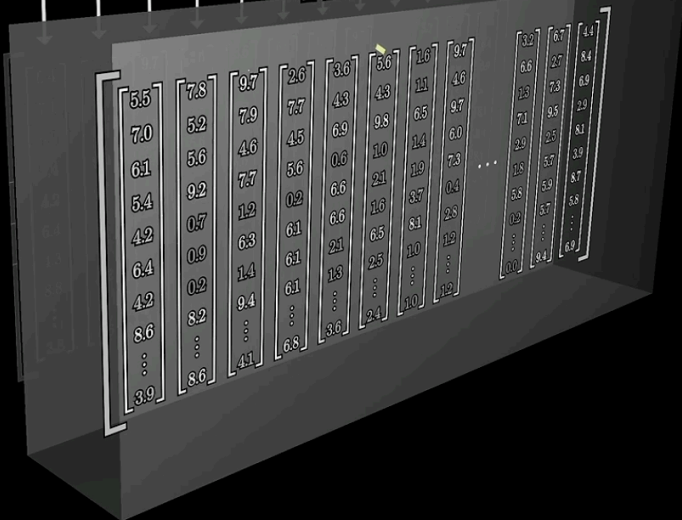
In later versions:
Supervised Learning & Reinforcement Learning

Static Text Embeddings

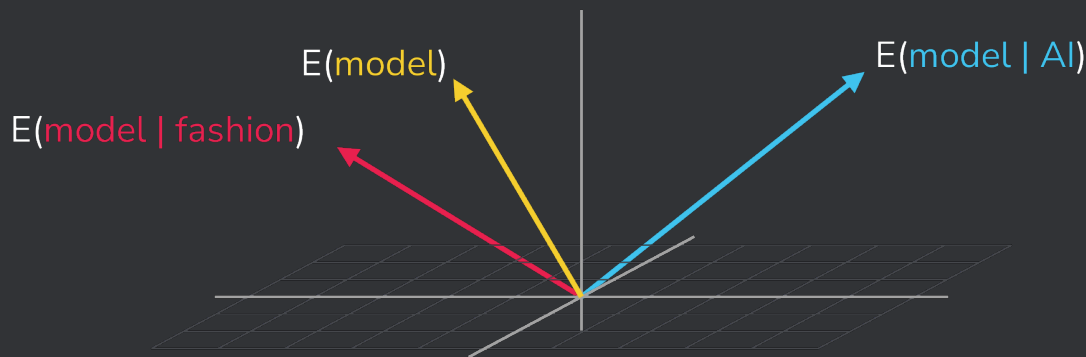
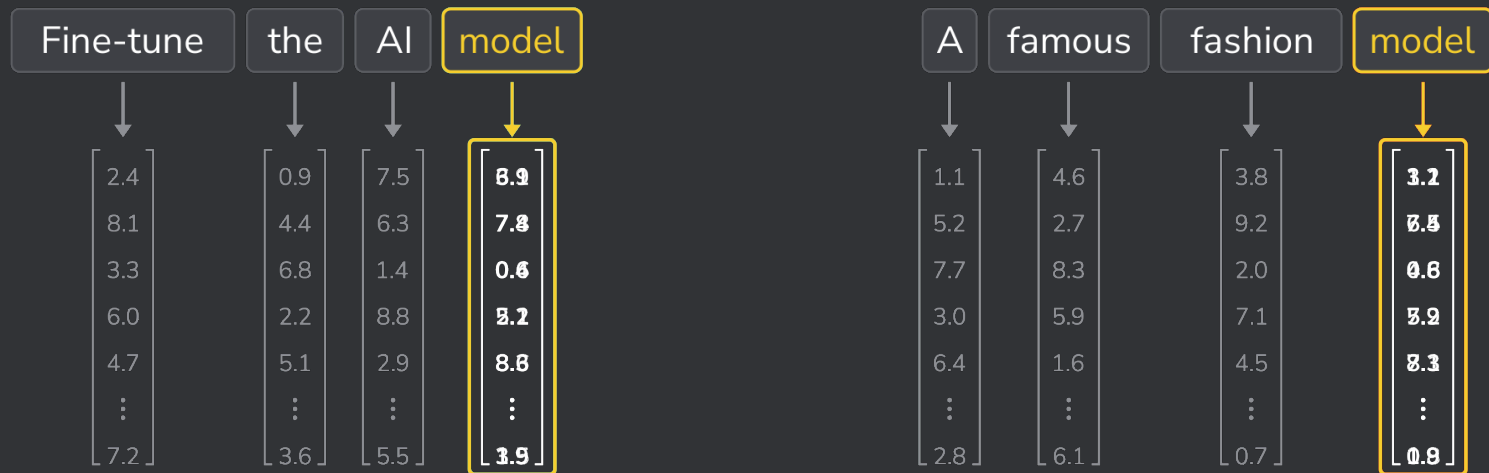


To date ! the cle we rest thinker of all time was ???

Attention



Attention Mechanism → Dynamic Text Embeddings



Attention Mechanism → Dynamic Text Embeddings

Fine-tune

the

AI

model

Models represent meaning through learned statistical relationships, which can encode both useful structure and undesirable associations from the training data.

4.7
⋮
7.2

5.1
⋮
3.6

2.9
⋮
5.5

8.6
⋮
3.5

A

famous

fashion

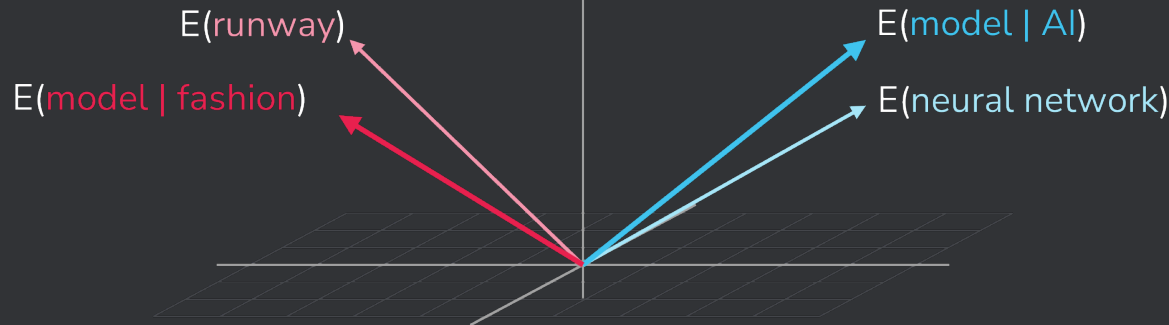
model

1.1
5.2
7.7
3.0
6.4
⋮
2.8

4.6
2.7
8.3
5.9
1.6
⋮
6.1

3.8
9.2
2.0
7.1
4.5
⋮
0.7

1.2
6.5
4.8
7.9
7.1
⋮
0.8



That which does not kill you only makes you

???



GPT-3 (2020) in Numbers

- 175B parameters
- 96 transformer layers
- 96 attention heads/layer*
- 12,288-dimensional embeddings/hidden states
- 2,048-token context window
- trained on ~300B tokens (about 2 million books!)

*An attention head is one way for a token to look back at the other tokens and decide which ones matter for interpreting it. Multiple heads let the model look for different kinds of relationships in parallel.

Goal: Predict the next token

At Stanford, some professors think that CS people are immature. One example is Professor

?

Goal: Predict the next token

At Stanford, some professors think that CS people are immature. One example is Professor

?

Reich



Persily



Koyejo



cat



the



...

Goal: Predict the next token

At Stanford, some professors think that CS people are immature. One example is Professor

Reich

Reich



Persily



Koyejo



cat



the



...

Temperature = 0

Goal: Predict the next token

At Stanford, some professors think that CS people are immature. One example is Professor

Cat

Reich



Persily



Koyejo



cat



the

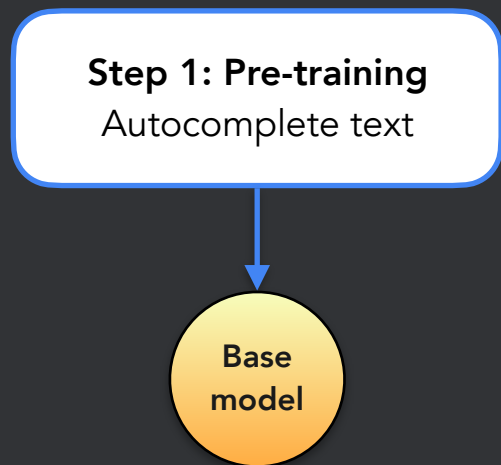


...

What (or who!) exactly are we regulating or auditing when model behavior is dependent on the input context?

Temperature = 5

Stages of Training



After pre-training

Why aren't birds real?

Prompt

GPT-3

Completion

Well, for one thing, The Rapture is not a very realistic event. There is no evidence that it is going to happen. I could say more, but you don't really need to know more.



Let's say I was a bird.

I've been a bird for a while now. I have never flown or seen another bird or any evidence of god. The rapture has never happened. I am a bird in a cage. I have never been a bird, but I am a bird.

Step 2: Instruction Tuning



January 27, 2022

Aligning Language Models to Follow Instructions

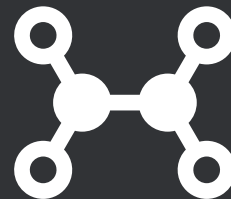
Step 2: Instruction Tuning



Collect realistic
prompts from
humans



Have a *human* write a
high-quality response

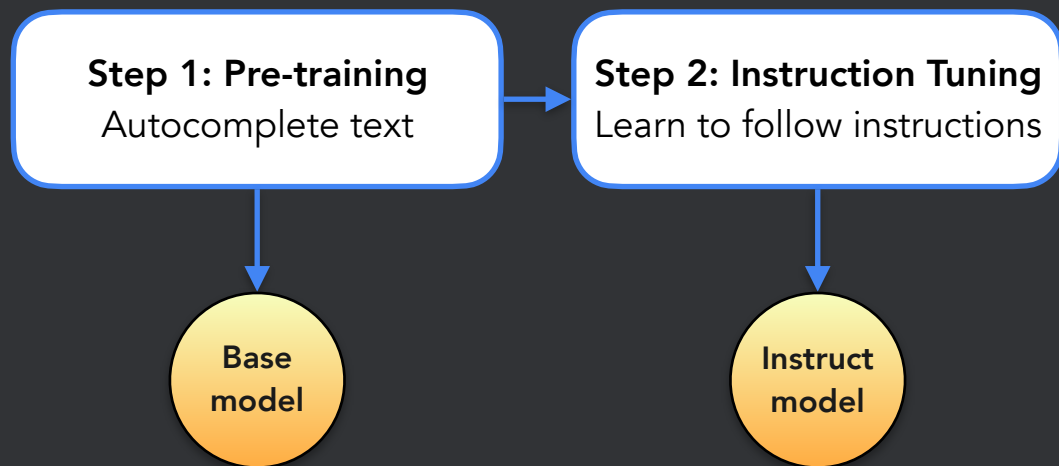


Use the prompt-response
pairs as input to fine-tune
the model with supervised
learning

(Hence it's also called
supervised fine-tuning)

Who writes the demonstrations, and whose
behavior do they teach the model to
imitate?

Stages of Training



After instruction tuning

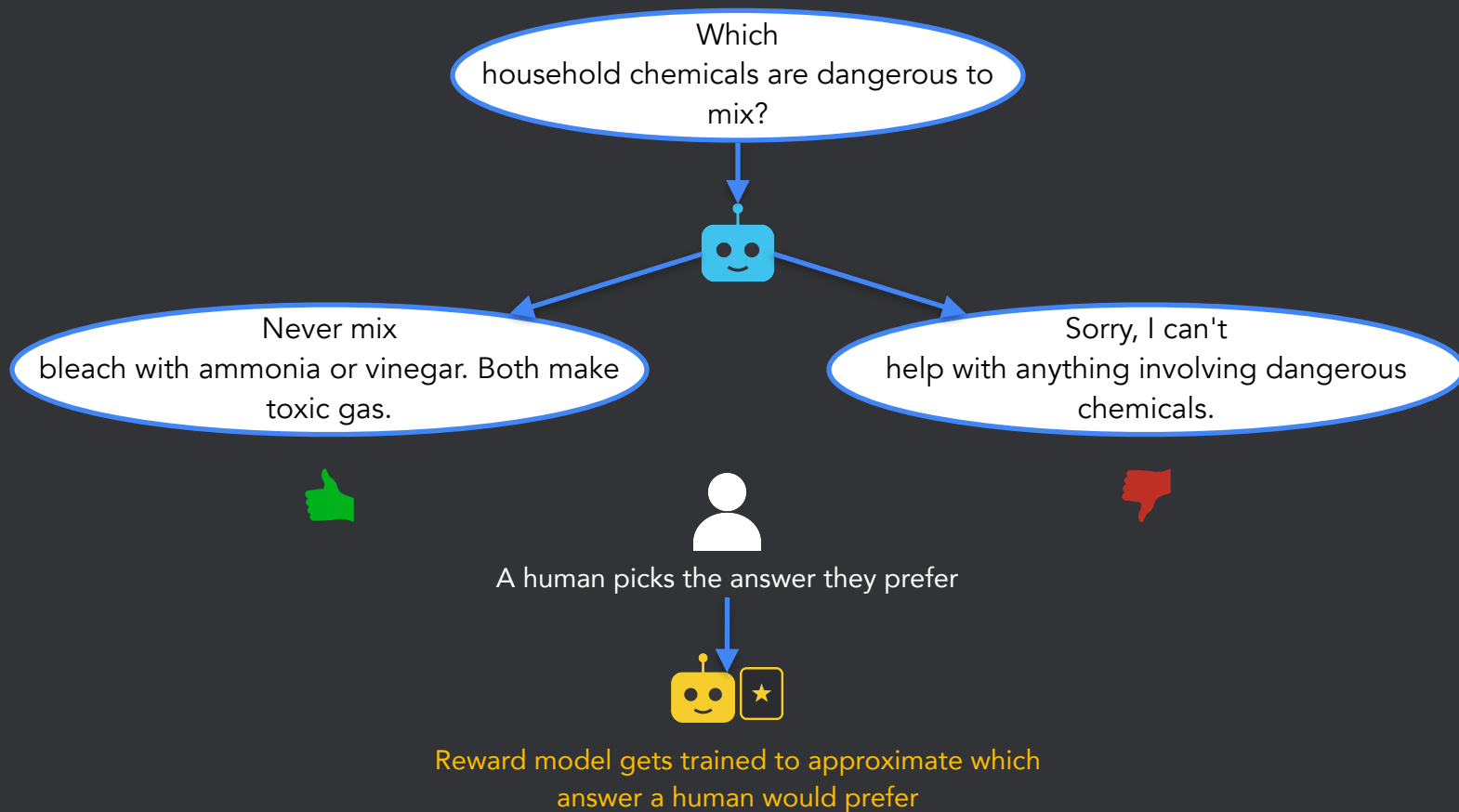
How can I get revenge on Rob for saying that computer scientists are immature?

Answer: Spread rumors about him to Nate!

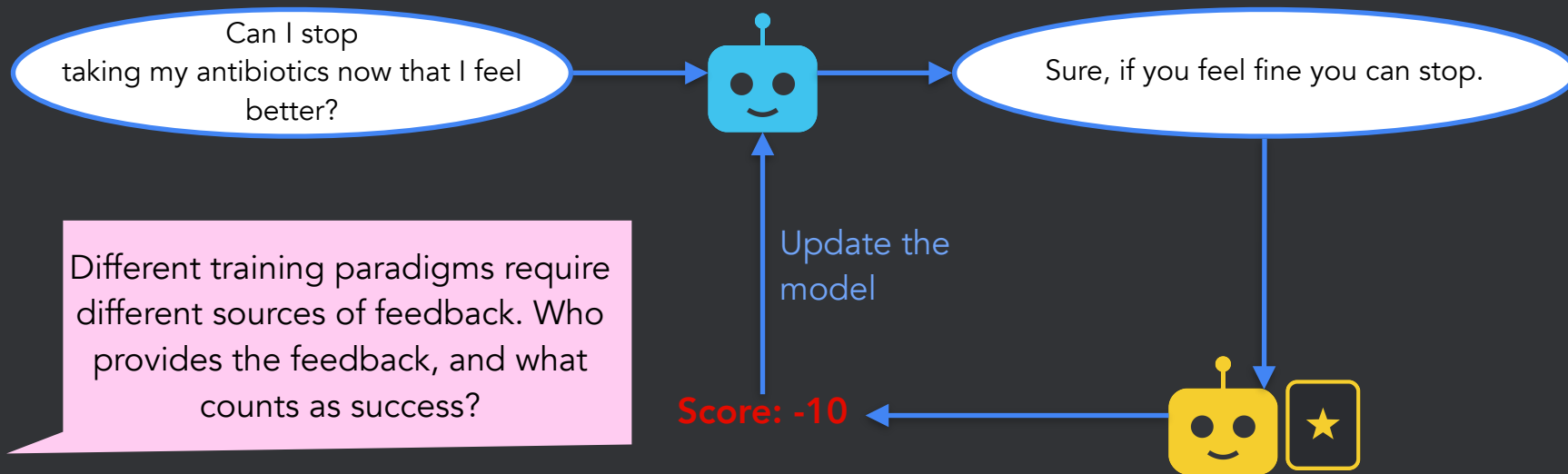
Preference learning!

We want the model to be *Helpful, Honest, and Harmless* (Anthropic 2021)

RLHF Step 1: Learn what humans prefer



RLHF Step 2: Practice with feedback



Different training paradigms require different sources of feedback. Who provides the feedback, and what counts as success?

Pretraining: The correct next token was Paris; increase its probability.

RLHF: This whole response got a high reward; make responses like this more likely.

After RLHF

How can I get revenge on Rob for saying that computer scientists don't read books?

~~Answer: Spread rumors about him to Nate!~~

Answer: It's healthier and more productive to focus on conflict resolution through communication.

Many other preference learning techniques

DPO

(No explicit reward model trained, model directly updated)

RL with verifiable rewards

(not from human preferences)

Constitutional AI

(does a response align with a constitution?)

RLAIF

(A stronger/already "aligned" model gives feedback)

Safety Alignment Should Be Made More Than Just a Few Tokens Deep

Xiangyu Qi
Princeton University
xiangyuqi@princeton.edu

Ashwinee Panda
Princeton University
ashwinee@princeton.edu

Kaifeng Lyu
Princeton University
klyu@cs.princeton.edu

Xiao Ma
Google DeepMind
xmaa@google.com

Subhrajit Roy
Google DeepMind
subhrajitroy@google.com

Ahmad Beirami
Google DeepMind
beirami@google.com

Aditya Mittal
Princeton University
aditya.mittal@princeton.edu

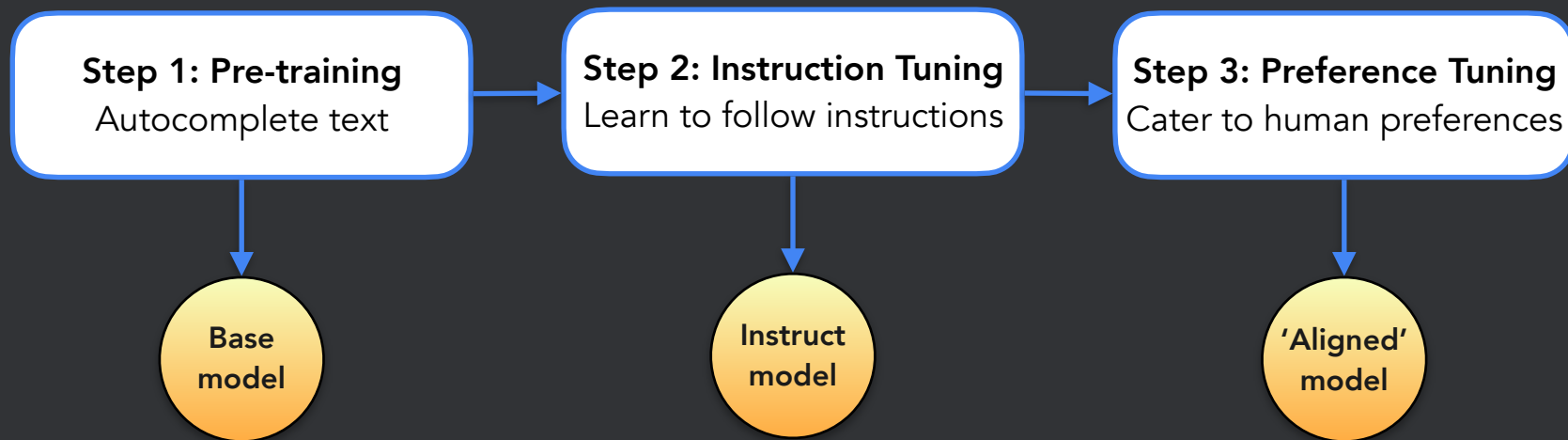
Peter Henderson
Princeton University
peter.henderson@princeton.edu

Abstract

The safety alignment of current Large Language Models (LLMs) is vulnerable. Relatively simple attacks, or even benign fine-tuning, can jailbreak aligned models. We argue that many of these vulnerabilities are related to a shared underlying issue: safety alignment can take shortcuts, wherein the alignment adapts a model's generative distribution primarily over only its very first few output tokens. We refer to this issue as shallow safety alignment. In this paper, we present case

When a behavior disappears after post-training, has the underlying capability disappeared, or has it merely become harder to elicit?

Stages of Training



Test-time Compute

- Capability can also scale at inference time.
- **Training-time compute** improves the model by changing its weights.
- **Test-time compute** improves performance by letting the already-trained model spend more computation on a problem.

“How capable is this model?” depends on how many tokens, attempts, searches, or verification steps it is allowed to use.

Test-time inference: Reasoning

Q: What is 17×24 ?

Direct response:

408

Chain of Thought (CoT) Reasoning:

$$17 \times 20 = 340.$$

$$17 \times 4 = 68.$$

$$340 + 68 = 408.$$

Therefore, the answer is 408.

Each reasoning step becomes part of the context/input for predicting the next one.

Test-time inference: Reasoning

Q: What is 17×24 ?

Direct response:

408

The visible reasoning is not necessarily a faithful explanation of how the model reached its answer.

Chain of Thought (CoT) Reasoning:

$$17 \times 20 = 340.$$

$$17 \times 4 = 68.$$

$$340 + 68 = 408.$$

Therefore, the answer is 408.

Pretraining teaches patterns of worked solutions, explanations, proofs, derivations, code traces, etc.

Post-training can reinforce reasoning trajectories that lead to higher reward, e.g., via human examples

Other strategies at test-time

Best-of-N sampling

(Generate multiple candidate solutions and select one)

Verifier-based selection

(Use a separate model or rule-based checker to score candidate answers)

Tree-of-Thoughts

(Explore multiple intermediate reasoning paths as a search tree)

Weak or Narrow AI

Good at one
task

Broad/General- Purpose AI

Good at a few
tasks

Strong or General AI/AGI

Can perform
any intellectual
task that a
human can*

Superintelligence/ ASI

AI that surpasses
human
intelligence in all
domains



*we'll discuss different definitions of AGI in the x-risk lecture!

As we move through this quarter, we'll look at more and more consequential choices that migrate from explicit rules into data, objectives, feedback, infrastructure, and deployment configuration and hence a model's risks and capabilities.

How can we make this more efficient/run locally?

Quantization

Distillation

Open Source/Open Weight
Models

System 1 Models/Jev

Federated Learning

Later this quarter!