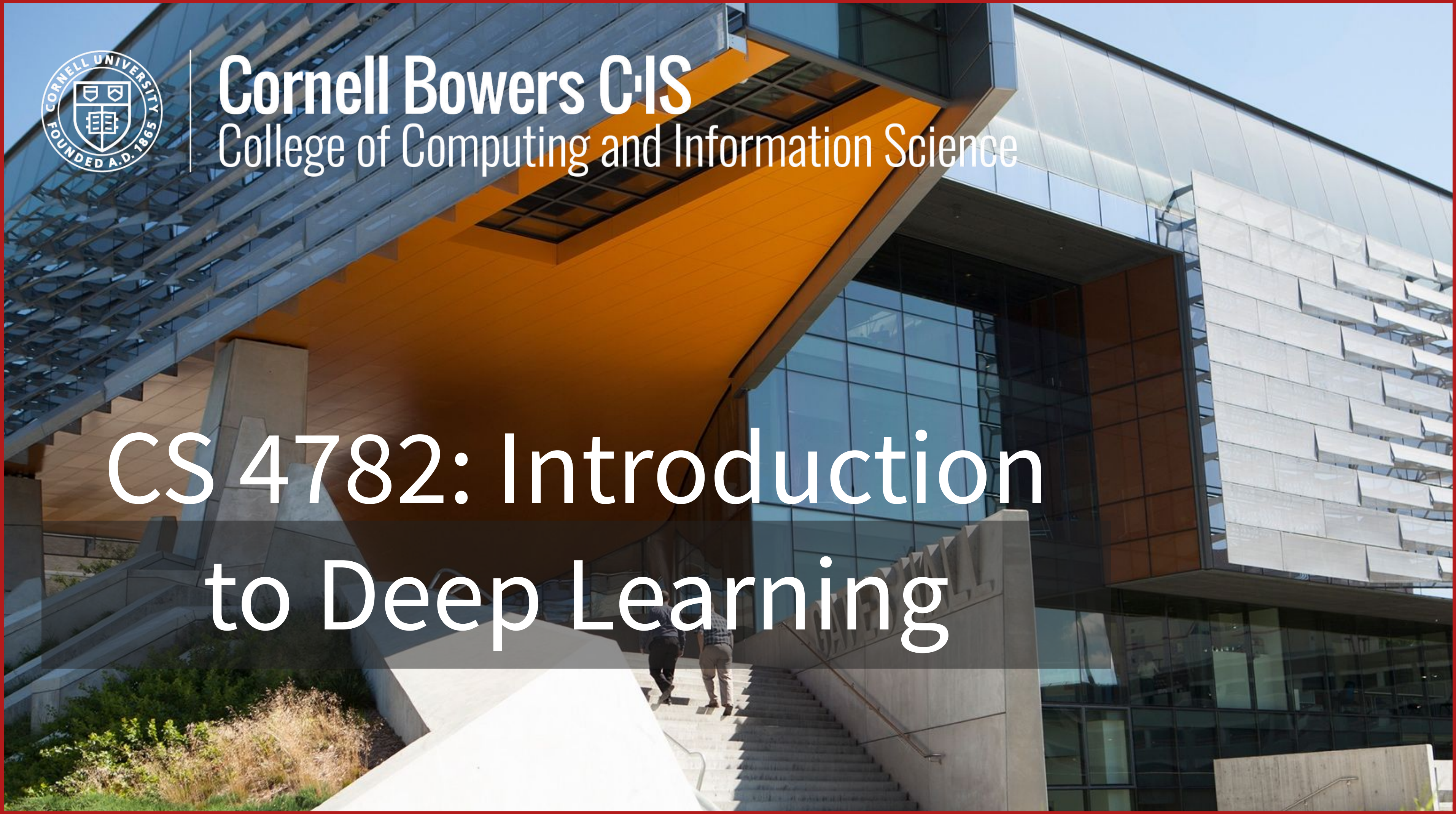
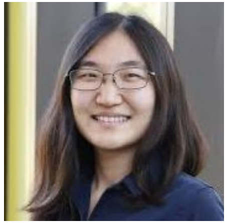


Cornell Bowers C·IS
College of Computing and Information Science

CS 4782: Introduction to Deep Learning

Cornell Bowers CIS

Course Staff

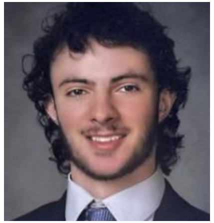


Prof. Jennifer J. Sun
Instructor

Prof. Kilian Q. Weinberger
Instructor

Žiga Kovačič
TA

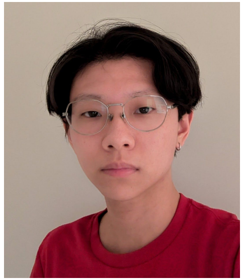
Adhitya Polavaram
TA



Sean Brynjólfsson
TA

Haozheng Yu
TA

Tyler King
TA

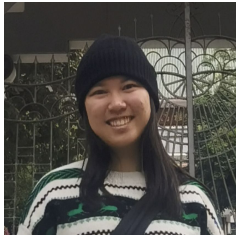


Lucas Li
TA

Course Staff

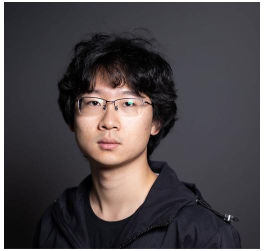
Jack Jansons
TA

Raphael Thesmar
TA



Vivian Chen
TA

Snehal Bhagat
TA



Youming Deng
TA

Selina Xiao
TA

Travis Zhang
TA

JJ Bai
TA

ML/AI Courses at Cornell

CS 3700: Foundations of AI Reasoning and Decision-Making

CS 3780: Introduction to Machine Learning

CS 4756: Robot Learning

CS 4670: Introduction to Computer Vision

CS 4744: Computational Linguistics I

CS 4789: Introduction to Reinforcement Learning

CS 4775: Computational Genetics and Genomics

CS 4740: Natural Language Processing

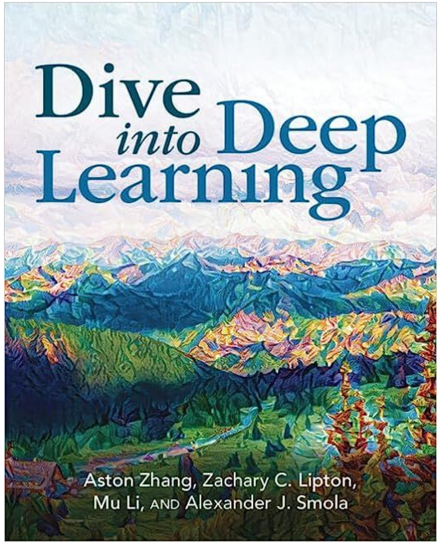
…

Logistics

- All lectures will be held in person
- Lectures will be on Tuesdays and Thursdays from 2:55 to 4:10pm
- Please attend and participate!!

Logistics

- Course website: https://www.cs.cornell.edu/courses/cs4782/
  - Tentative **schedule**, homework policies, grading policies, etc. are on the course page
- Slides / **Office hours** are on the course website!
- No **course book,** but we will link to **DiDL** chapters
- Hub that links to everything: **Canvas page**
- Questions / Answers: **Ed Discussion**
- Projects: **Google Colab** (You will get free credits)
- Course email address: **CS4782SP25@gmail.com**
- Notes will usually be printed



Rules

- Never email the instructors directly
  - Post privately on **Ed Discussion**
  - Or email CS4782SP25@gmail.com
- No **laptops/mobiles/smart devices** in class
- **Class Code of Conduct** applies
- **Projects:** teams up to 5
- **Homework**: teams up to 2

Grading (4782)

- Homework (35%)
  - There will be written assignments and coding projects
    - Google Cloud Credits for compute!
  - We recommend doing them in pairs!
  - 2-slip days for every assignment
- Mid-term exam (30%)
  - Will be similar to the homework assignments
- Project (30%)
  - Goal: familiarize yourself with deep learning libraries
  - Implement a method from a recent research paper and reproduce their results
- Participation (5%)
  - Attend classes!
  - Engage in class discussions and/or post on EdStem
  - Provide feedback to improve the class (we will reach out to you)

Grading (5782) [voluntary opt-in for 4782 students]

- Homework (30%)
  - There will be written assignments and coding projects
    - Google Cloud Credits for compute!
  - We recommend doing them in pairs!
  - 2-slip days for every assignment
- Mid-term exam (30%)
  - Will be similar to the homework assignments
- Project (25%)
  - Goal: familiarize yourself with deep learning libraries
  - Implement a method from a recent research paper and reproduce their results
- Participation (5%)
  - Attend classes!
  - Engage in class discussions and/or post on EdStem
  - Provide feedback to improve the class (we will reach out to you)
- **Paper Quizzes (10%)**
  - **Read specified research papers**
  - **Answer quizzes on Canvas**

Academic Integrity

- Do not disclose exact solutions to members from other groups for assignments
  - High-level discussion is allowed
- Cite any external sources
- You can use ChatGPT/Gemini/other AI assistants
  - But you **must add a note** explaining what you used it for and how you used it

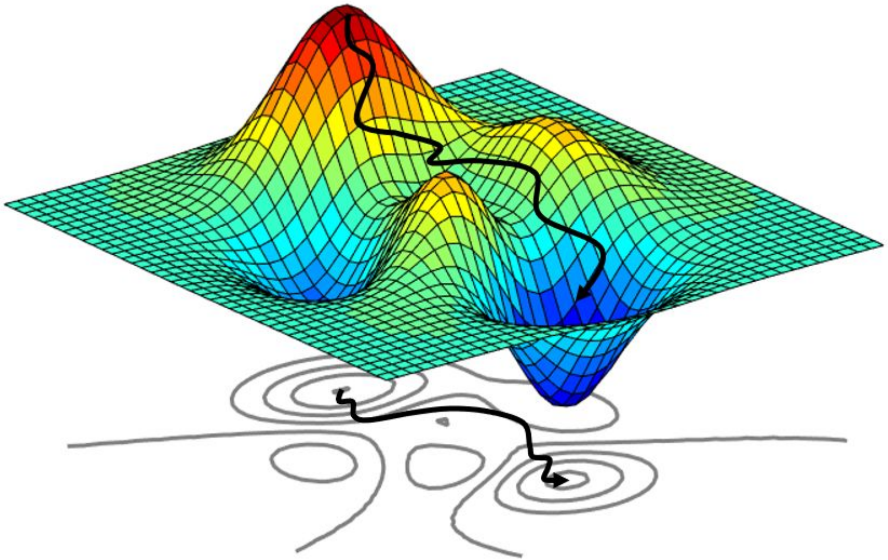
Course Objectives

By the end of the course you will be able to…

1. Design, train, and evaluate deep neural networks
2. Apply deep learning techniques to solve real-world problems in computer vision, natural language processing, and other complex domains
3. Critically evaluate pros/cons of different model architectures
4. Read and understand research in deep learning
5. Understand the core design principles behind leading deep learning systems like GPT-4, DALL-E 2/3, and Stable Diffusion

Training Neural Networks

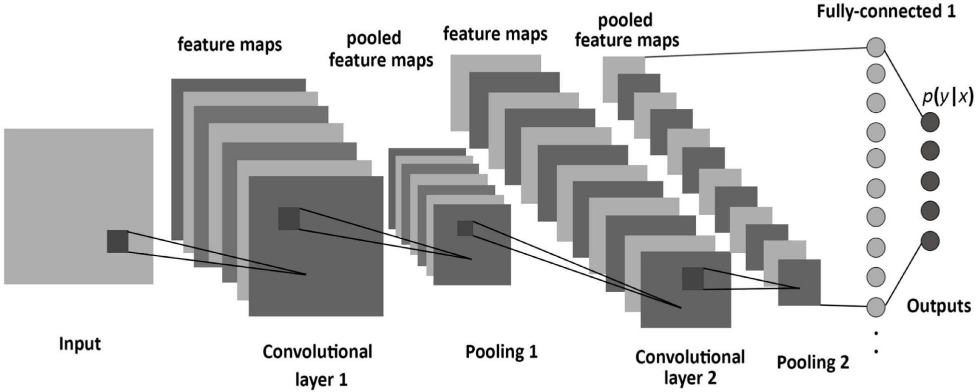
- Optimization algorithms - gradient descent, SGD, AdaGrad, Adam
- Learning rate scheduling
- Hyperparameter Optimization
- Regularization



https://towardsdatascience.com/an-introduction-to-surrogate-optimization-intuition-illustration-case-study-and-the-code-5d9364aed51b

Computer Vision

- Convolutional neural networks
- Different convolutional architectures - vanilla CNN, LeNet, ResNet, DenseNets



https://www.mdpi.com/1099-4300/19/6/242

CVPR 2018 WAD Video Segmentation

Natural Language Processing

- Word Embeddings
- Recurrent Neural Networks
  - RNNs/ LSTMs
- Attention and Transformers
- Large Language Models (LLMs)

Explaining a Joke

Input: Did you see that Google just hired an eloquent whale for their TPU team? It showed them how to communicate between two different pods!

https://arxiv.org/abs/2204.02311

Natural Language Processing

- Word Embeddings
- Recurrent Neural Networks
  - RNNs/ LSTMs
- Attention and Transformers
- Large Language Models (LLMs)

Explaining a Joke

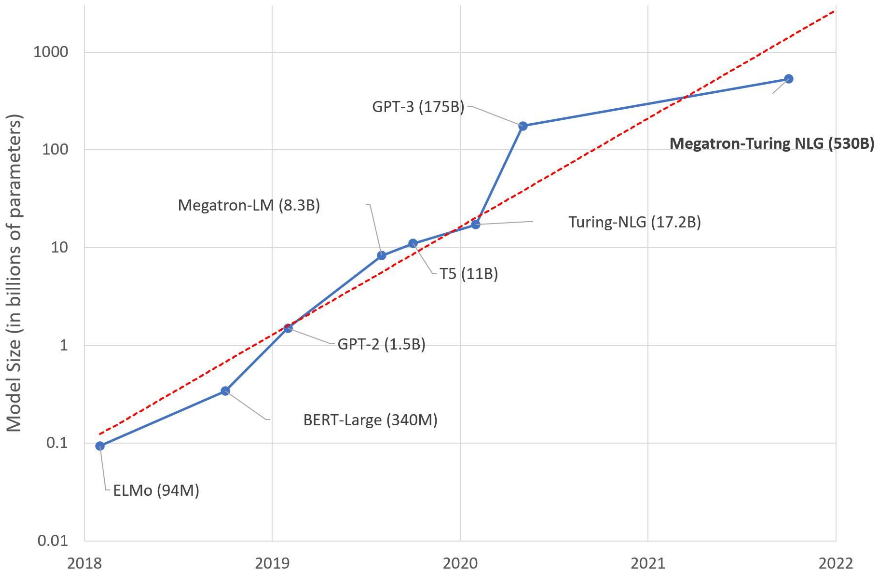
Input: Did you see that Google just hired an eloquent whale for their TPU team? It showed them how to communicate between two different pods!

Model Output: TPUs are a type of computer chip that Google uses for deep learning. A "pod" is a group of TPUs. A "pod" is also a group of whales. The joke is that the whale is able to communicate between two groups of whales, but the speaker is pretending that the whale is able to communicate between two groups of TPUs.

https://arxiv.org/abs/2204.02311

Natural Language Processing

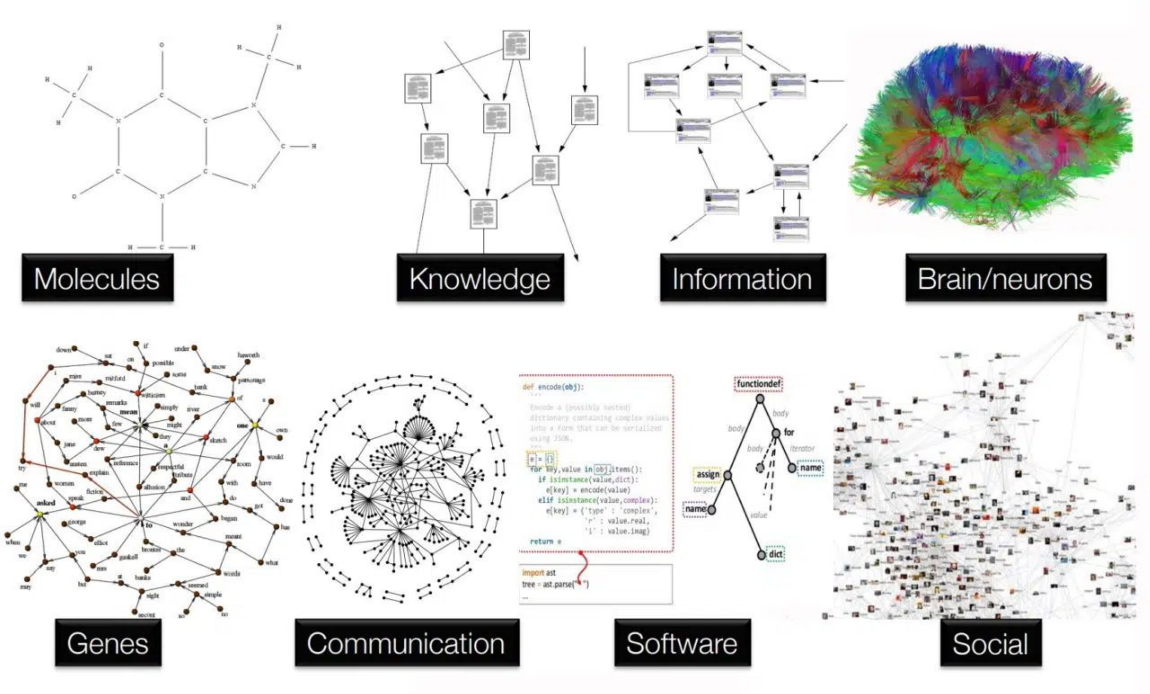
- Word Embeddings
- Recurrent Neural Networks
  - RNNs/ LSTMs
- Attention and Transformers
- Large Language Models (LLMs)



https://huggingface.co/blog/large-language-models

Graph Neural Networks

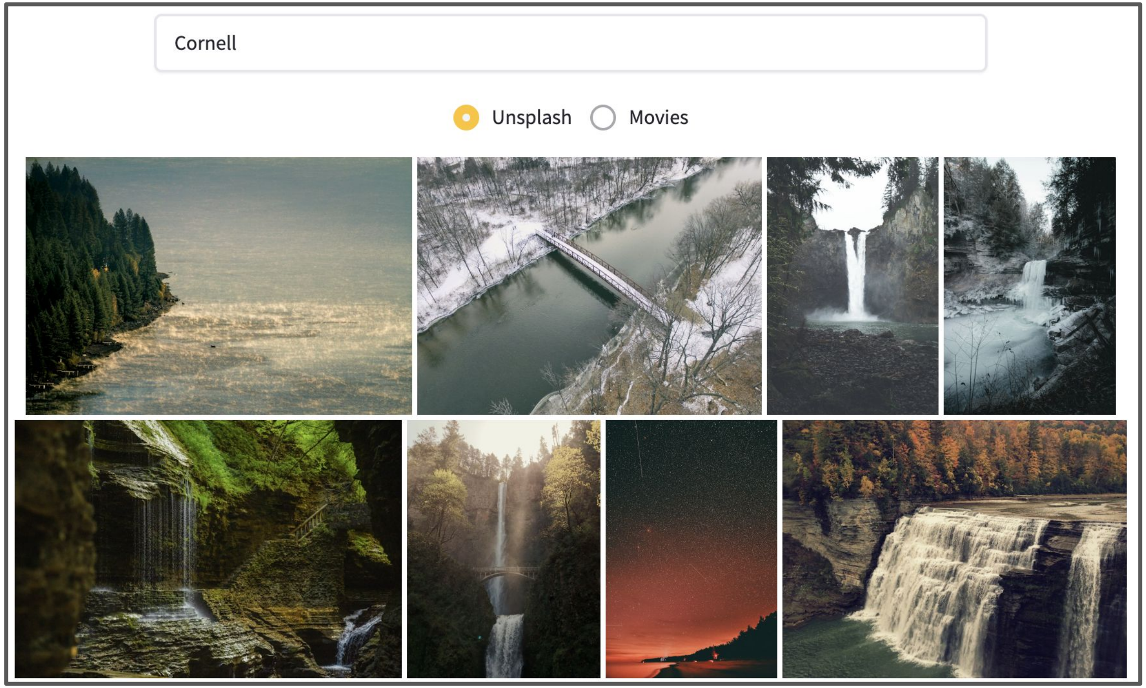
- Neural networks for data represented as graphs!



https://blogs.nvidia.com/blog/what-are-graph-neural-networks/

Modern Vision Networks

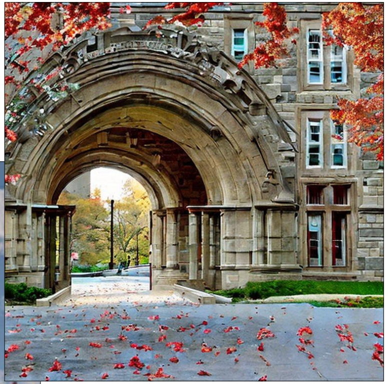
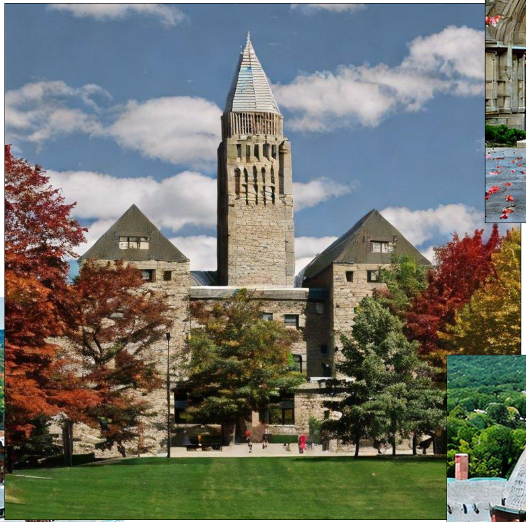
- Vision Transformers (ViTs)
- Vision Pre-Training
  - (Supervised, Self-supervised)
- Vision-Language Models



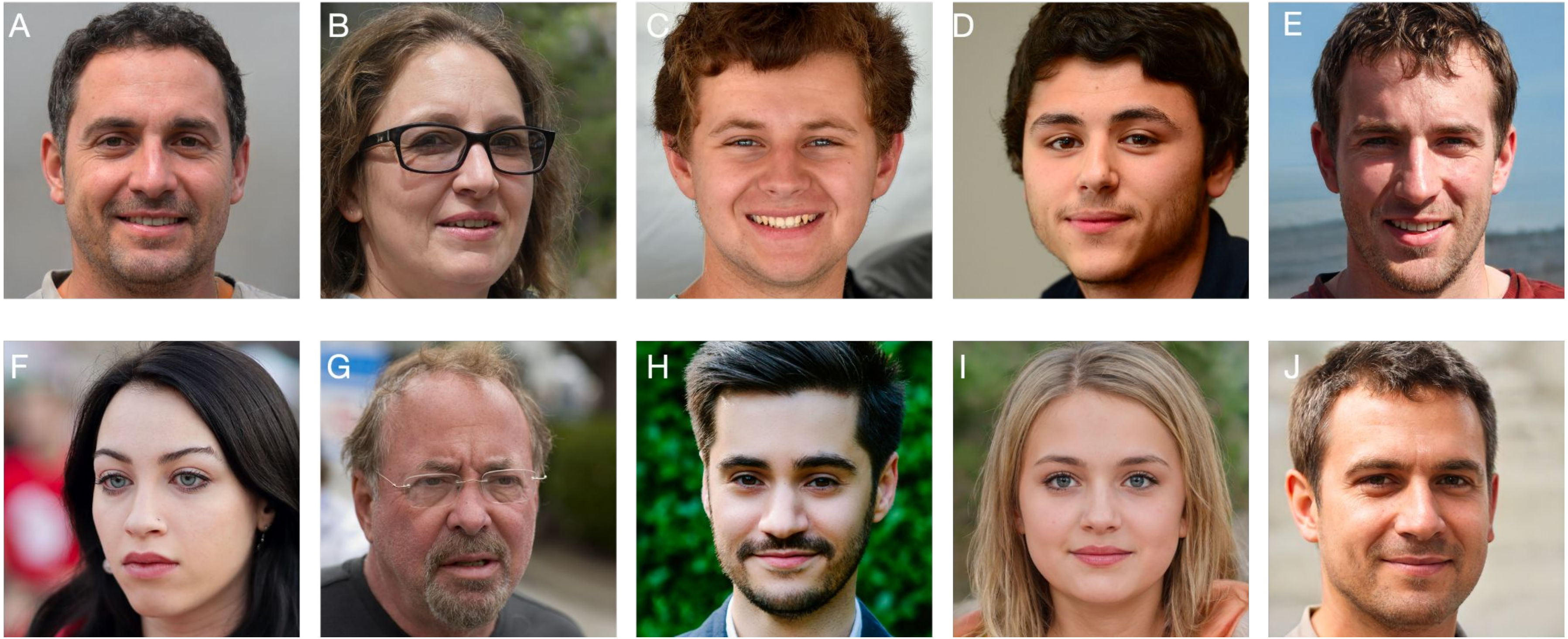
https://huggingface.co/spaces/vivien/clip

Generative Models

- U-Nets
- Variational Autoencoders (VAEs)
- Generative Adversarial Networks (GANs)
- Diffusion Models
- Multi-Modal Diffusion

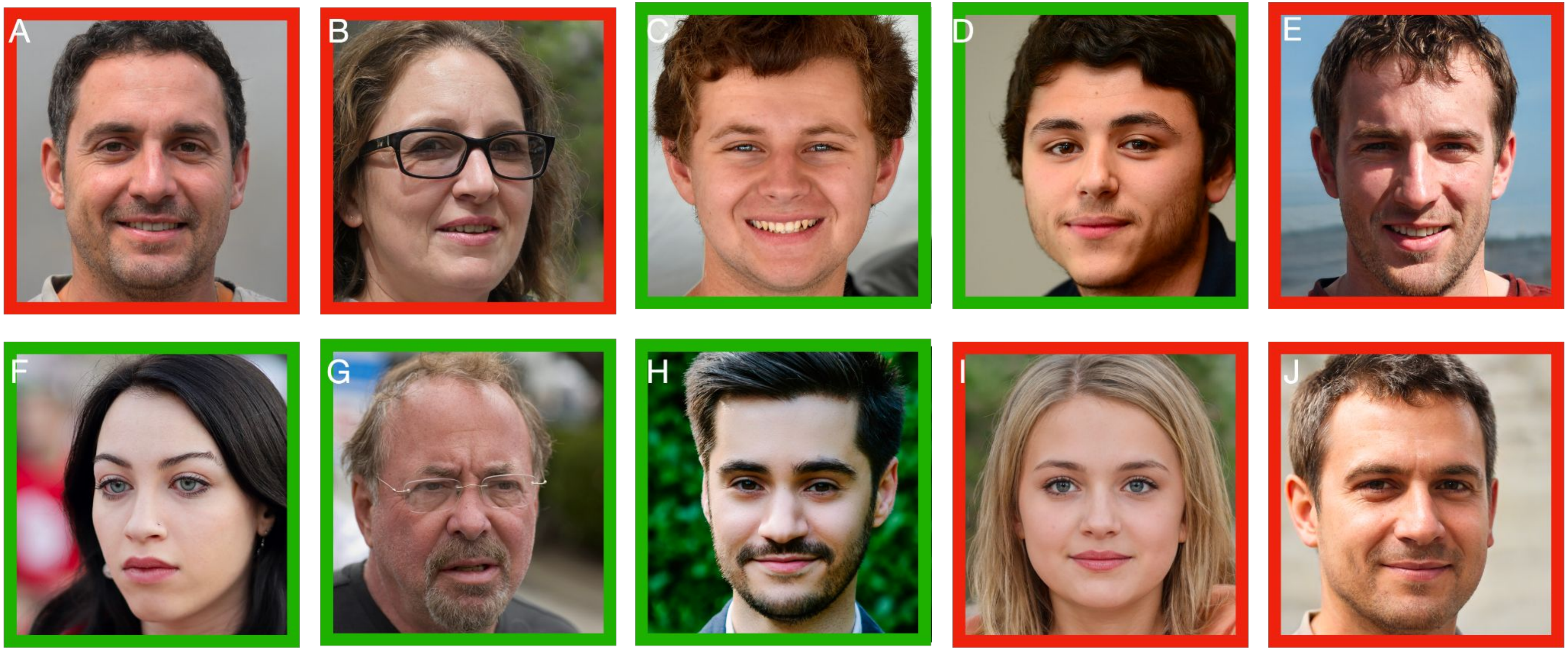



Real or Fake?



https://www.nytimes.com/interactive/2024/01/19/technology/artificial-intelligence-image-generators-faces-quiz.html

Real or Fake?

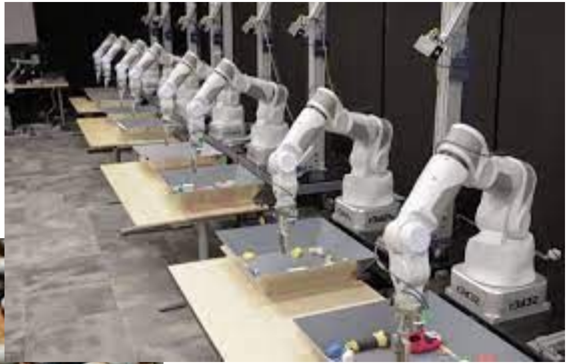


https://www.nytimes.com/interactive/2024/01/19/technology/artificial-intelligence-image-generators-faces-quiz.html

Reinforcement Learning

Technique for an agent to learn in an interactive environment by testing different actions and obtaining feedback from its experiences.

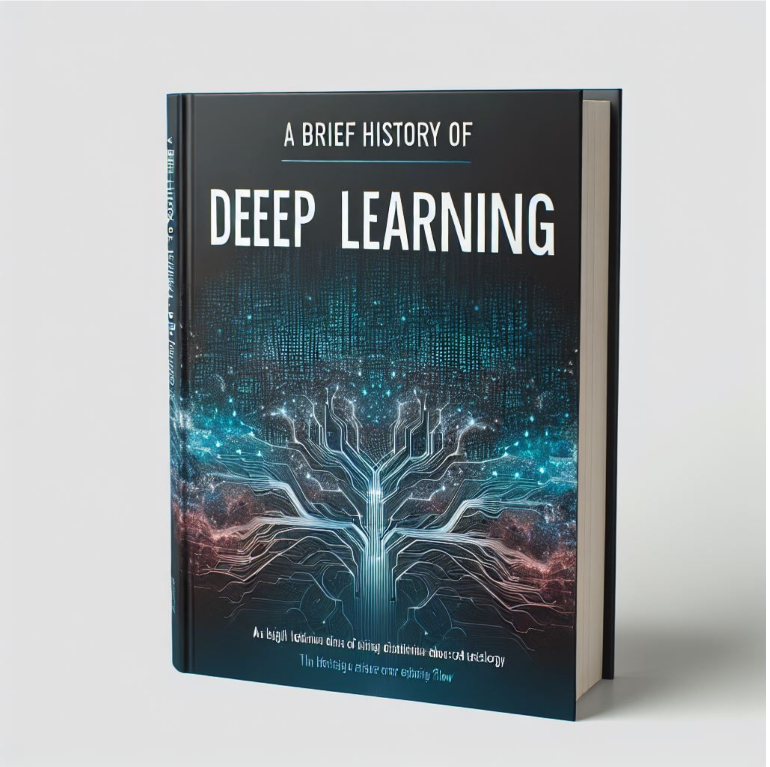
- Markov Decision Process
- Q-learning/Deep Q-learning
- Policy Gradients
- Exploration strategies
- RL from Human Feedback

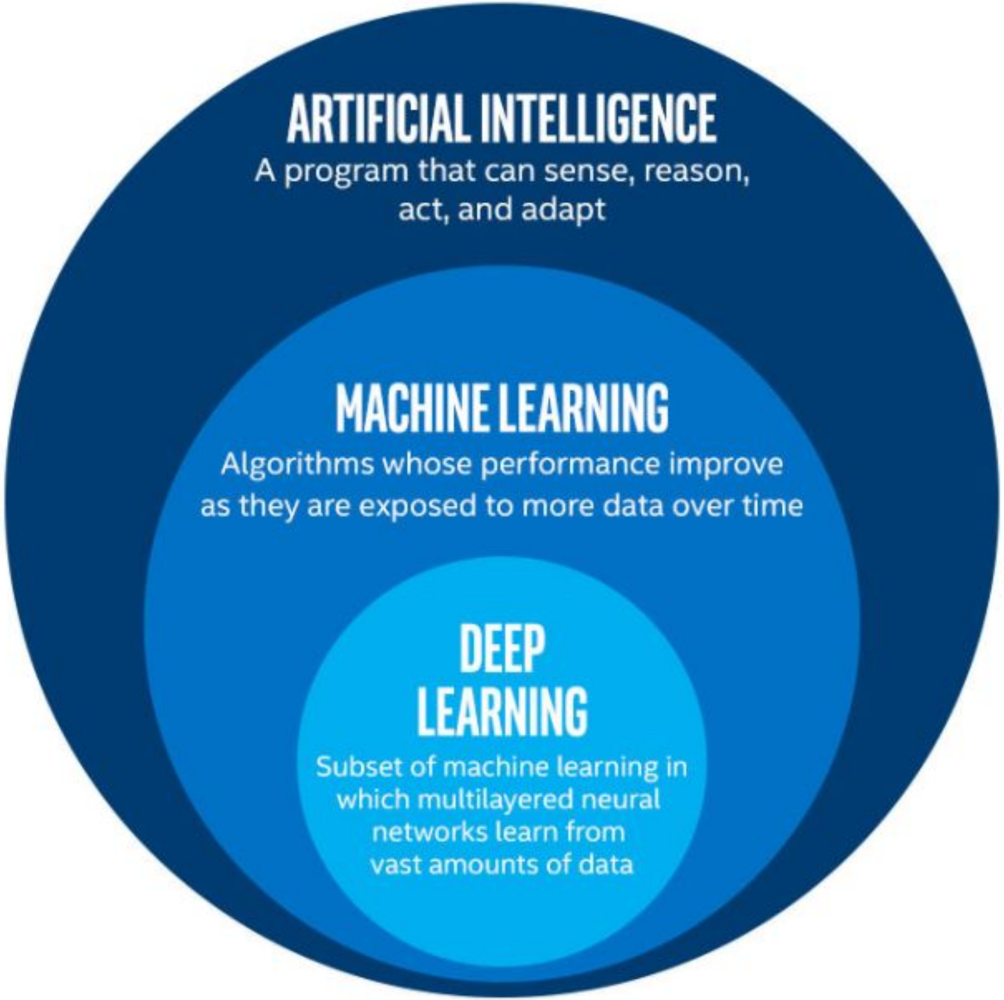


AI in Human Society

https://builtin.com/articles/ai-lawsuits-and-regulations

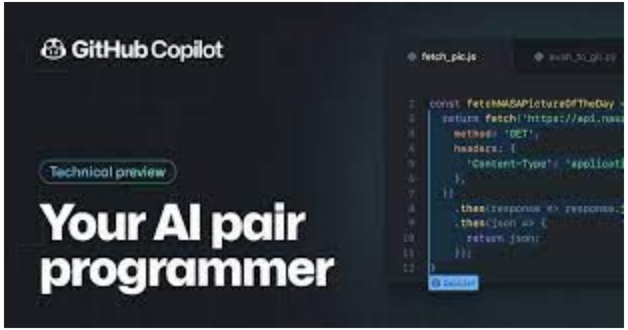
A brief history of Deep Learning




ARTIFICIAL INTELLIGENCE
A program that can sense, reason, act, and adapt
MACHINE LEARNING
Algorithms whose performance improve as they are exposed to more data over time
DEEP LEARNING
Subset of machine learning in which multilayered neural networks learn from vast amounts of data

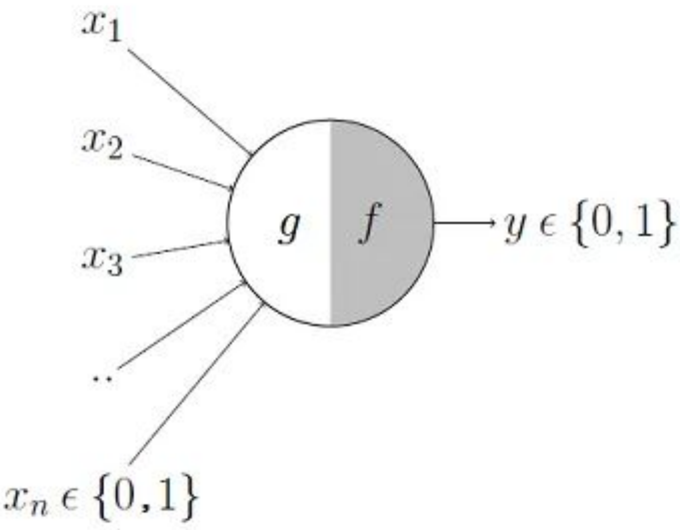
ChatGPT

DALL·E 2



McCulloch-Pitts Neuron

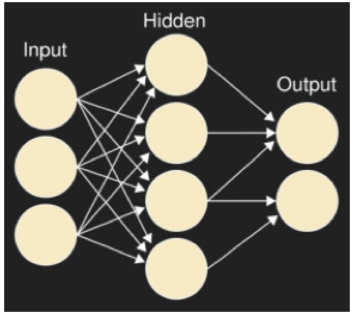
Computational model of a neuron that was proposed by Warren MuCulloch (neuroscientist) and Walter Pitts (logician) in 1943.

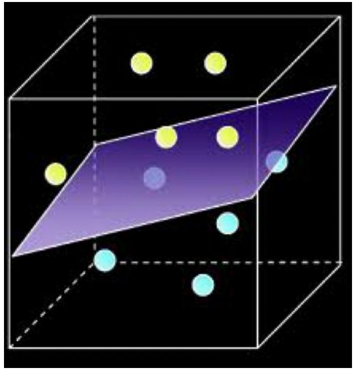


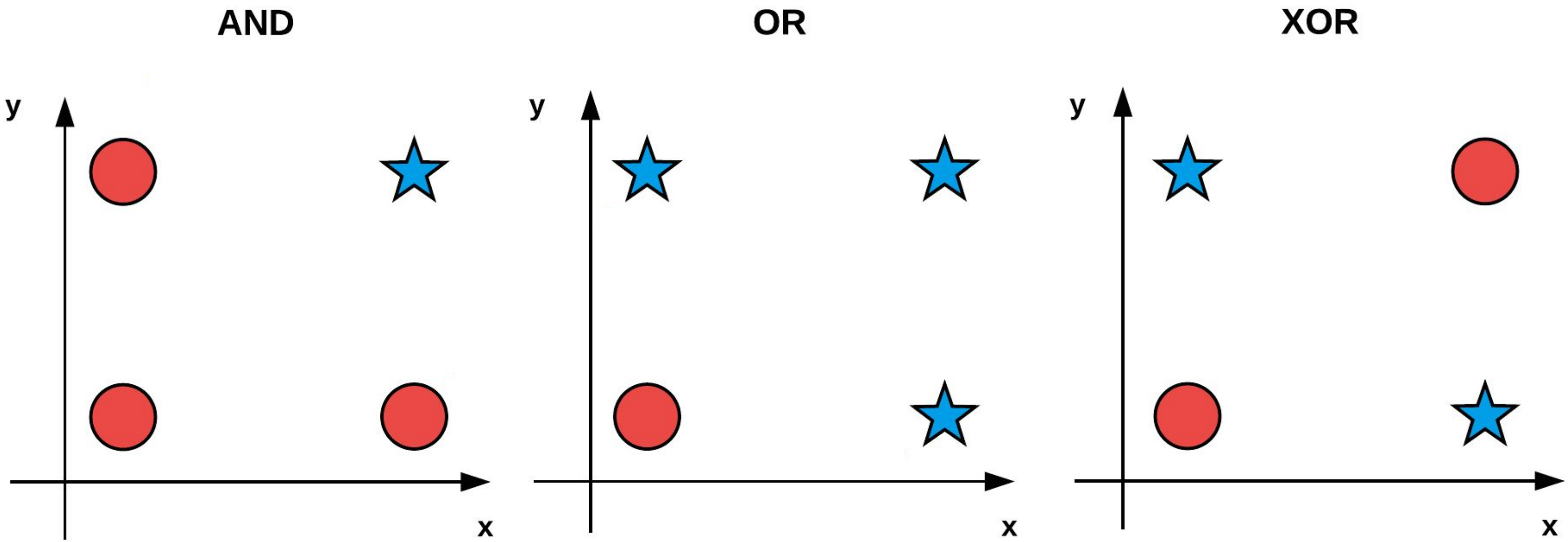
https://towardsdatascience.com/mcculloch-pitts-model-5fdf65ac5dd1

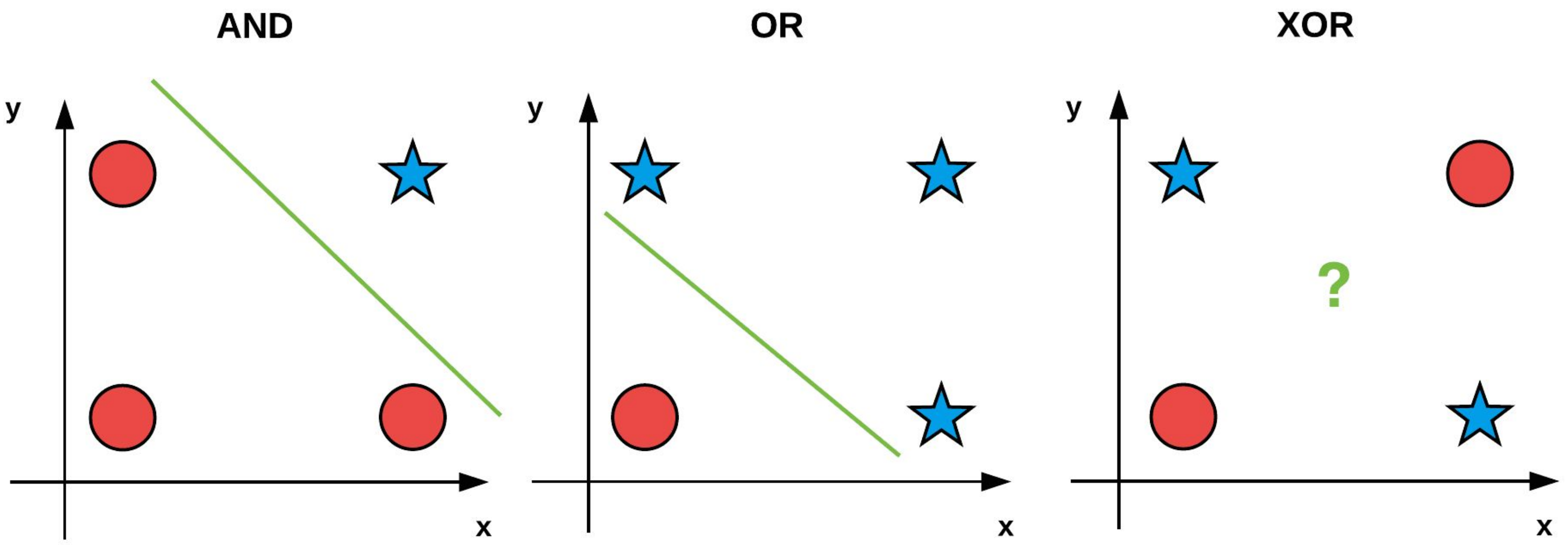
Perceptron (1957)

- Linear classifier, predecessor to Neural Network
- Trained with the perceptron update rule
- Invented @ Cornell University
  - First task: Recognize the Cornell “C” Logo



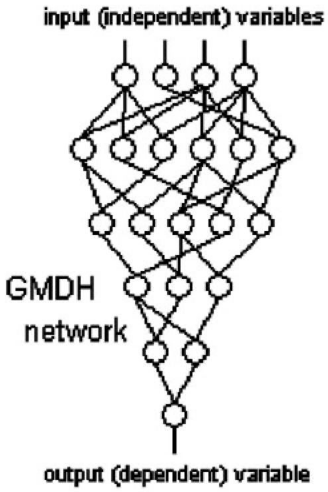



AND
y
x
OR
y
x
XOR
y
x


AND
y
x
OR
y
x
XOR
y
?
x

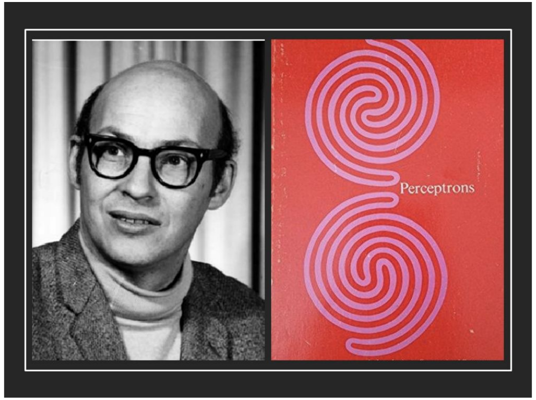
Multi-layer neural networks

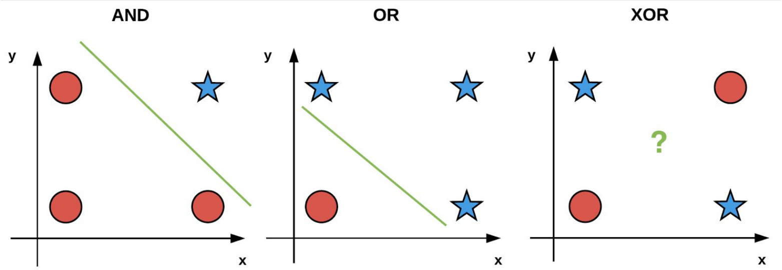
- Multi-Layer Perceptron, Rosenblatt (around 1965)
- Alexey Grigoryevich Ivakhnenko 1965 Group Method of Data Handling (GMDH)
    - 1971 Eight Layer Neural Nets with skip connections!



AI Winter (1974-1980)

- (1969) Minsky & Papert "killed" AI
- Burst huge expectation bubble
- Speech understanding / translation fails
- UK and US stop funding AI research

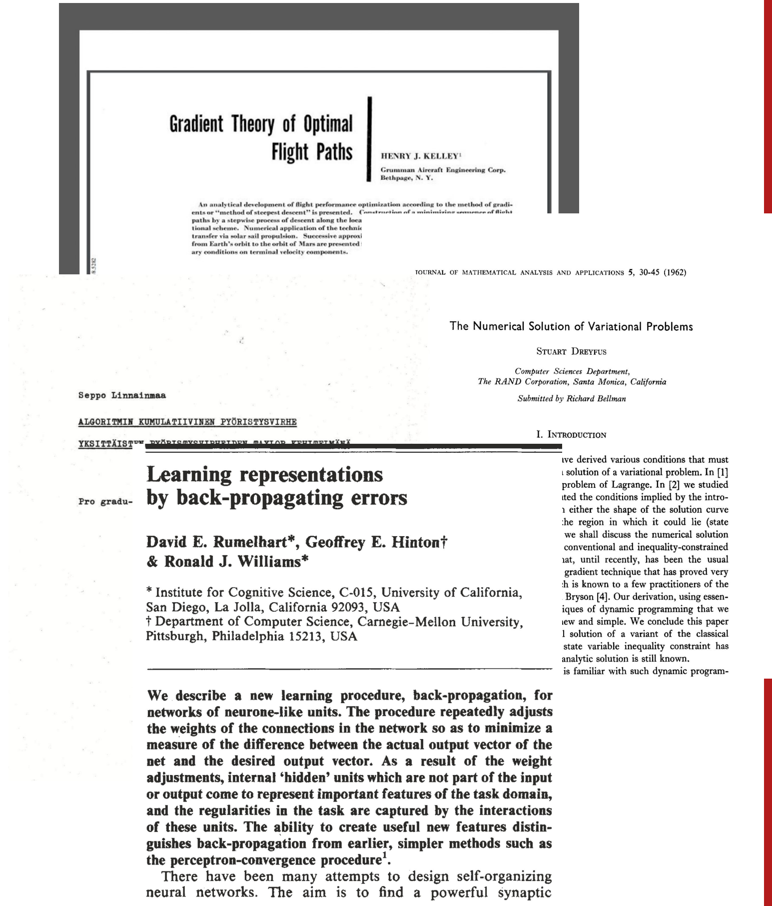




https://www.pyimagesearch.com/2021/05/06/implementing-the-perceptron-neural-network-with-python/

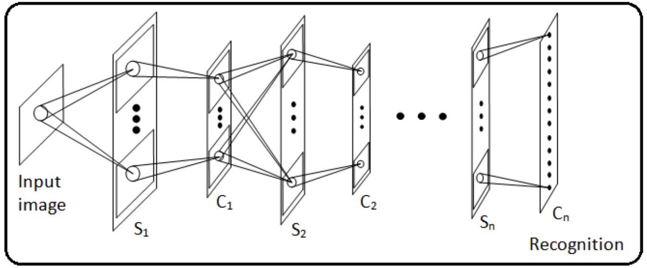
Backprop

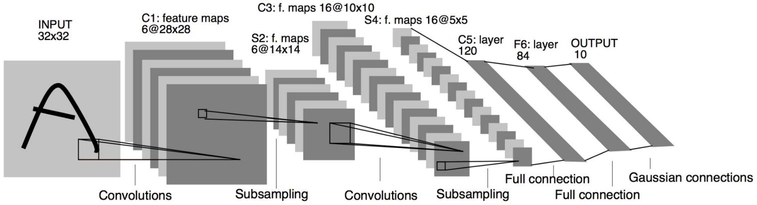
- 1960 Henry J. Kelly Initial formulation in control theory (rocket science)
- 1962 Stuart Dreyfuss (use of chain rule)
- 1979 Seppo Linnainman (modern backdrop with automatic differentiation [not in context of neural nets])
- 1982 Paul Werbos proposes backprop for artificial Neural Networks in PhD thesis
- 1986 Rumelhart, Hinton, Williams (coin the term “back-propagation”) make the algorithm popular (published in Nature)



ConvNets

- 1979 Kunihiko Fukushima invents Neocognitron
  - Heavily inspired by human Visual Cortex
  - Alternates between Simple Cells / Complex Cells
  - Unsupervised
- 1986 Yann LeCun introduces BackProp to ConvNets for Handwritten Digits (creates MNIST)





Recurrent Neural Nets

- 1982 John Hopfield “Hopfield Networks”
- 1991 Sepp Hochreiter formulates Vanishing Gradient Problem
- 1997 S. Hochreiter and Jürgen Schmidhuber publish “Long Short-Term Memory” (LSTM)
  - https://web.archive.org/web/20231216143334/https://people.idsia.ch/~juergen/ai-priority-disputes.html

Proc. Natl. Acad. Sci. USA
Vol. 79, pp. 2554–2558, April 1982
Biophysics

Neural networks and physical systems with emergent collective computational abilities

(associative memory/parallel processing/categorization/content-addressable memory/fail-soft devices)

J. J. Hopfield

Division of Chemistry and Biology, California Institute of Technology, Pasadena, California 91125; and Bell Laboratories, Murray Hill, New Jersey 07974

Contributed by John J. Hopfield, January 15, 1982

ABSTRACT Computational properties of use to biological organisms or to the construction of computers can emerge as collective properties of systems having a large number of simple equivalent components (or neurons). The physical meaning of content-addressable memory is described by an appropriate phase space flow of the state of a system. A model of such a system is given, based on aspects of neurobiology but readily adapted to integrated circuits. The collective properties of this model produce a content-addressable memory which correctly yields an entire memory from any subpart of sufficient size. The algorithm for the time evolution of the state of the system is based on asynchronous parallel processing. Additional emergent collective properties include some capacity for generalization, familiarity recognition, categorization, error correction, and time sequence retention. The collective properties are only weakly sensitive to details of the modeling or the failure of individual devices.

Given the dynamical electrochemical properties of neurons and their interconnections (synapses), we readily understand schemes that use a few neurons to obtain elementary useful biological behavior (1–3). Our understanding of such simple circuits in electronics allows us to plan larger and more complex circuits which are essential to large computers. Because evolution has no such plan, it becomes relevant to ask whether the ability of large collections of neurons to perform “computational” tasks may in part be a spontaneous collective consequence of having a large number of interacting simple neurons.

In physical systems made from a large number of simple elements, interactions among large numbers of elementary components yield collective phenomena such as the stable magnetic orientations and domains in a magnetic system or the vortex patterns in fluid flow. Do analogous collective phenomena in

calized content-addressable memory or categorizer using extensive asynchronous parallel processing.

The general content-addressable memory of a physical system

Suppose that an item stored in memory is “H. A. Kramers & G. H. Wannier *Phys. Rev.* **60,** 252 (1941).” A general content-addressable memory would be capable of retrieving this entire memory item on the basis of sufficient partial information. The input “& Wannier, (1941)” might suffice. An ideal memory could deal with errors and retrieve this reference even from the input “Vannier, (1941)”. In computers, only relatively simple forms of content-addressable memory have been made in hardware (10, 11). Sophisticated ideas like error correction in accessing information are usually introduced as software (10).

There are classes of physical systems whose spontaneous behavior can be used as a form of general (and error-correcting) content-addressable memory. Consider the time evolution of a physical system that can be described by a set of general coordinates. A point in state space then represents the instantaneous condition of the system. This state space may be either continuous or discrete (as in the case of N Ising spins).

The equations of motion of the system describe a flow in state space. Various classes of flow patterns are possible, but the systems of use for memory particularly include those that flow toward locally stable points from anywhere within regions around those points. A particle with frictional damping moving in a potential well with two minima exemplifies such a dynamics.

If the flow is not completely deterministic, the description is more complicated. In the two-well problems above, if the frictional force is characterized by a temperature, it must also produce a random driving force. The limit points become small

Cornell Bowers CIS

Universal Approximation

- 1989 George Cybenko proofs universal approximation of single hidden-layer neural networks
- Also yields wide-spread believe that more than one layer is unnecessary

Math. Control Signals Systems (1989) 2: 303–314

Mathematics of Control, Signals, and Systems

Approximation by Superpositions of a Sigmoidal Function*

G. Cybenko†

Abstract. In this paper we demonstrate that finite linear combinations of compositions of a fixed, univariate function and a set of affine functionals can uniformly approximate any continuous function of n real variables with support in the unit hypercube; only mild conditions are imposed on the univariate function. Our results settle an open question about representability in the class of single hidden layer neural networks. In particular, we show that arbitrary decision regions can be arbitrarily well approximated by continuous feedforward neural networks with only a single internal, hidden layer and any continuous sigmoidal nonlinearity. The paper discusses approximation properties of other possible types of nonlinearities that might be implemented by artificial neural networks.

Key words. Neural networks, Approximation, Completeness.

1. Introduction

A number of diverse application areas are concerned with the representation of general functions of an n-dimensional real variable, $x \in \mathbb{R}^n$, by finite linear combinations of the form

$$\sum_{j=1}^{N} \alpha_j \sigma(y_j^{\mathrm{T}} x + \theta_j), \tag{1}$$

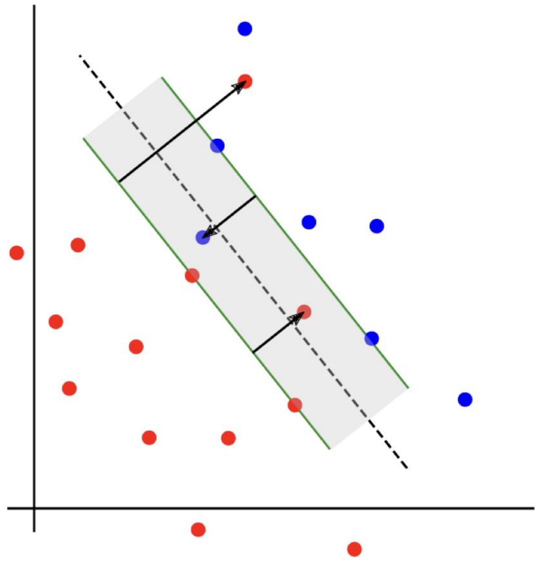
where $y_j \in \mathbb{R}^n$ and $\alpha_j, \theta \in \mathbb{R}$ are fixed. (y^{T} is the transpose of y so that $y^{\mathrm{T}}x$ is the inner product of y and x.) Here the univariate function σ depends heavily on the context of the application. Our major concern is with so-called sigmoidal σ's:

$$\sigma(t) \to \begin{cases} 1 & \text{as} \quad t \to +\infty, \\ 0 & \text{as} \quad t \to -\infty. \end{cases}$$

Such functions arise naturally in neural network theory as the activation function

Summer of SVMs 1995-2008

- 1993-1995 Corinna Cortes, Isabella Guyon, Vladimir Vapnik invent Support Vector Machines
- Mid 2000s ICML and NeuRIPS (NIPS) exclusively papers on non-neural network approaches
    - Mostly SVM, Graphical Models, Boosting
    - These algorithms are more efficient, easier to train / modify, have strong theoretical guarantees / frameworks



Neural Network Resurgence (2010s)

- Relentless effort by Hinton, Bengio, LeCun: Kept pushing Neural Nets when they were not cool - but did not join other communities (e.g. ICANN)
- Invent Deep Belief Nets in effort to attract experts in Graphical Models (mimics Graphical Models)
- Rename Neural Nets as “Deep Learning” (in effort to brand SVMs as “shallow”)
- Create ICLR as a venue to accept research on Neural Nets
- 2007 NeuRIPS Workshop on Deep Learning (rejected, changed to Hinton’s 60th birthday party)
- 2009 Fei-Fei Li creates ImageNet (after Caltech 4, 101, 256)
- 2012 Hinton’s deep network research creates AlexNet

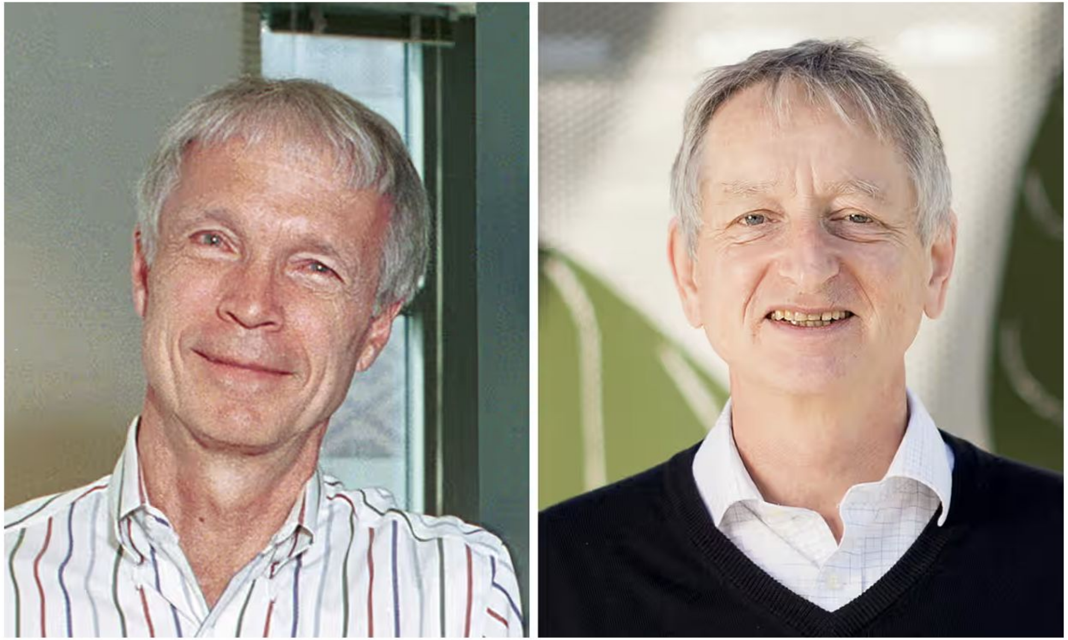
Turing Award 2018

Cornell Bowers CIS

NB 2024

Machine learning pioneers win Nobel prize in physics

Geoffrey Hinton, 'godfather of AI', and John Hopfield honoured for work on artificial neural networks



John Hopfield, left, and Geoffrey Hinton will share the 11m Swedish kronor (about £810,000) prize. Photograph: AP

Cornell Bowers CIS

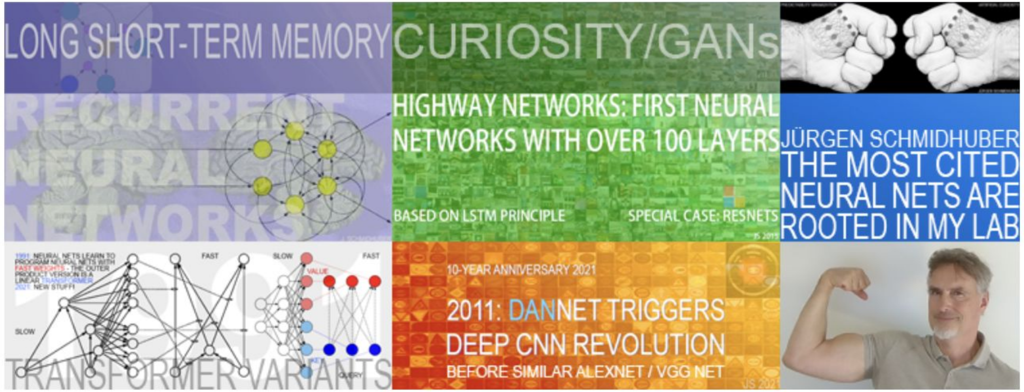
Controversy

Jürgen Schmidhuber
Pronounce: You_again Shmidhoobuh
Technical Report IDSIA-24-24, IDSIA

AI Blog
Twitter: @SchmidhuberAI
7 Dec 2024

A Nobel Prize for Plagiarism

Cornell Bowers CIS



Jürgen Schmidhuber (2021, updated 2024)
Pronounce: You_again Shmidhoobuh

AI Blog
Twitter: @SchmidhuberAI

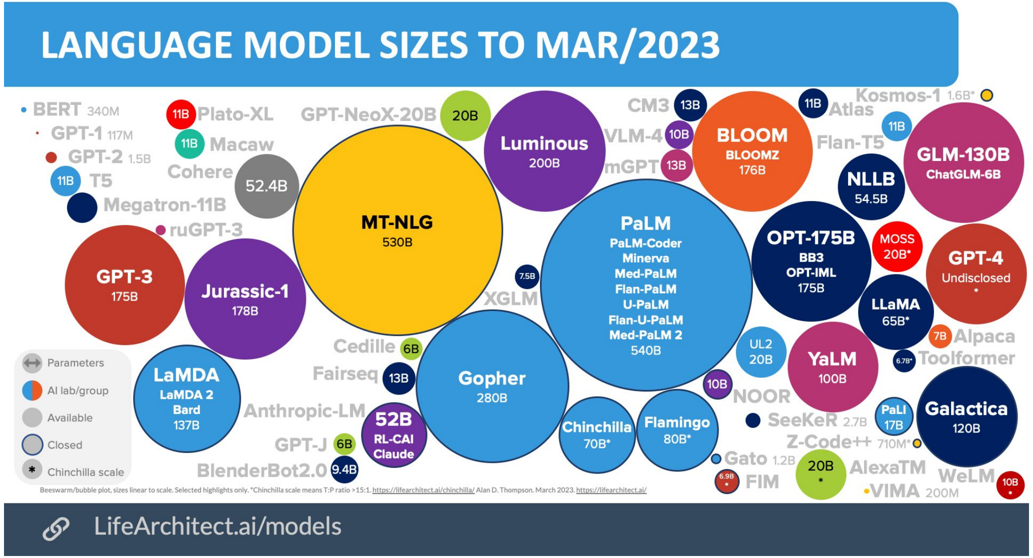
The most cited neural networks all build on work done in my labs

---

Abstract. Modern Artificial Intelligence is dominated by artificial neural networks (NNs) and deep learning.[DL1-4] Foundations of the most popular NNs originated in my labs at TU Munich and IDSIA. Here I discuss: **(1)** Long Short-Term Memory[LSTM0-17] (LSTM), the most cited NN of the 20th century, **(2)** ResNet, the most cited NN of the 21st century (which is an open-gated variant of our earlier Highway Net:[HW1-3] the first working really deep feedforward NN), **(3)** AlexNet and VGG Net, two of the most cited NNs of the 21st century (both building on our similar earlier DanNet:[GPUCNN1-9] the first deep convolutional NN[CNN1-4] to win image recognition competitions), **(4)** Generative Adversarial Networks[GAN0-1] (an instance of my earlier Adversarial Artificial Curiosity[AC90-20][DLH]), and **(5)** variants of Transformers (unnormalised linear Transformers are formally equivalent to my earlier Fast Weight Programmers).[TR1-6][FWP0-1,6][DLH] Most of this started with our Annus Mirabilis of 1990-1991[MIR] when compute was a million times more expensive than today. Back then we laid foundations of Generative AI, publishing principles of (4) GANs (1990, now used for deepfakes),[AC90-20][DLH] (5) Transformers (1991, the "T" in "ChatGPT" stands for "Transformer"),[TR1-6][FWP0-1,6][DLH] and (6) self-supervised pre-training for deep NNs (1991, the "P" in "GPT" stands for "pre-trained").[UN][UN0-3]

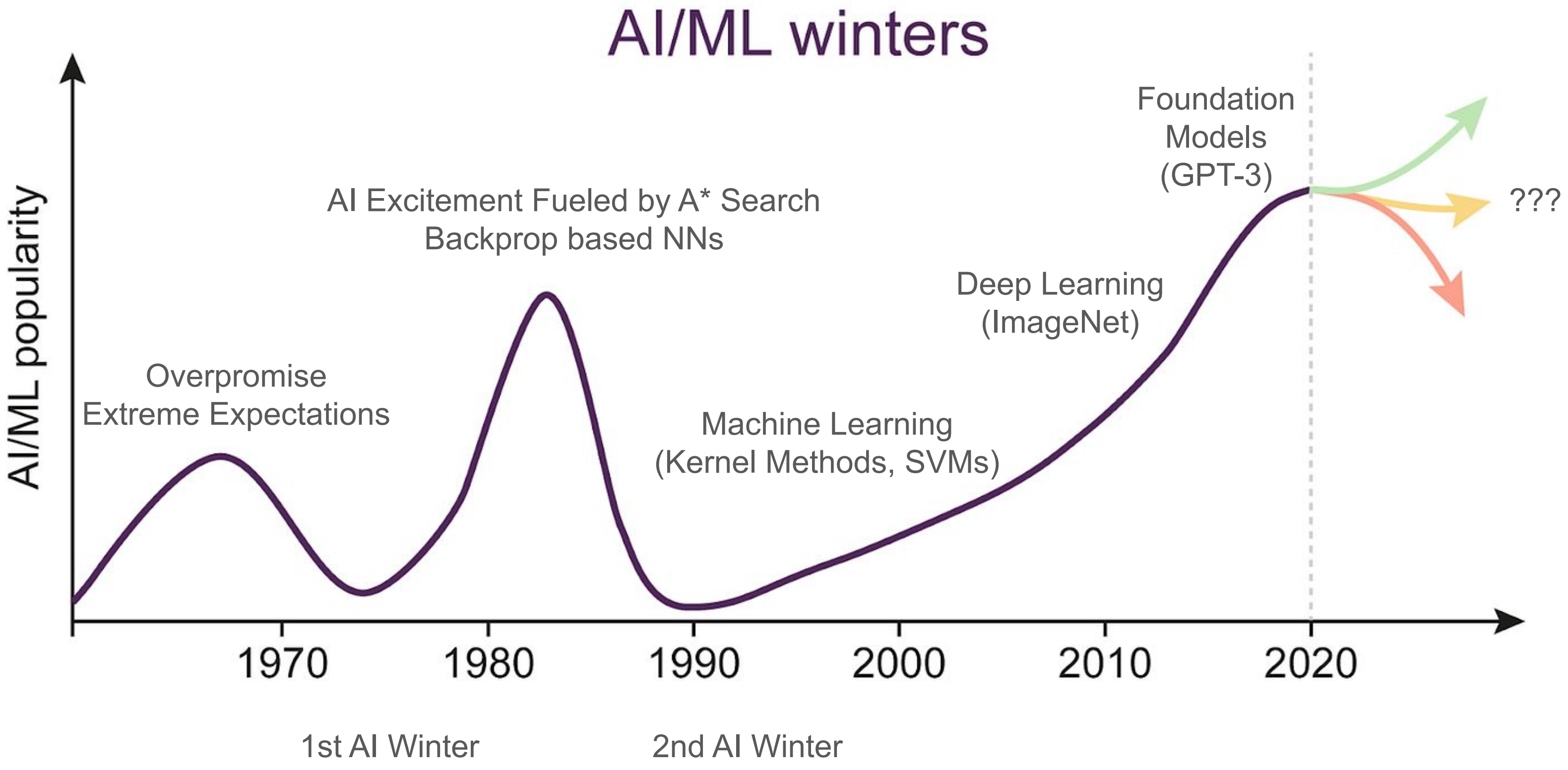
The Era of Scale (2020-Present)

- GPT-3 introduced in 2020
    - “Language Models are Few-Shot Learners”
- Stable Diffusion released in 2022



https://stability.ai/stable-image

Public Perception of AI/ML



https://medium.com/ersiliaio/beyond-the-hype-of-ai-ml-in-biomedicine-76c198c07467

Task: Predict whether an image contains an eye.

Thanks!

- If you have received a permission number
  - Enroll today if you’d like to take the course
- We will start sending out permission numbers to people on the waitlist later this week

- If you have not received a permission number and want to enroll
  - Come talk to us after class