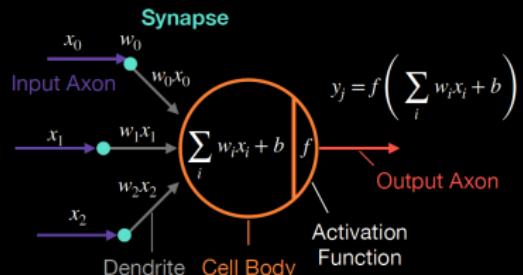
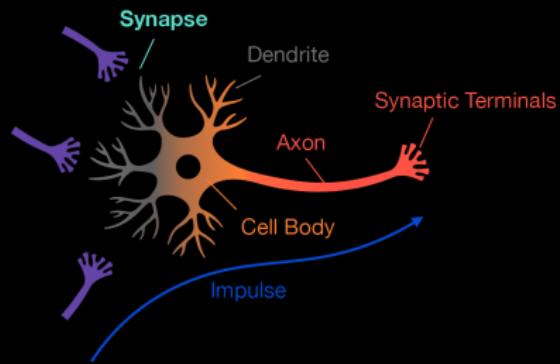


Welcome to Deep Neural Networks Course





► Instructors:

Course Responsible and Examiner

Name: Masoud Daneshthalab

Email: masoud.daneshthalab@mdu.se



Lecturer

Name: Seyedhamidreza Mousavi

Email: seyedhamidreza.mousavi@mdu.se



Lab Assistant

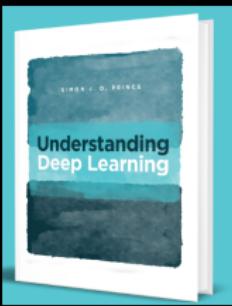
Name : Bostan Khan

Email: bostan.khan@mdu.se



- Homepage: <https://canvas.mdu.se/courses/18520>
- Questions: Post all questions on Canvas (Don't send email (except in an emergency))

- ▶ **Primary texts:** Understanding Deep Learning (Simon J.D. Prince 2023)



- ▶ Additional texts:
 - ▶ Machine Learning a Probabilistic Perspective (K.P. Murphy)
 - ▶ Deep Learning (Ian Goodfellow)

- ▶ **Proficiency in Python:** All lab assignments will be in Python (and use Numpy and Pytorch)
- ▶ **Calculus, Linear Algebra:** You should be comfortable taking derivatives and understanding matrix vector operations and notation.
- ▶ **Basic Probability and Statistics:** You should know the basics of probabilities, Gaussian distributions, mean, standard deviation, etc.

1. Introduction to Deep Learning
2. Fundamentals of Neural Networks and Deep Learning Frameworks and Tools
3. Convolutional Neural Networks (CNNs) deviation, etc.
4. Object Detection
5. Transformers
6. Large Language Models (LLMs)
7. Generative Adversarial Networks (GAN) and Diffusion models

Assignment and Grading

- We don't have any exam
- 5 Lab assignments, each 1.5 credit, grades Fail (U), 3, 4 or 5

```
1. import numpy as np
2.
3. def logistic_sigmoid_numpy(x):
4.
5.     Compute sigmoid of x.
6.
7.     Arguments:
8.         x : --- a scalar
9.         x : --- a vector or matrix
10.
11.    Returns:
12.        y : --- sigmoid(x)
13.
14.
15. # Add START CODE HERE *** (1 line of code)
16. x = ...
17. # Add END CODE HERE ***
18.
19. return y
```

Important: The input of all functions in the pytorch should be a tensor (use torch.tensor()) to convert the values and array to tensor.

```
1. import torch
2.
3. def logistic_sigmoid_torch(x):
4.
5.     Compute sigmoid of x in pytorch.
6.
7.     Arguments:
8.         x : --- a scalar
9.         x : --- a vector or matrix
10.
11.    Returns:
12.        y : --- sigmoid(x)
13.
14.
15. # Add START CODE HERE ***
16. x = torch.tensor(0)
17. # x is now a tensor
18. y = torch.sigmoid(x)
19. # Add END CODE HERE ***
20.
21. return y
```

- Your score < 0.9: U
- $0.9 \leq \text{your score} \leq 1.2$: 3
- $1.2 < \text{your score} \leq 1.4$: 4
- $1.4 < \text{your score}$: 5

- ▶ You must present all labs to the lab assistant.
- ▶ Each lab assignment will be submitted one week before the lab session.
- ▶ In each lab session, you can ask your questions or present a lab
- ▶ For presenting a lab all group members should be available and we ask some questions from each member for evaluation.
- ▶ There are three deadlines for presentation and upload the reports and python files:
 - ▶ During laboratory time
 - ▶ The first deadline
 - ▶ The second deadline

- ▶ When you finish each lab please present it as soon as possible in the next following lab sessions.
- ▶ **Do not wait until the last deadline.**
- ▶ You must upload the codes and reports on Canvas, Do not email anything.
- ▶ For the report, you should write a summary about the lab and also answer the questions in the lab assignment

1. You learn the LLM concept (ChatGPT) in this course. Don't use the GPT models for your implementations.
2. If someone copies, She/he will fail, all of them will fail.
3. It is not allowed to use codes from the internet or from previous year's students.
4. More information can be found at: <https://www.mdh.se/student/handbok-under-studietiden/fusk-storningar-och-trakassasserier>

Study Groups (2-3 people)

1. You must join a study group by the end of next week.
2. Find people on Canvas. You can join to a group in canvas (Don't send email)
3. This course will cover non-trivial material, learning in a group makes it easier and more fun!
4. Join to the lectures because we need to have a discussion to learn better, Also I will show some interesting demos just in the lectures

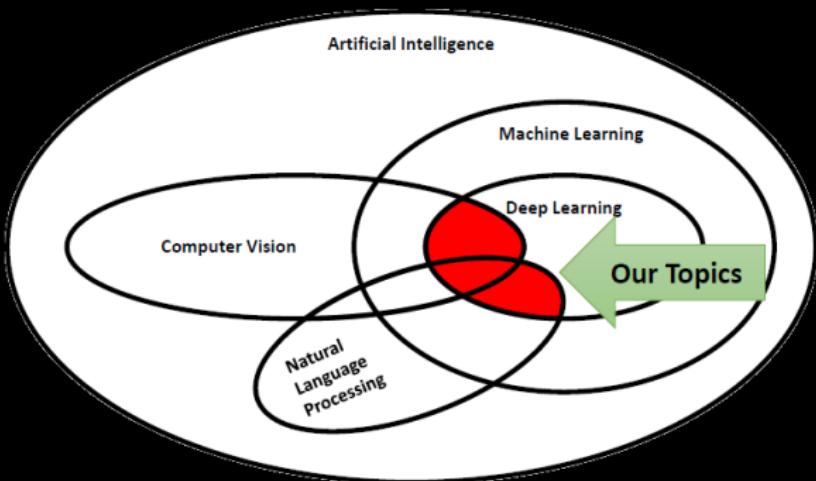


Take this course if ...

- ▶ you are interested in Machine Learning
- ▶ you are comfortable with a decent amount of mathematics
- ▶ you are not scared of programming

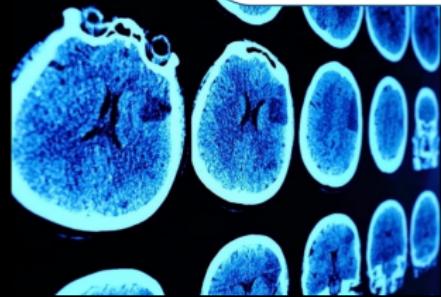
Don't take this course if ...

- ▶ matrices scare you
- ▶ you don't remember how to take derivatives
- ▶ you want an easy A



Machine Learning (ML)

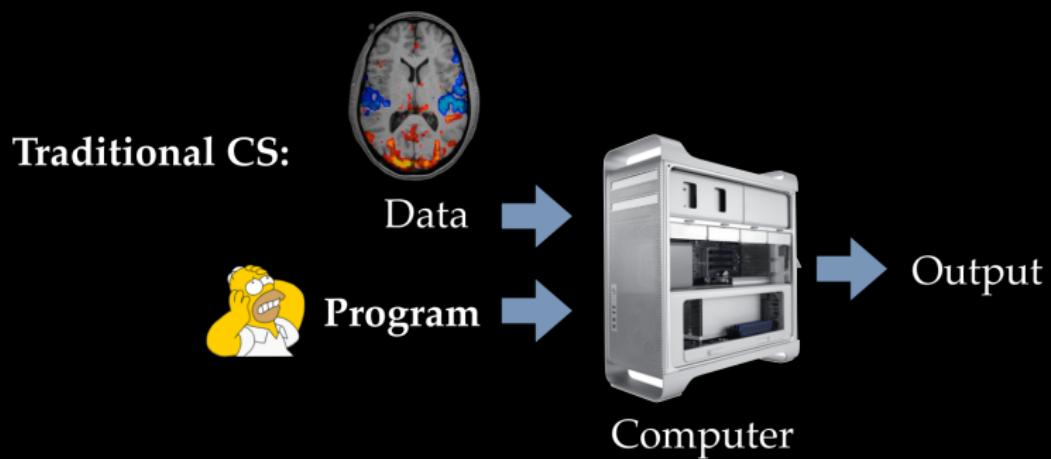
Programs that **improve** with **experience**.



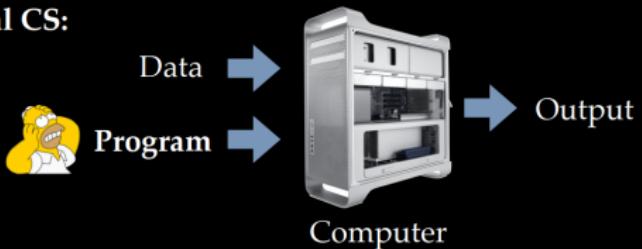
Siri



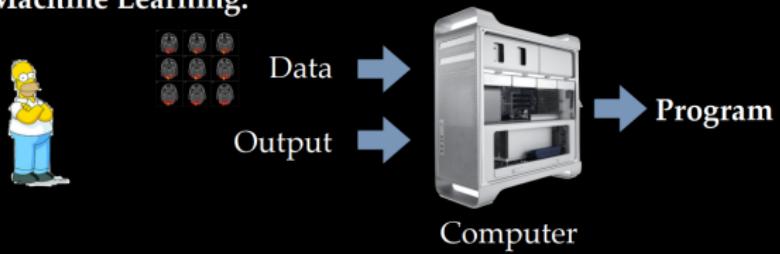
Traditional Computer Science

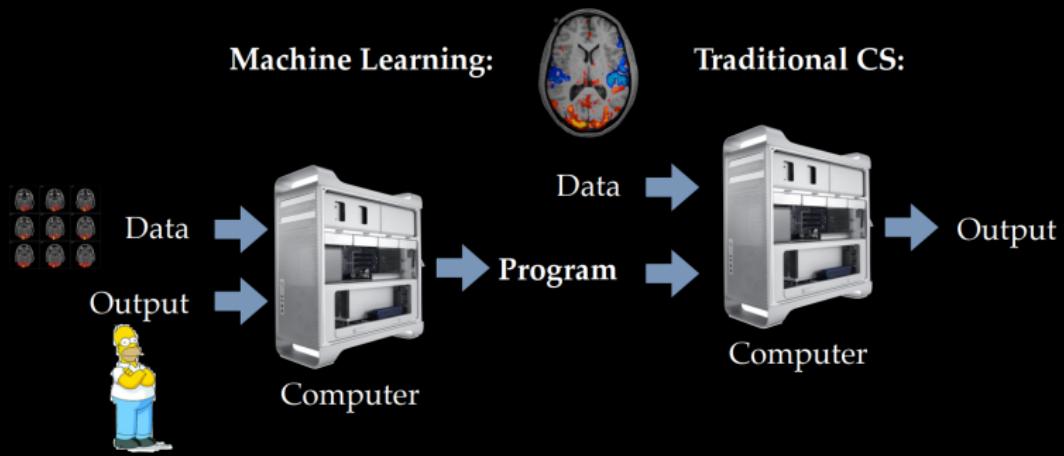


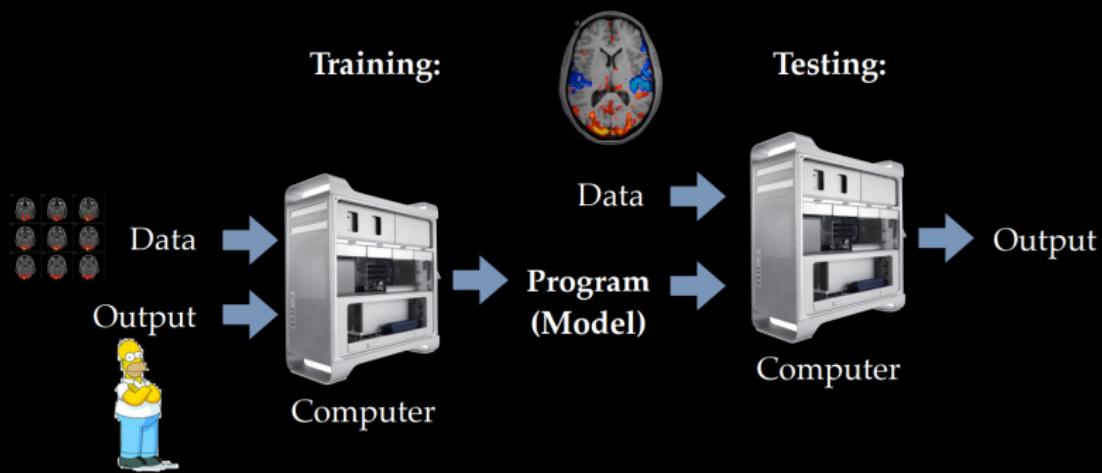
Traditional CS:



Machine Learning:







The Turing Test, 1950



Alan Turing

The Turing Test, 1950

A machine is intelligent if its answers are indistinguishable from a human's





Arthur Samuel

Checkers Program, 1952

Created a Checkers-playing program that got better overtime.



Also introduced the term
“Machine Learning.”

The term **Artificial Intelligence** originated in 1956 at a Workshop at Dartmouth

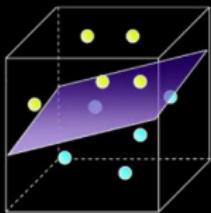
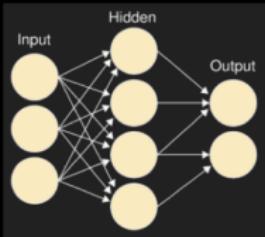


Frank Rosenblatt
@ Cornell!

Perceptron, 1957

Predecessor of deep networks.

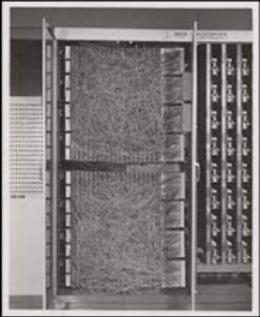
Separating two classes of objects
using a linear threshold classifier.



Perceptron, 1957



Frank Rosenblatt
@ Cornell!



NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Endings of Computer Designed in Head and Given Name

WASHINGTON, July 8 (UPI)—The Navy revealed the design of a new computer yesterday that it expects will be the first to learn by doing, to represent itself and to make mistakes.

The device, the Weather Bureau's Electronic Computer, can learn to identify faces, numbers, letters, shapes, objects and tell other computers what they mean, determine where the sequences, etc., come from, and even attempt to hold the field, or, if it makes a mistake, correct it.

The computer was designed and built at the Naval Research Laboratory by Frank Rosenblatt, a research psychologist at the Bureau's Psychological Research Division. He said Perceptrons might be used in space ships.

Without Human Control

The Navy's new computer, which is called a "perceptron," is designed to learn by doing. It is programmed to learn what is in the lists given on punch cards. It can learn to identify faces, numbers, letters, shapes, objects and tell other computers what they mean, determine where the sequences, etc., come from, and even attempt to hold the field, or, if it makes a mistake, correct it.

Mr. Rosenblatt said in preliminary tests he had a computer that could recognize faces, numbers, letters, shapes, objects and tell other computers what they mean.

"This was fed three cards, one card with a face, one card with a number and the other with squares and 'O' for the right."

Learning by Doing

In the latest model, the machine can learn to do more difficult tasks, such as reading like writing devices with 400 characters, writing like reading, identifying 14,000 words and translate sentences with the eyes.

UPI Photo Service

Perceptron, 1957

New Navy Device Learns by Doing

- The New York Times (July 8, 1958)

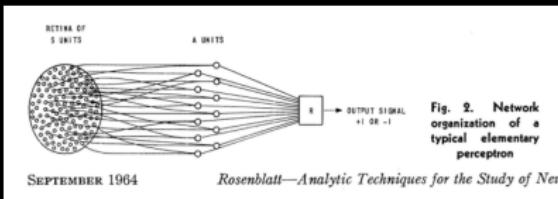
"Later perceptrons will be able to recognize people and call out their names and instantly translate speech in one language to speech or writing in another language, it was predicted."

Neural Network Boom, 1960-1979



Frank Rosenblatt
@ Cornell!

Neural Network Boom, 1960-1979



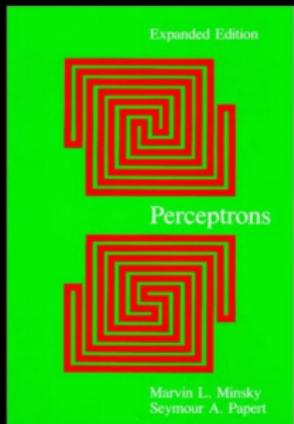
- 1962 Rosenblatt invents Multi-Layer Perceptron (MLP) (fixed hidden layer)
- 1965 Ivakhnenko and Lapa introduce first Feed Forward Neural Net (FFNN)
- 1967 First FFN trained with SGD [Amari 1967]
- 1970 Modern back-propagation is introduced by Seppo Linnainmaa

Who is the Father of Deep Learning?

<https://ieeexplore.ieee.org/document/9070967>

First AI Winter (1974-1980)

- (1969) Minsky & Papert “killed” AI
 - Perceptron cannot learn XOR function
- Burst huge expectation bubble
- Speech understanding / translation fails
- UK and US stop funding AI research
- Neural Networks and AI become “bad words”
- Rise of Rule based Systems



A.I. Boom - then Winter [1980-1993]

- AI BOOM: 1980-1987
 - Search Algorithms (No Learning)
 - Expert Systems
- AI winter: 1987-1993
 - Expectations too overhyped
 - Exponential algorithms misunderstood
 - A.I. Bubble bursts



Rebirth of A.I. as Machine Learning

- Machine Learning:
 - Originally: Mostly a name game to get funding.
- Differences in approach:
 - ML: Bottom up, AI: Top down
 - ML: More practical smaller goals
 - Based on **Statistics and Optimization, not Logic and Search**



TD-Gammon (1994)

- Gerry Tesauro (IBM) teaches a neural network to play Backgammon. The net plays 100K+ games **against itself** and beats world champion [Neurocomputation 1994]
- Algorithm teaches **itself** how to play so well!!!
- Algorithm found new techniques that people had erroneously ruled out.



Deep Blue (1997)

- IBM's Deep Blue wins against Kasparov in chess. Crucial winning move is made due to Machine Learning (G. Tesauro).
- (Mostly a more classical AI system)



Expanding the reach, 2000s

Learning to rank

- Powering search engines: Yahoo, Google, Bing, ...

Topic Modeling:

- Detecting and organizing documents by subject matter.
- Making sense of the unstructured data on the web.

Online economy:

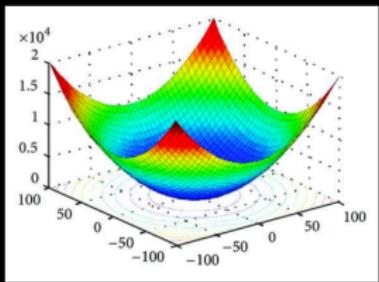
- Ad placement and pricing.
- Product recommendation.

**Machine learning became profitable!
Euphoria! Excitement!**



2000-2010 The Convex Years

- Success of **Support Vector Machines** (SVM) and **Convex** Optimization
- **Kernelization** enables non-linear classifiers that are still convex
- Allows **rigorous proofs** about convergence, learning guarantees
- Finally **solid theory** and **empirical success**
- (Non-convex algorithms, e.g. Neural Networks fell out of favor and were frowned upon.)



Return of Neural Networks, 2010s

Neural networks (aka Multi-Layer Perceptrons) return and excel at image recognition, speech recognition, ...

The 2018 Turing award was given to Yoshua Bengio, Geoff Hinton, and Yann LeCun.



2016 Alpha Go

- 1920 CPUs and 280 GPUs
- Deep Mind's Alpha Go wins against Lee Sedol 5:1
- Big shock
 - China invests heavily in AI research
- Beginning of “AI arms race”



Deep Learning is everywhere



Vision



Language



Multimodal

Deep Learning for Image Classification

DNNs achieve super-human classification accuracy on ImageNet

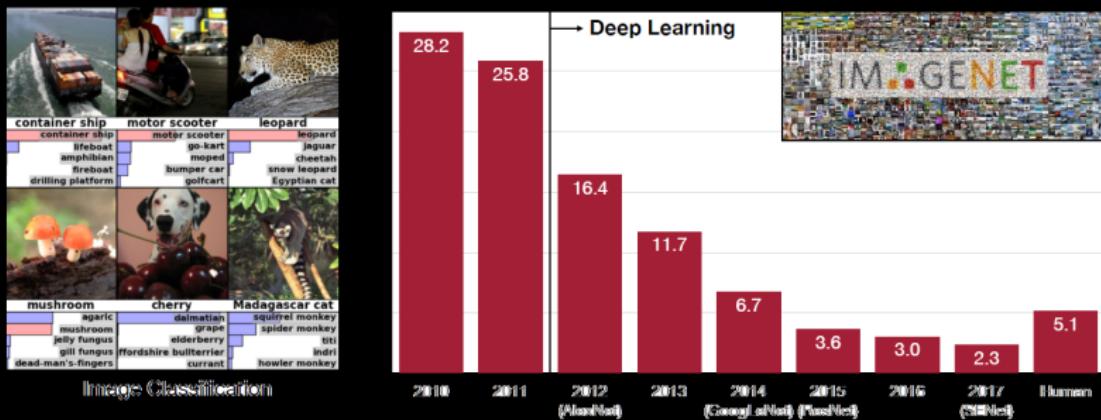
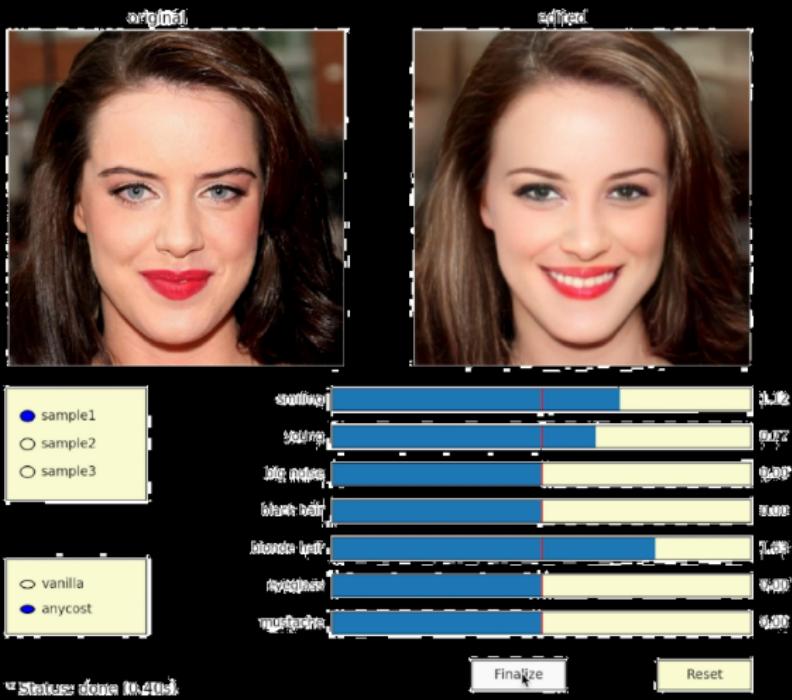
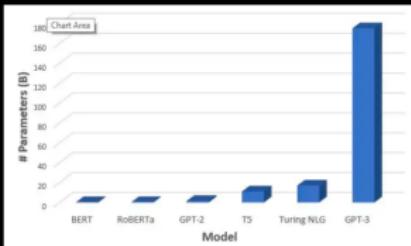


Image Generation



2021 Generative Models become Mainstream

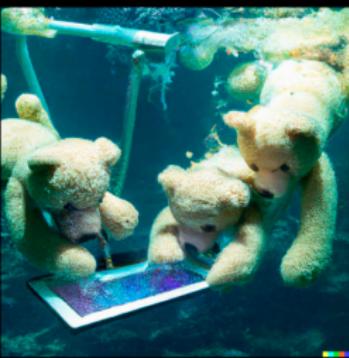
- 178 Billion Parameters
- Cost: \$4.6 Million (est.)
- 355 GPU years of training (est.)
- Trained on 45 TB of text



DALLE Text2Images - ML is creative



Teddy bears
mixing sparkling chemicals as mad scientists
in a steampunk style



Teddy bears
working on new AI research
underwater with 1990s technology



An astronaut
playing basketball with cats in space
in a watercolor style

<https://openai.com/dall-e-2/>

Diffusion models create realistic videos from a natural language description



<https://openai.com/index/sora/>

Deep Learning is everywhere



Vision



Language



Multimodal

Please draft a short description of Deep Learning and neural networks course



Course Title: Deep Learning and Neural Networks

Course Description: This course provides a comprehensive introduction to deep learning, a subset of machine learning focused on neural networks. Students will explore the fundamental concepts behind neural networks, including the architecture of layers, activation functions, backpropagation, and optimization techniques. Through hands-on projects and real-world applications, learners will build, train, and fine-tune deep neural networks for tasks such as image classification, natural language processing, and time series analysis. By the end of the course, participants will gain practical experience in implementing state-of-the-art models using popular deep learning frameworks such as TensorFlow and PyTorch.

GitHub CoPilot can make meaningful coding suggestions based on context

Please draft a short description of Deep Learning and neural networks course



Course Title: Deep Learning and Neural Networks

Course Description: This course provides a comprehensive introduction to deep learning, a subset of machine learning focused on neural networks. Students will explore the fundamental concepts behind neural networks, including the architecture of layers, activation functions, backpropagation, and optimization techniques. Through hands-on projects and real-world applications, learners will build, train, and fine-tune deep neural networks for tasks such as image classification, natural language processing, and time series analysis. By the end of the course, participants will gain practical experience in implementing state-of-the-art models using popular deep learning frameworks such as TensorFlow and PyTorch.

Deep Learning is everywhere



Vision



Language



Multimodal

Vision-Language Models



User



User

Do you know who drew this painting?



LLaVA

The painting depicts a woman, commonly believed to be Mona Lisa, the famous artwork by Leonardo da Vinci. It is a portrait painting that showcases the woman's enigmatic smile and has become one of the most famous and iconic art pieces in the world. The original work is displayed in the Louvre Museum in Paris, and it is known for its intricate details, use of oil paint, and the artist's innovative techniques that contributed to its enduring appeal and mystery.

Applications

<https://lovable.dev/>

<https://gamma.app/>

<https://notebooklm.google.com/>

<https://www.buildai.space/app/trip-planner-ai>

<https://krisp.ai/>

Expectations: Attendance

Please understand that consistent presence and engagement are highly valued in this course.

