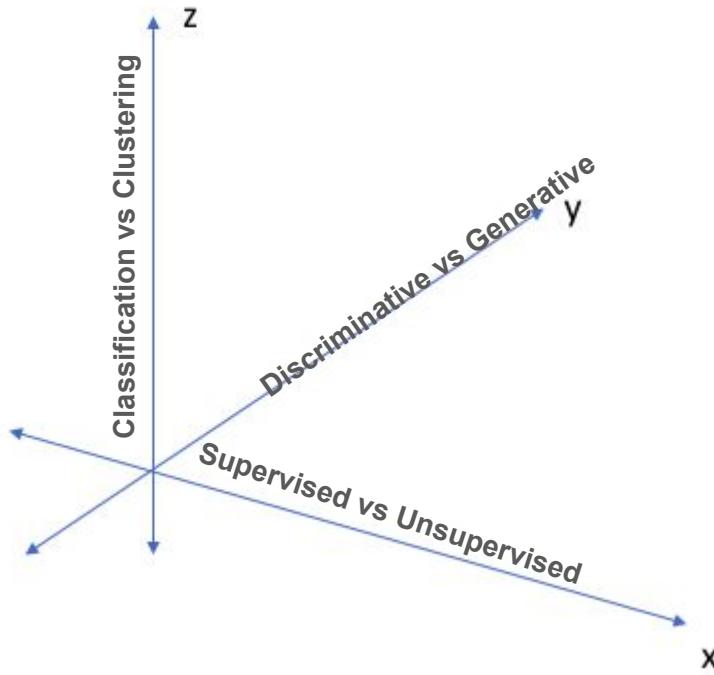


# Generative AI: History

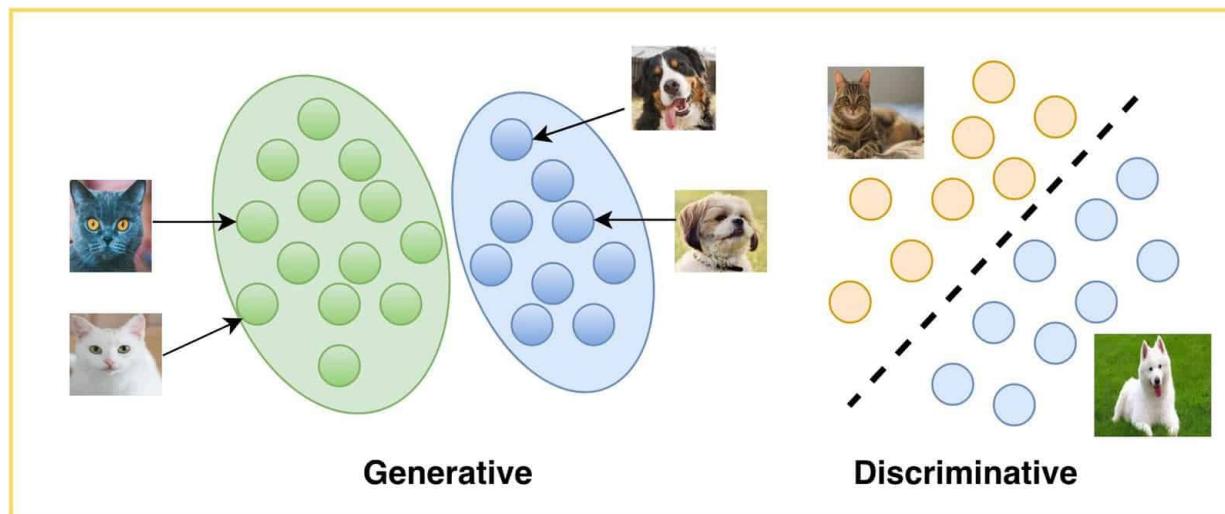
Arin Ghazarian  
Chapman University

# ML Models Space



# Generative vs Discriminative Models

- Discriminative Modelling: learn the boundaries between classes or labels in a dataset (decision boundary). Modeling  $P(Y|X)$
- Generative Modelling: learn to generate new data samples similar to the training data. They try to learn the underlying distribution of the data. Modeling  $P(X,Y)$



# Discriminative Models Limitation

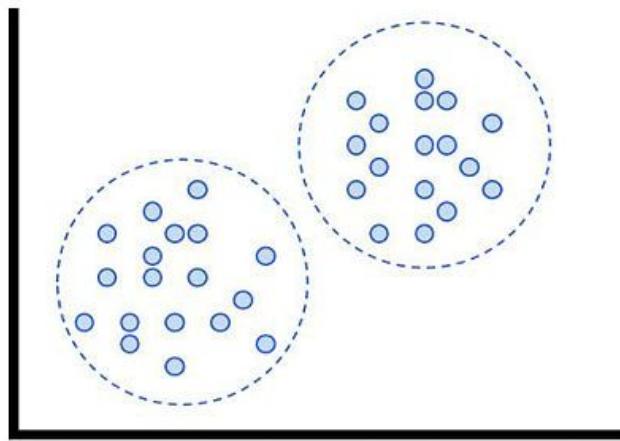
- Can't model  $P(X)$ : the probability of seeing a certain  $X$
- Thus, we can't sample from  $P(X)$  and generate samples

## $P( X , Y )$ vs $P ( Y | X = x )$

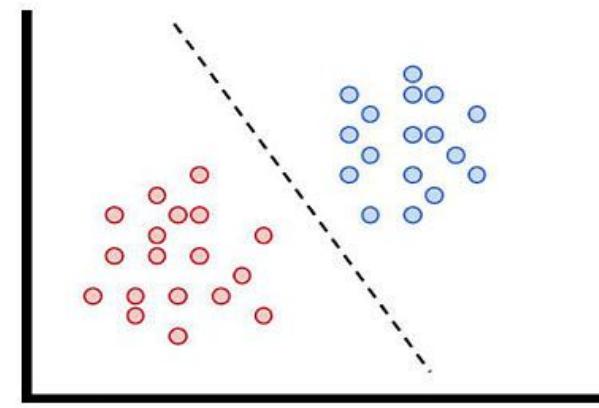
- A generative model is a statistical model of the joint probability distribution  $P( X , Y )$  on a given observable variable X and target variable Y
- A discriminative model is a model of the conditional probability  $P ( Y | X = x )$  of the target Y, given an observation x. It can be used to "discriminate" the value of the target variable Y, given an observation x.

# Supervised vs Unsupervised

- Supervised: Identification of a pattern as a member of a known category
- Unsupervised: Classification (known categories) vs Clustering (learning categories)



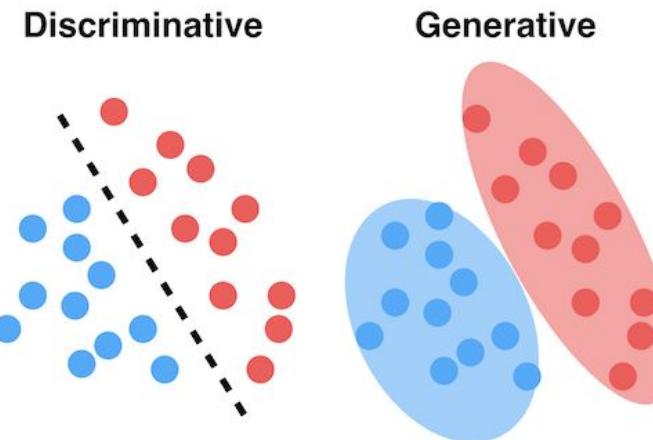
**Clustering**



**Classification**

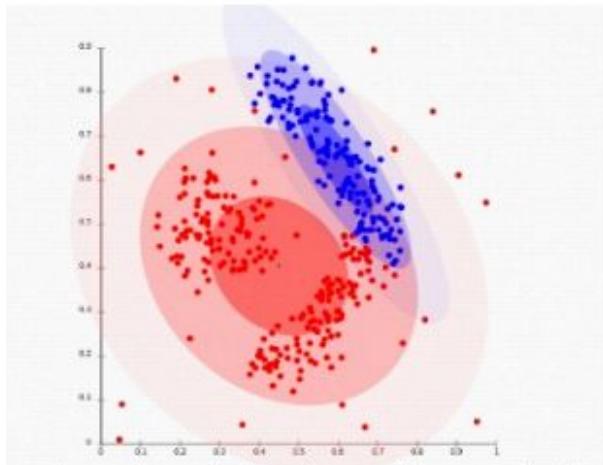
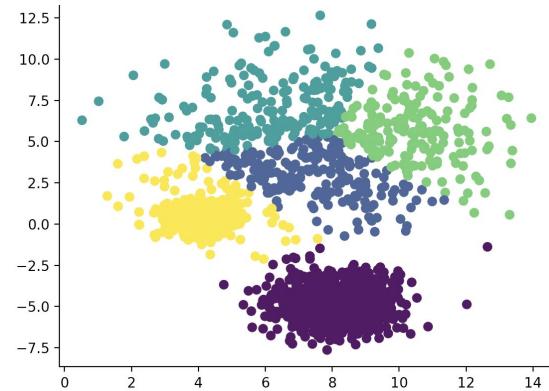
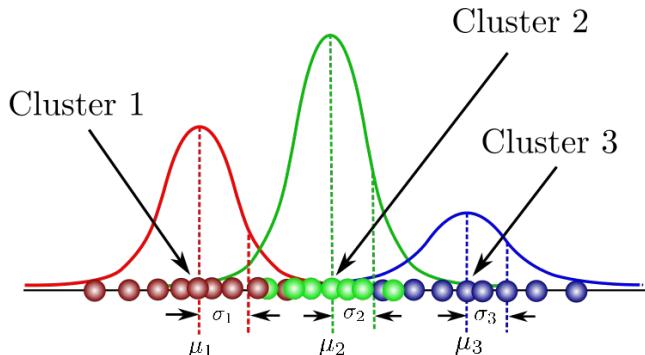
# Discriminative vs Generative Classification

- Discriminative classification
  - Finds a decision boundary
  - Uses conditional probability  $P(Y|X)$
- Generative classification
  - Finds the distribution of a dataset  $P(X,Y)$
  - Selects  $y_k$  which has the largest  $P(X,Y=y_k)$



# Discriminative vs Generative Clustering

- Discriminative model
  - No Predictive ability for new instances
  - Assigns labels to inputs
  - Hard clustering
- Generative models
  - Predictive ability for new instances
  - Assigns probabilities to inputs
  - Soft clustering



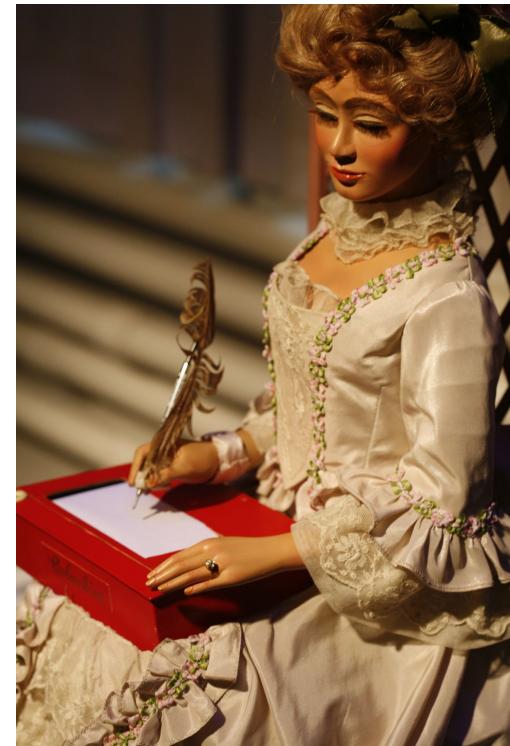
# Ancient Greeks

- Automata in ancient Greek
- Daedalus and Hero of Alexandria designed machines capable of writing text, generating sounds, and playing music.



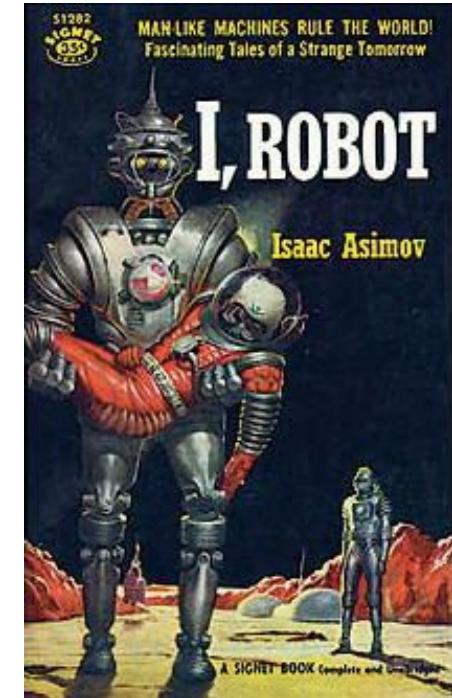
The book About automata by Hero of Alexandria (1589 edition)

# Automaton in the Swiss Museum CIMA (18th century)



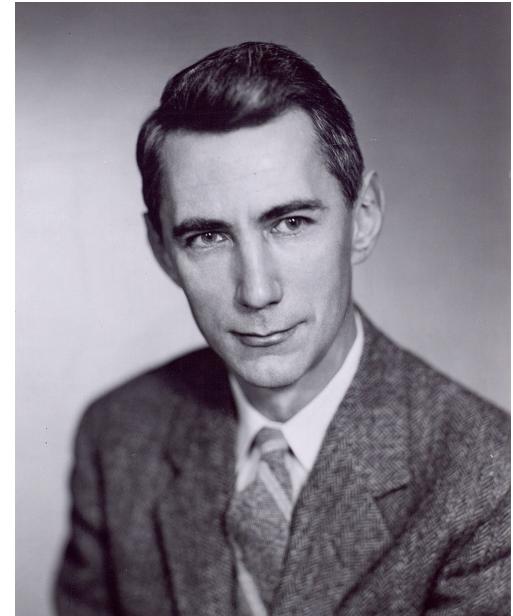
# Asimov's Three Laws of Robotics: 1940s

1. First Law: A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. Second Law A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
3. Third Law A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.
4. Zeroth Law A robot may not harm humanity, or, by inaction, allow humanity to come to harm.



# Shannon Text generation: 1948

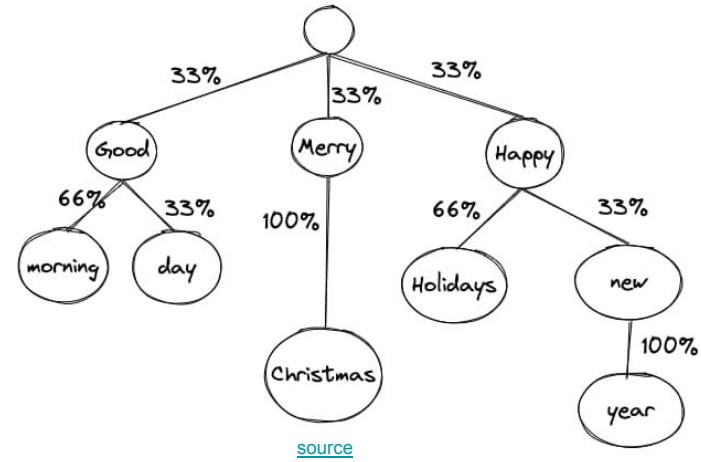
- Claude Shannon publishes his paper “[A Mathematical Theory of Communications](#)”
  - Mentions the idea of n-grams
  - Tries to answer the question: Given a sequence of letters (a sentence, for example), what's the likelihood of the next letter?
  - Founded the field of information theory
  - He proposed using a Markov chain to create a statistical model of the sequences of letters in a piece of English text



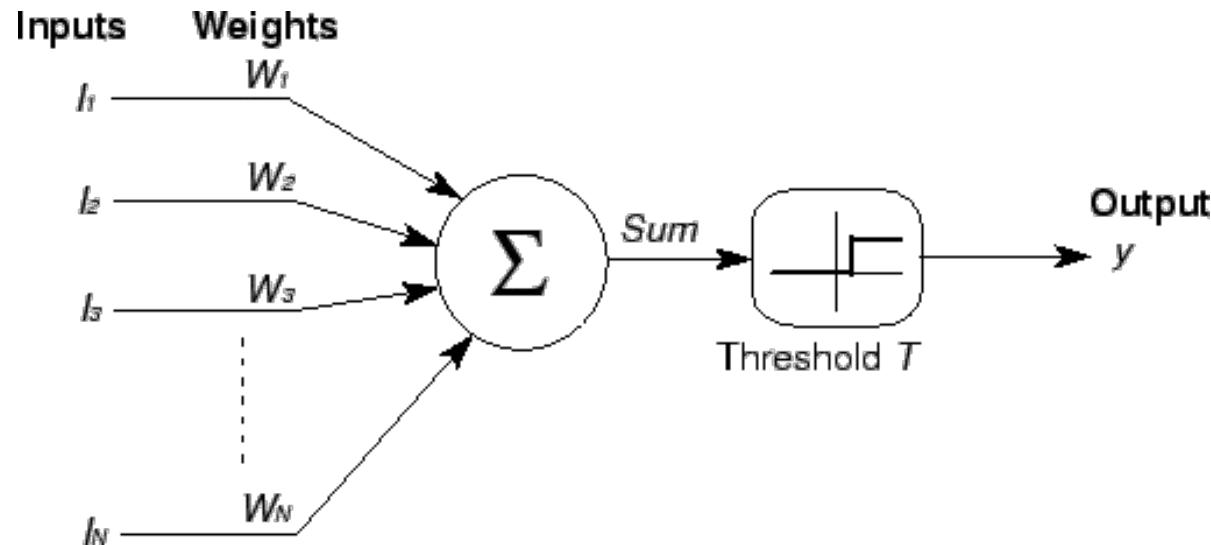
# Claude Shannon's Text Generation Algorithm

- Claude Shannon founded the field of information theory
- In his paper, Shannon proposed using a Markov chain to create a statistical model of the sequences of letters in a piece of English text.

*“To construct the Markov model of order 1, for example, one opens a book at random and selects a letter at random on the page. This letter is recorded. The book is then opened to another page and one reads until this letter is encountered. The succeeding letter is then recorded. Turning to another page this second letter is searched for and the succeeding letter recorded, etc. It would be interesting if further approximations could be constructed, but the labor involved becomes enormous at the next stage.”*

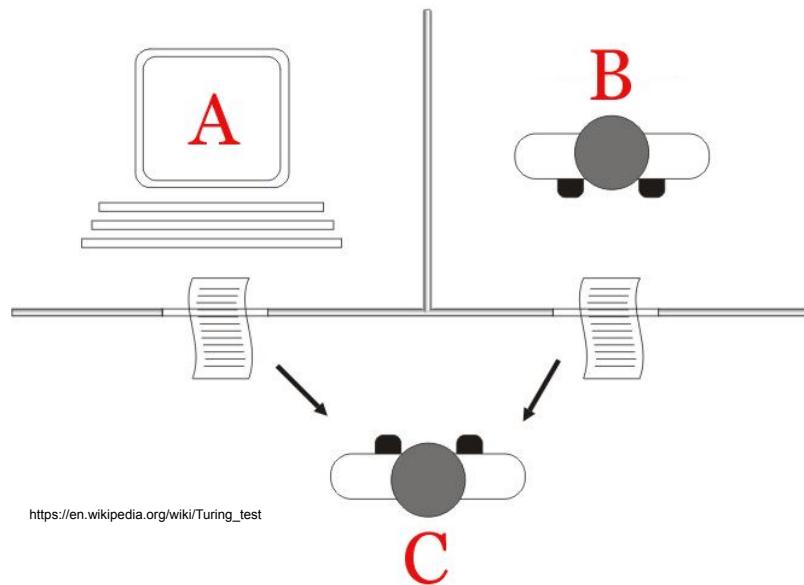


# McCullough and Pitts Neuron Model: 1944



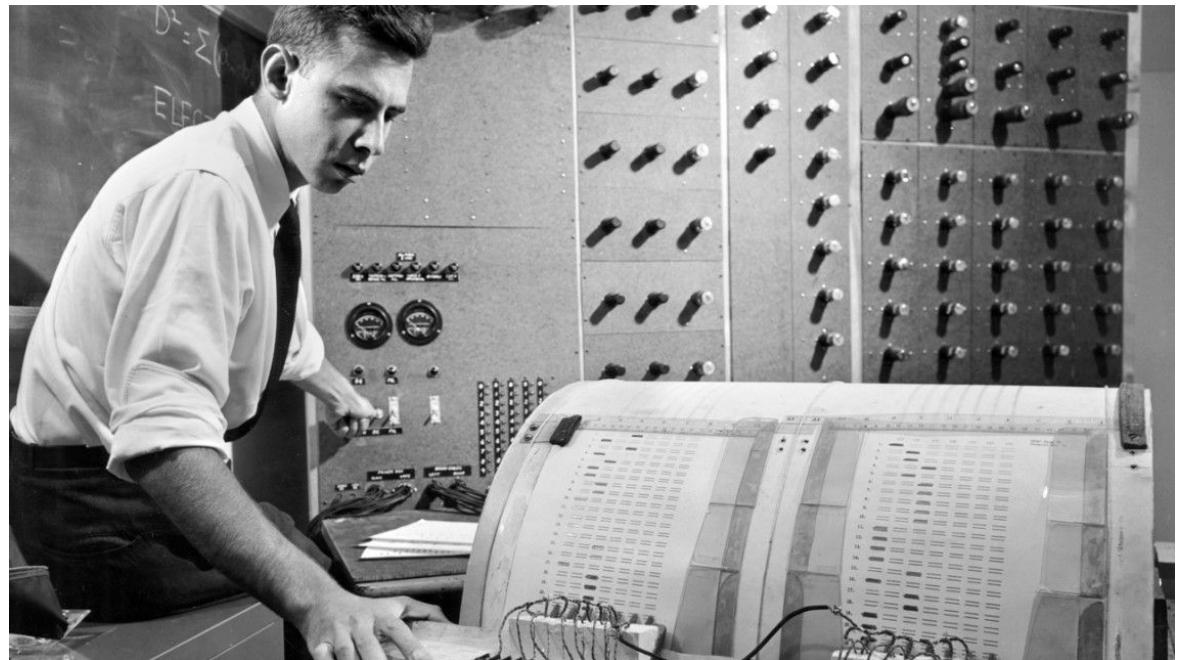
# Turing Test: 1950

- Alan Turing published his paper “[Computing Machinery and Intelligence](#),” which introduces the Turing Test, a test of a machine’s ability to exhibit intelligent behavior equivalent to, or indistinguishable from, that of a human.
- Criterion of intelligence
  - Can a machine exhibit intelligent behavior indistinguishable from that of a human
- Is it possible to distinguish between human-to-human and human-to-machine natural language conversations?



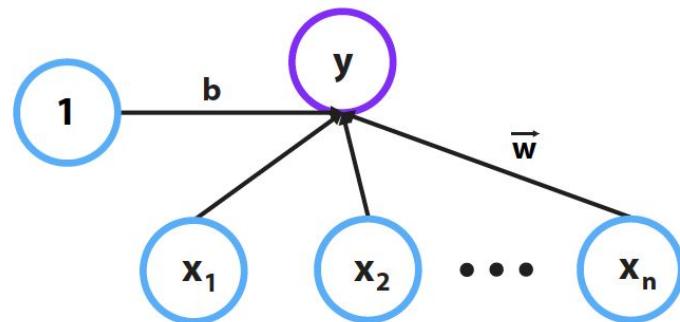
# Perceptron: 1957

- In the late 1950s, Frank Rosenblatt introduced the perceptron. It could be described as the first ever operational realization of a neural network. A perceptron represents a basic ML model that was designed to aid computers in learning from a diverse range of data.
- He described perceptron as the first machine “capable of having an original idea.”



# Perceptron

- Outputs the sign of the sum operator.  
It outputs 1 for Class 1 and -1 for  
Class 2.
- Binary classifier
- The first learning neural network



$$y = \text{sign}(\vec{x} \cdot \vec{w} + b)$$

# “Artificial Intelligence” Term Was Coined: 1956

- The Dartmouth Summer Research Project on Artificial Intelligence, considered the birth of AI, brings together more than 100 researchers from a variety of disciplines, including computer science, linguistics and philosophy, to discuss the possibility of creating machines that can think.
  - Among other events at the workshop, John McCarthy, an American computer scientist, proposed a name for a new academic discipline: *artificial intelligence*.

In the back row from left to right are Oliver Selfridge, Nathaniel Rochester, Marvin Minsky, and John McCarthy. In front on the left is Ray Solomonoff; on the right, Claude Shannon. The identity of the person between Solomonoff and Shannon remained a mystery for some time. THE MINSKY FAMILY



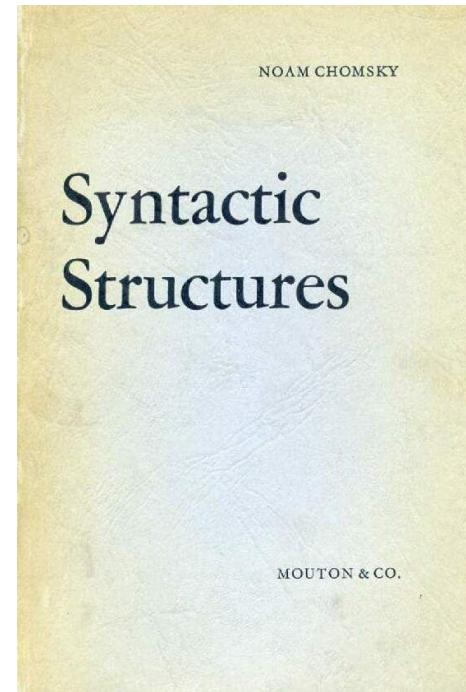
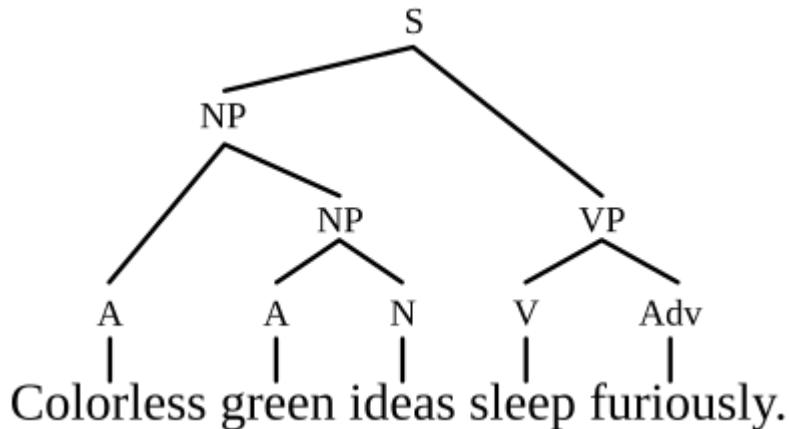
# Logic Theorist: 1956

- Logic Theorist is a computer program written in 1956 by Allen Newell, Herbert A. Simon, and Cliff Shaw able to perform automated reasoning,
- Known as the first artificial intelligence program
- Logic Theorist proved 38 of the first 52 theorems in chapter two of Whitehead and Bertrand Russell's Principia Mathematica, and found new and shorter proofs for some of them
- They continued their work 1957 to create General Problem Solver (GPS): a computer program created intended to work as a universal problem solver machine. GPS worked with means–ends analysis.



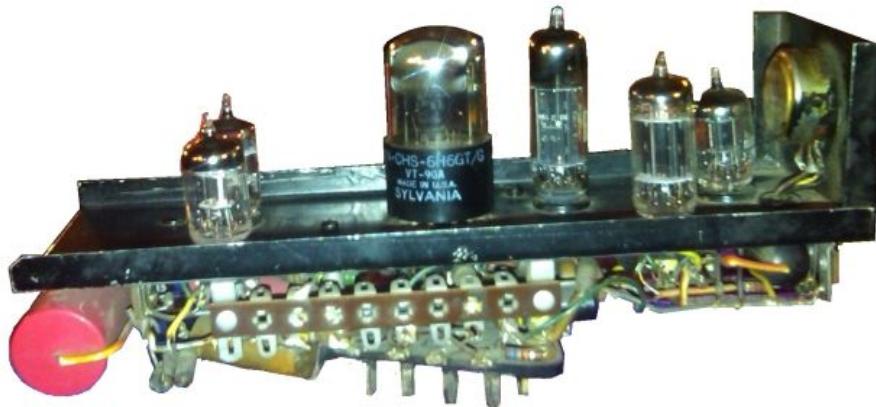
# Noam Chomsky releases “Syntactic Structures”: 1957

- Invented the “Phase–Structure Grammar”
  - Translates natural language (or human) sentences into a format that computers can understand and use



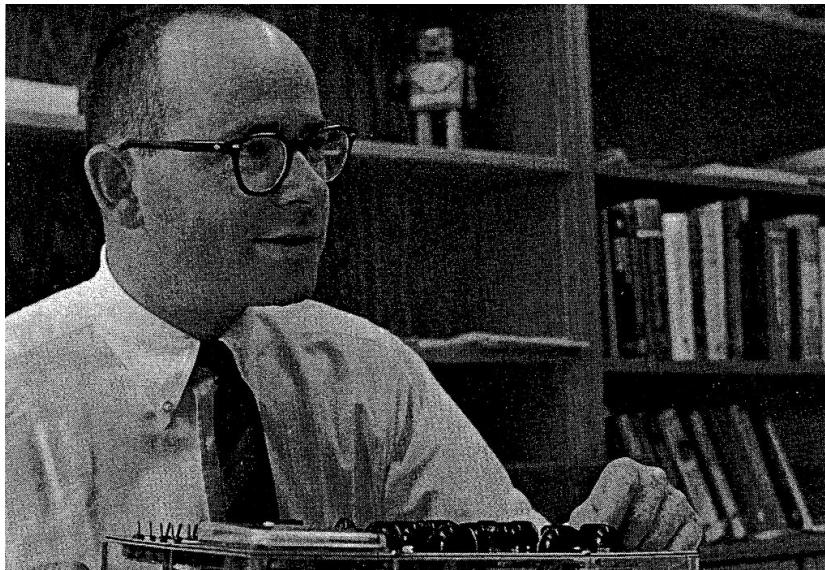
# SNARC: 1957

- The Stochastic Neural Analog Reinforcement Calculator (SNARC) is a neural-net machine designed by Marvin Lee Minsky.
- The machine itself is a randomly connected network of approximately 40 Hebb synapses



# ADALINE (Adaptive Linear Neuron) : 1959

- Shortly after Rosenblatt's perceptron, the ADALINE (Adaptive Linear Neuron) was introduced in 1959, by Bernard Widrow and Ted Hoff



Widrow demonstrating the "Knobby Adaline" device (1963)

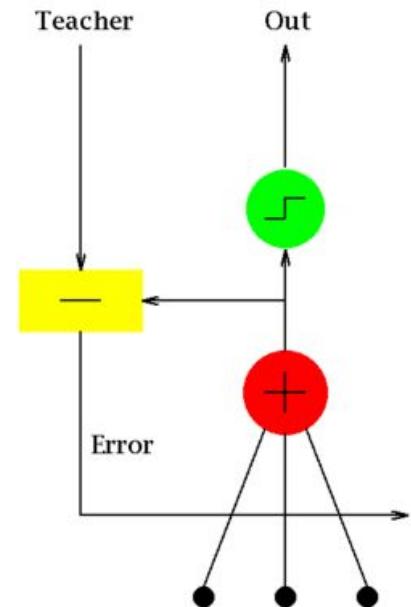
[https://en.wikipedia.org/wiki/Bernard\\_Widrow](https://en.wikipedia.org/wiki/Bernard_Widrow)

# Widrow-Hoff (delta) rule

- Assume that, we have access not only to the perceptron final output, but also to its activity before the sign function as well.

$$y = \text{sign}(\vec{x} \cdot \vec{w})$$

- Thus, we have access to  $x.w$



# “Machine Learning” Term Was Coined: 1959

- Arthur Samuel builds checkers program using an algorithm called “alpha-beta pruning.”
  - The game could learn on its own using a reward mechanism
  - Arthur Samuel introduced the *machine learning* term in 1959





Arthur  
Samuel  
(Our hero)



Richard Sutton  
(TD Learning)



Chris Bishop  
(Q-learning)



Demis Hassabis  
& Google  
(Deep Q-learning)

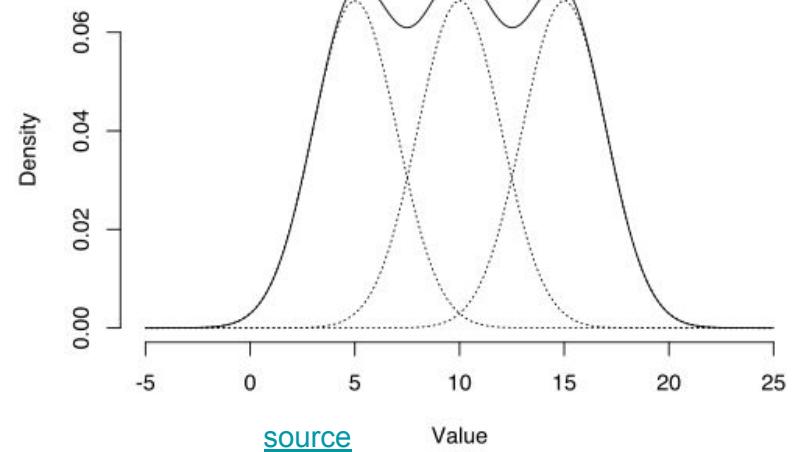
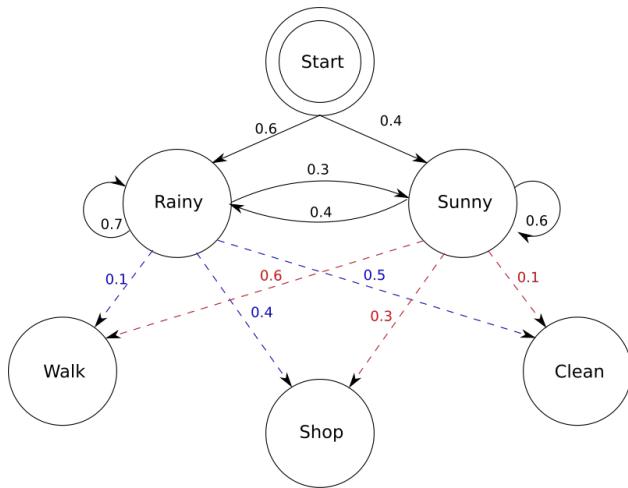
It all comes back to Samuel!

"Although this work was informal and had a number of flaws, it contained most of the modern ideas in reinforcement learning."

- Russell and Norvig

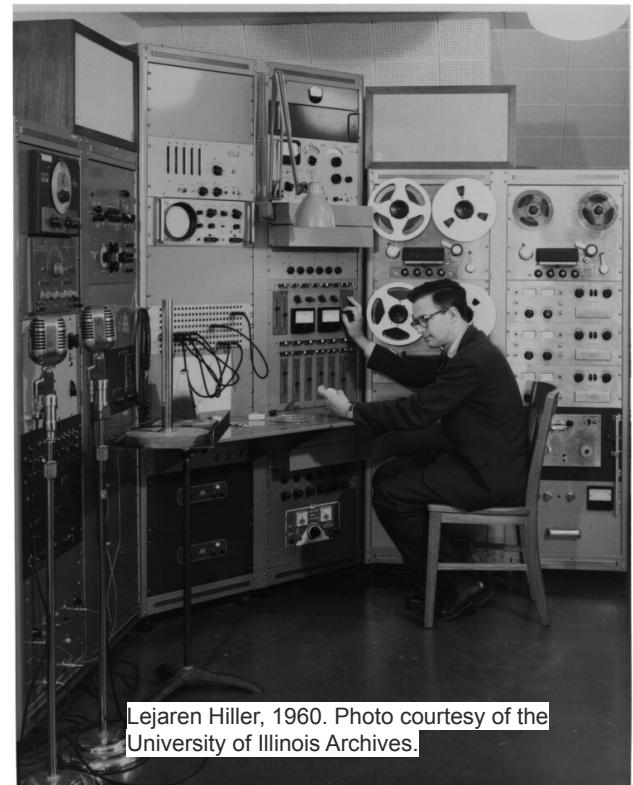
# HMMs and GMMs: 1950s

- Some of the first generative models were Hidden Markov Models (HMMs) and Gaussian Mixture Models (GMMs), build in the 1950s.
- Generated successive pieces of data, such as speech



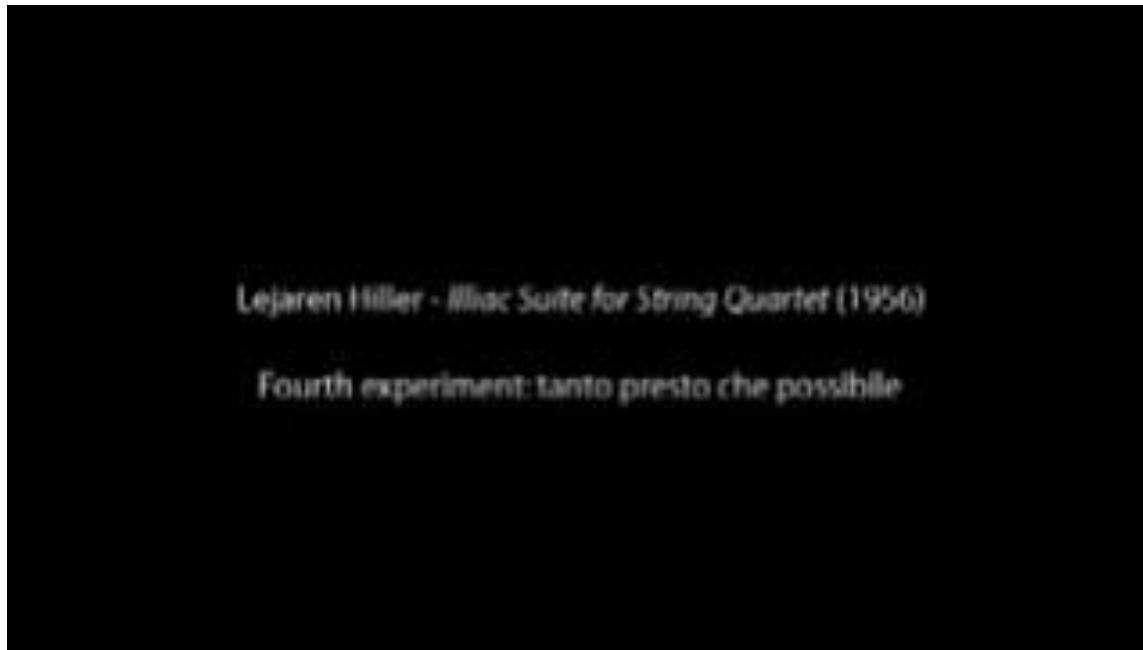
# Early Generative Creative Models: 1950s

- Harry F. Olson at Bell Labs used Markov chains in the 1950s to analyse the music of American composer Stephen Foster, and generate scores based on the analyses of 11 of Foster's songs.
- Lejaren Hiller used a computer at Princeton in 1955 to generate the Illiac Suite. Combined Markov chaining and application of rules of 16th century counterpoint.
- Hiller and Robert Baker also worked with Markov processes to produce their 'Computer Cantata' in 1963.



Lejaren Hiller, 1960. Photo courtesy of the University of Illinois Archives.

# Hiller's Iliac Suite for String Quartet [4/4]



<https://www.youtube.com/watch?v=QyqiSbbwHls>

# Hiller and Robert Baker's 'Computer Cantata': 1963



<https://www.youtube.com/watch?v=85fvyWJFq20>

# ELIZA (1964-1966): The First ChatBot

- ELIZA was an early chatbot system that convinced users that they were talking to an actual human
- DOCTOR, simulated a psychotherapist by simply parroting back at patients what they had just said

```
Welcome to
EEEEEELL      IIII    ZZZZZZ   AAAAAA
EE     LL      II      ZZ      AA     AA
EEEEEELL      II      ZZZ      AAAAAAAA
EE     LL      II      ZZ      AA     AA
EEEEEELLLLL  IIII    ZZZZZZ   AA     AA

Eliza is a mock Rogerian psychotherapist.
The original program was described by Joseph Weizenbaum in 1966.
This implementation by Norbert Landsteiner 2005.

ELIZA: Is something troubling you ?
YOU: Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU: They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU: Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU: He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU: It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:
```

# ELIZA (1964-1966): The First ChatBot

- Eliza went viral again in 2018, because she was featured in an episode of the TV show Young Sheldon. (The episode, "A Computer, a Plastic Pony, and a Case of Beer,")
- Try Eliza here: <https://web.njit.edu/~ronkowit/eliza.html>



# Early Generative AI Art

- Harold Cohen created AARON a generative AI system to generate paintings and showed in Exhibits
- In development between 1972 and the 2010s
- The software was not open source, its development ended with Cohen's death in 2016



# AARON Generative Art Computer Program

- Cohen's work on AARON began in 1968 at the University of California, San Diego.
- He initially wrote AARON in the C programming language but later converted to Lisp, citing that C was "too inflexible, too expressive, to deal with something as conceptually complex as color."



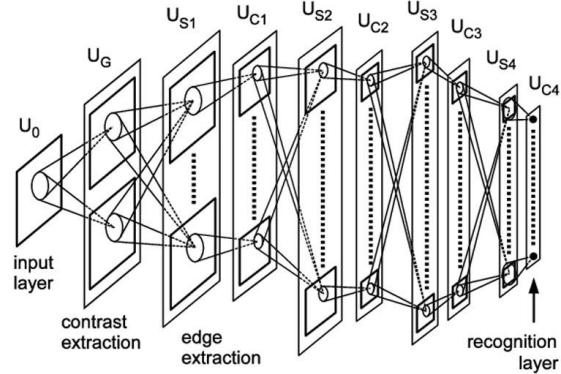
# AI Winter I

- The first AI winter spanned from 1973 to 1979. Promises were made but not fulfilled, leading to disappointment among funding agencies like DARPA, NRC, and the British government due to the lack of progress.



# Neocognitron: 1979

- Kunihiko Fukushima suggested developing a hierarchical, multilayered artificial neural network, that he named Neocognitron.
- The first deep learning neural network. Applied to computer vision and handwritten character recognition.



# AI winter II

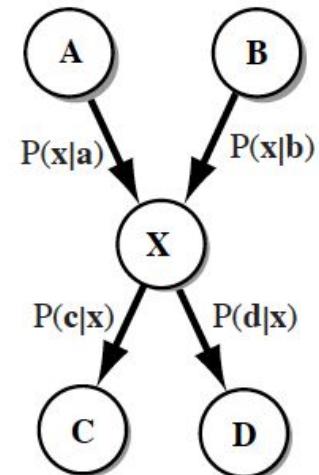
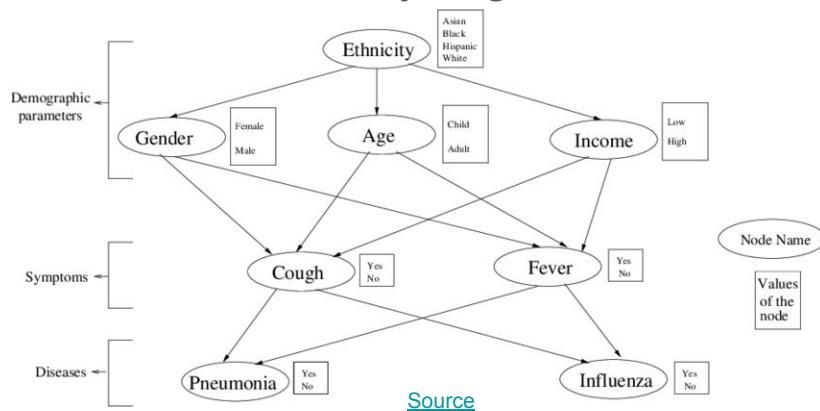
- The second AI winter, from 1984 to 1990, hindered AI and generative AI development. Intense frustration over unmet expectations led to AI being regarded as pseudoscience and spoken of with contempt. Skepticism grew, and most funding for AI and deep learning research was cut.

Generated by GPT-4o



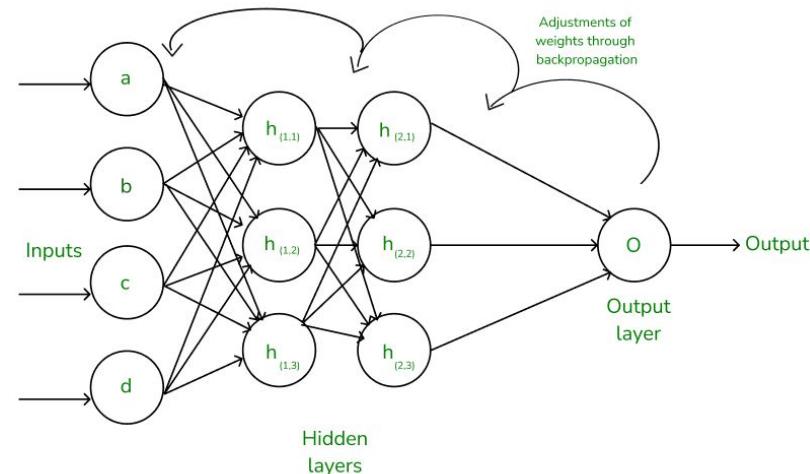
# Bayesian Networks:1985

- Jude pearl's bayesian networks allowed content generation in specific styles
- A Bayesian network represents a group of variables and their conditional dependencies.
- Probabilistic Graphical Models
- Valuable tool for analyzing causal relationships



# Backpropagation: 1986

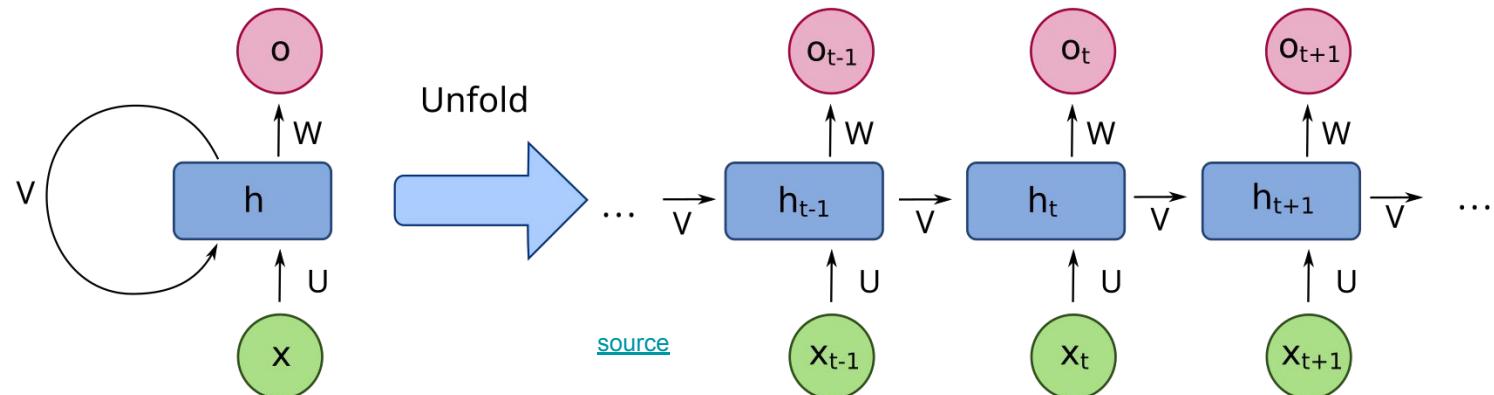
- Modern backpropagation introduced by Seppo Linnainmaa as "reverse mode of automatic differentiation" for discrete connected networks of nested differentiable functions (1970)
- David Rumelhart and Geoffrey Hinton introduced a new way of training neural networks using the backpropagation technique which was originally developed in the 1970s by
- This is the most fundamental technique enabling modern deep learning architecture



[source](#)

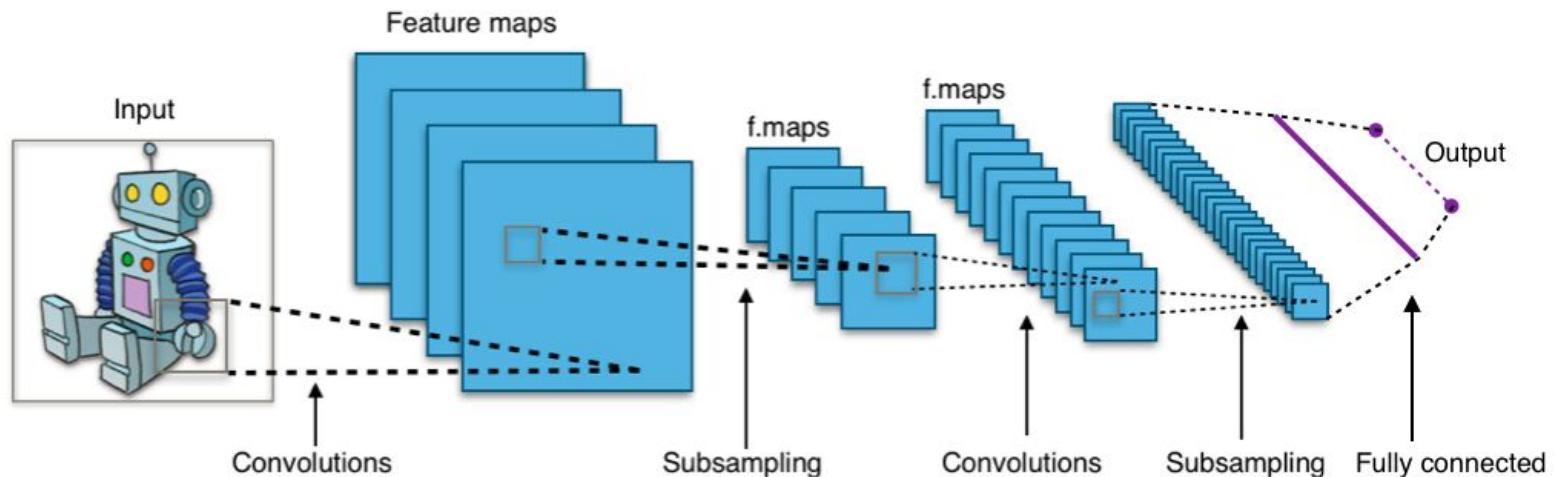
# RNNs: 1980s

- Recurrent neural networks (RNNs) can model longer dependencies and allowed generating longer sentences
- But still they had memory loss problem for long sequences
- Vanishing Gradient problem



# Convolutional Neural Networks: 1989

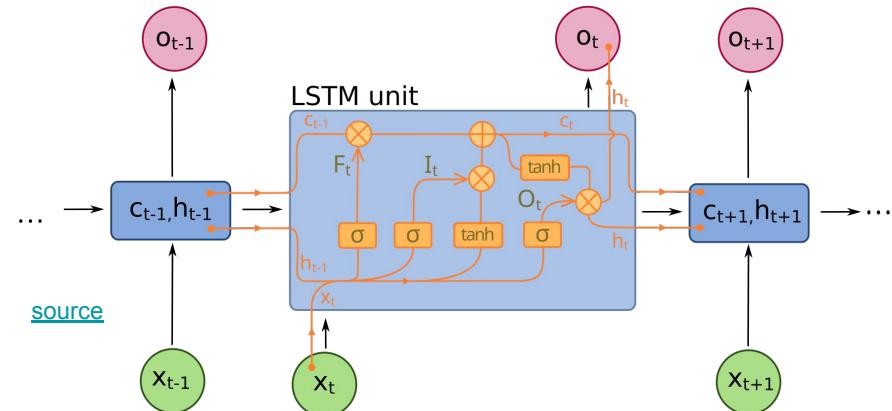
- Deep learning became a functional reality in the year 1989, when Yann LeCun and his team used a backpropagation algorithm with neural networks to recognize handwritten ZIP codes.



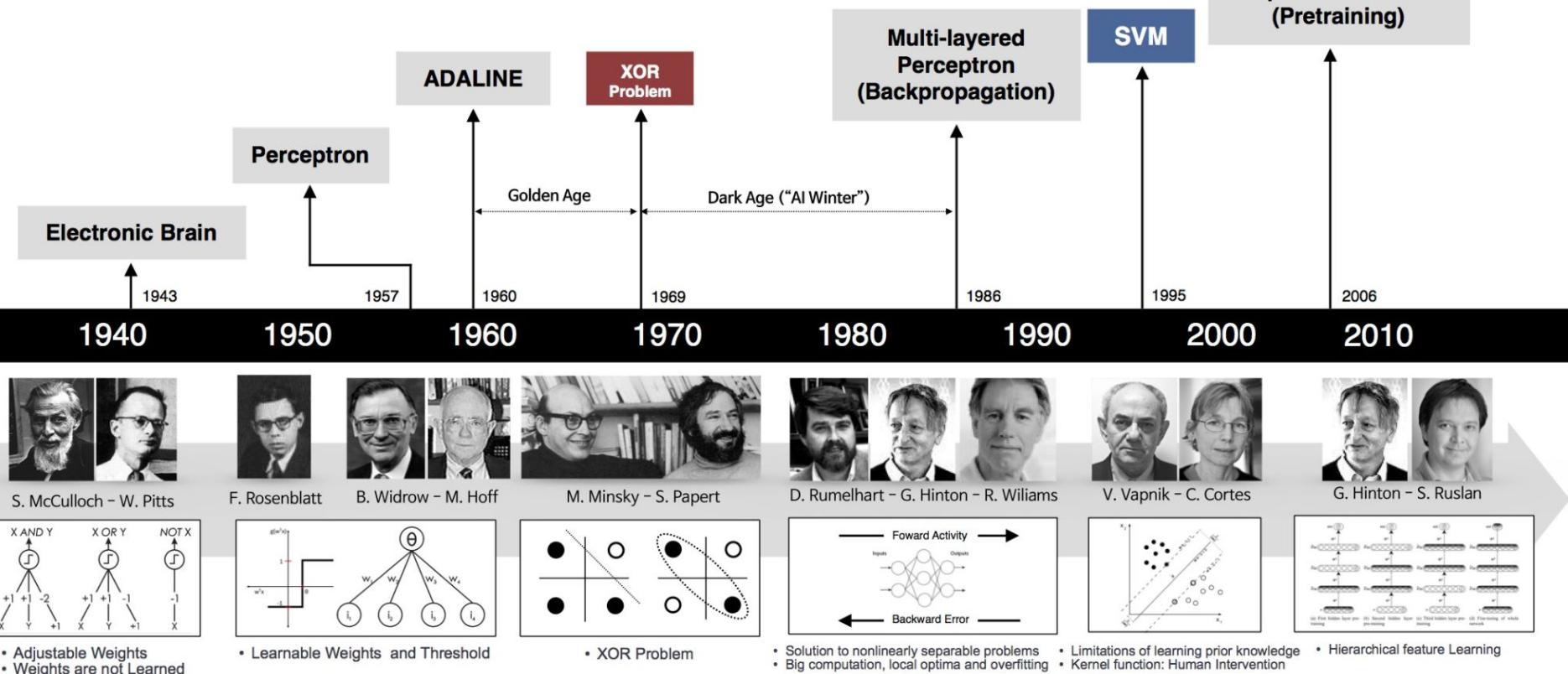
[source](#)

# LSTM: 1997

- Juergen Schmidhuber and Sepp Hochreiter created the “long short-term memory” (LSTM )
- Addressed the vanishing gradient problem and allowed for longer sequences
- But still limited in terms of how far back it can go in terms of memory and context
- Slow to train as the data is fed sequentially

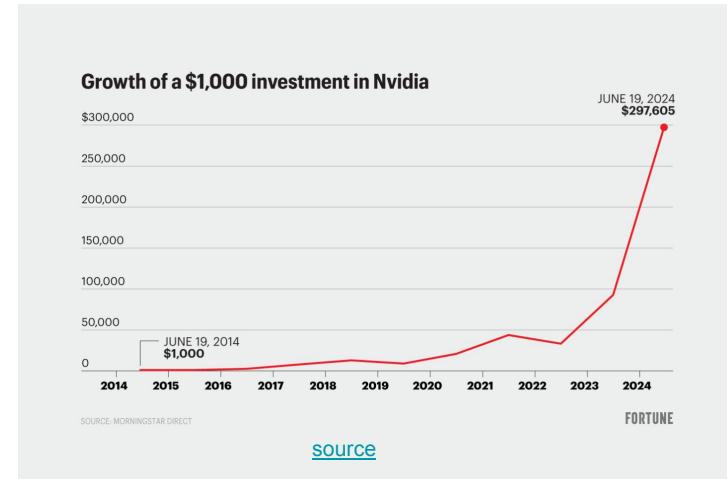


# Neural Networks Development Timeline in Summary



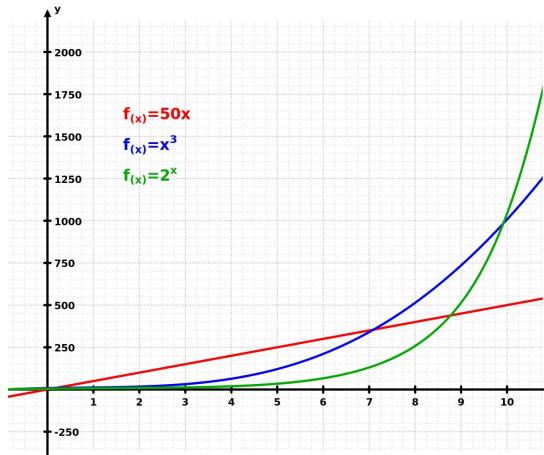
# The Invent of GPUs: 1990s

- The computer gaming industry played a crucial role in the evolution of generative AI by introducing 3D graphics cards in the early 1990s, which later evolved into GPUs
- Nvidia's development of the GeForce 256 GPU in 1999, which significantly increased computational speeds, revealed GPUs' potential beyond gaming, particularly in artificial neural networks
- GPUs have become essential in machine learning, offering around 200 times more processors per chip compared to central processing units (CPUs)



# Abundance of Data from WWW

- By the 2000s and 2010s, Large data availability and computational power made deep learning practical.
- "There was 5 exabytes of information created between the dawn of civilization through 2003, but that much information is now created every 2 days, and the pace is increasing." (Eric Schmidt)



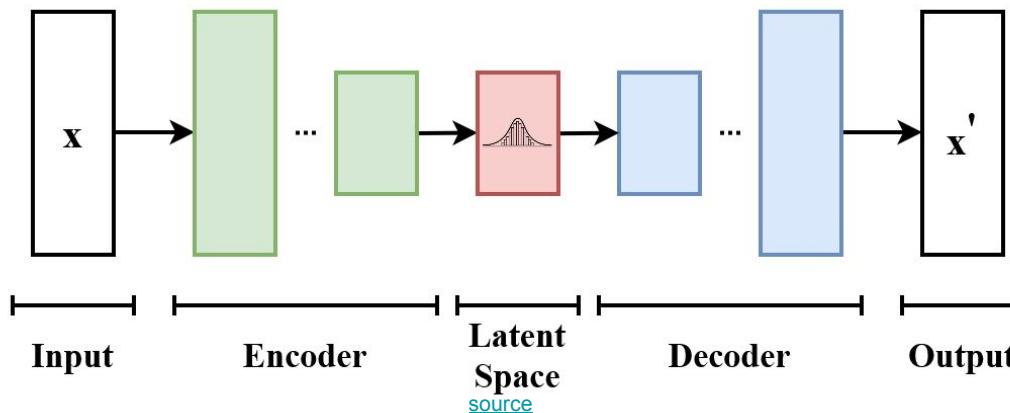
# Siri: 2011

- Considered as the first functional virtual assistant released with the iPhone 4S.



# Variational AutoEncoders: 2013

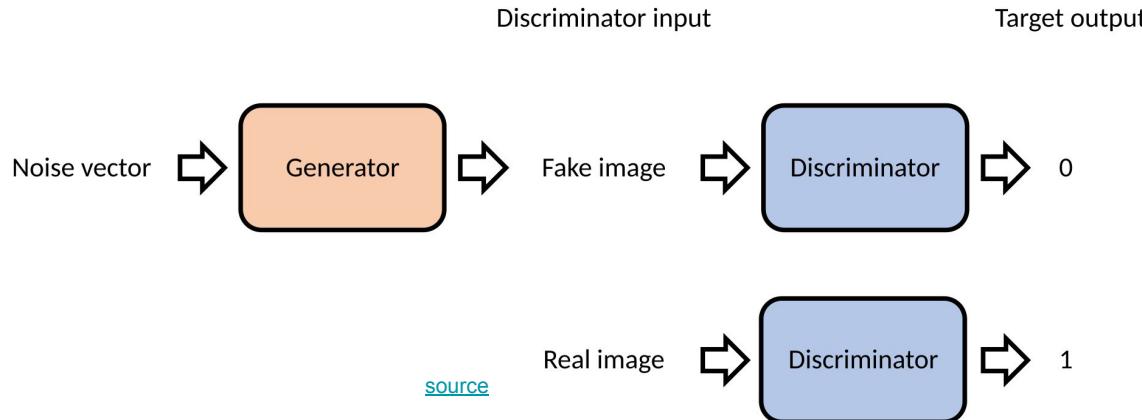
- Part of the families of probabilistic graphical models and variational Bayesian methods.
- [Auto-Encoding Variational Bayes DP Kingma, M Welling](#)



# Generative Adversarial Networks (GANs): 2014

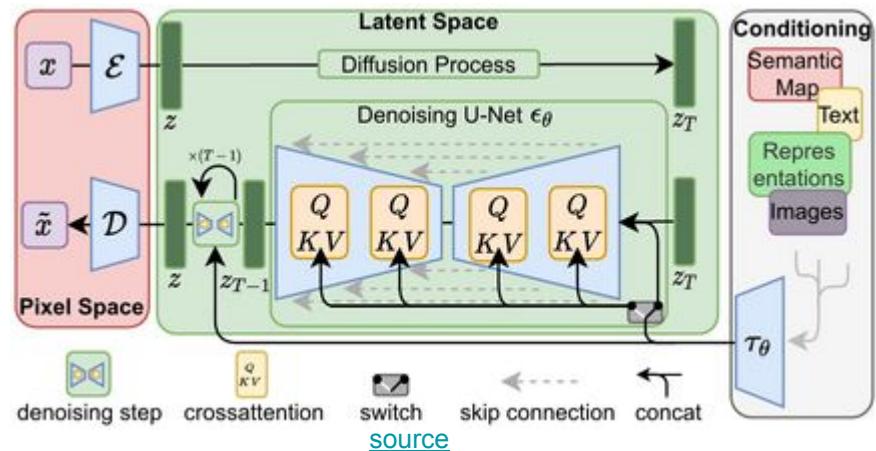


- The GAN paper was published two months after the VAE: [Ian J. Goodfellow and Jean Pouget-Abadie and Mehdi Mirza and Bing Xu and David Warde-Farley and Sherjil Ozair and Aaron Courville and Yoshua, "Generative Adversarial Networks"](#)
- One of the fundamental breakthroughs in generative AI is the creation of Generative Adversarial Networks (GANs) in 2014 by Ian Goodfellow.
- A generative adversarial network (GAN) uses two networks.
  - The discriminator is trained to differentiate between real and generated data
  - The generator creates synthetic data, attempting to mimic real data.



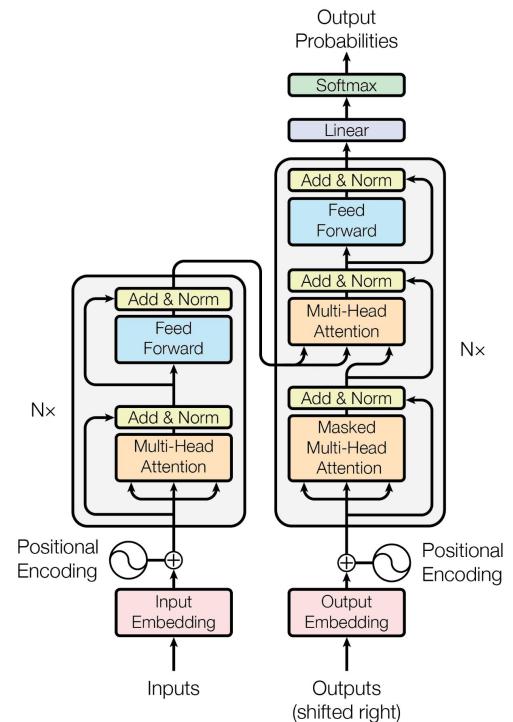
# Diffusion Models 2015

- Deep Unsupervised Learning using Nonequilibrium Thermodynamics, Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, Surya Ganguli
- Function by adding noise into the existing training data and then reversing the process to reconstruct the data;
- These models have shown great success in generating high-quality images and other complex data types.



# Transformers and GPT:2017

- "Attention Is All You Need," Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin
- Major advancements in generative models compared to older Long-Short Term Memory models
- All the tokens can be fed in parallel to the model leading to major speedups
- The first generative pre-trained transformer, known as GPT-1, was released in 2018.



# DALL-E: 2021

- Capable of generating high-quality and realistic images, as well as being able to perform some additional functions, such as adding, replacing, or removing certain objects, or generating alternative variants of a given image according to a textual description.
- Continuously trained on new data.
- Uses three neural networks:
  - CLIP (Contrastive Language-Image Pre-training) is responsible for recognizing text and creating a sketch of the future image;
  - GLIDE is responsible for converting the sketch into a final low-resolution image
  - The third neural network is responsible for increasing the resolution of the AI-generated art and adding additional details.

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



Théâtre D'opéra Spatial, an image generated with Midjourney: 2022



# Stable Diffusion:2022

- Stable Diffusion, developed by Stability.ai, is an open-source neural network that generates images from text prompts
- It also assists in completing sketches and editing original images.

Prompt: "a photograph of an astronaut riding a horse"



# GPT4: 2023

- Capable of generating longer texts up to 25000 words.
- Two versions of GPT-4
  - Context windows of 8,192 tokens
  - Context windows of 32,768 tokens
- A significant improvement over GPT-3.5 and GPT-3, which were limited to 4,096 and 2,049 tokens respectively
- Multimodal model
  - It can accept both images and text as input, such as summarize text from screenshots and answer exam questions with diagrams.
  - It now also interacts with users through spoken words and responds to images facilitating more natural conversations

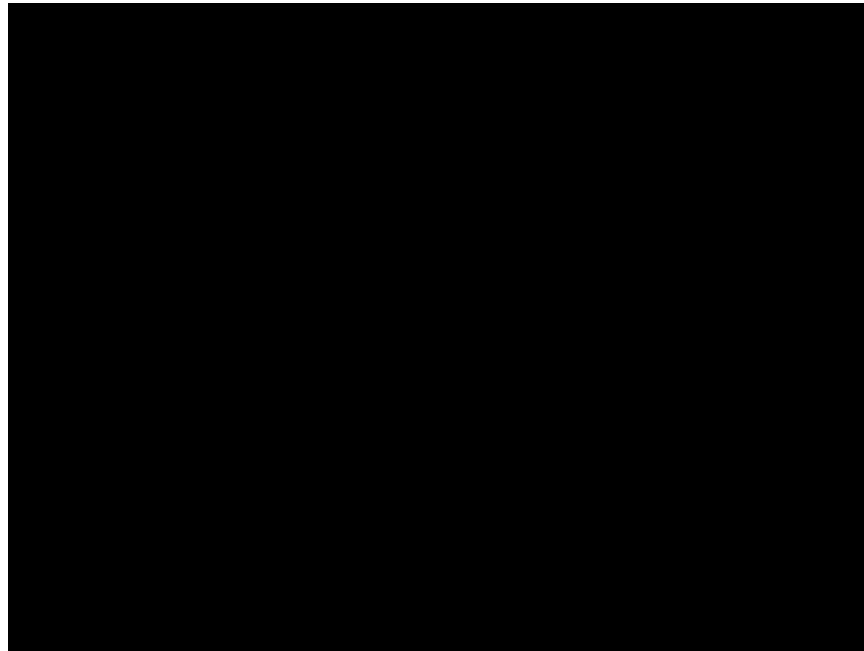
# Text-to-Video Generative Models

- Runway Gen-2
- Sora



# Text-to-3D

- DreamFusion, Magic3D & Get3D



[source](#)

# Github Copilot

- GitHub's Copilot, built on OpenAI's Codex and trained on all public GitHub code repositories, gained widespread adoption. It assists developers by converting natural language into executable software code.

The screenshot shows the GitHub Copilot interface integrated into a GitHub repository. On the left, there's a sidebar with file navigation and a search bar. The main area has tabs for 'parse\_expenses.py', 'addresses.rb', and 'sentiments.ts'. The 'parse\_expenses.py' tab is active, displaying Python code. A message from 'monalisa' asks for unit tests for a function. GitHub Copilot has provided a template for a test class:

```
import unittest
import datetime

class TestParseExpenses(unittest.TestCase):
    def test_parse_expenses_with_valid_input(self):
        expenses_string = '2023-01-02 -34.01 USD\n2023-01-03 2.59 DKK'
        expected_output = [(datetime.datetime(2023, 1, 2),
                           float(-34.01), 'USD'),
                           (datetime.datetime(2023, 1, 3),
                           float(2.59), 'DKK')]
        self.assertEqual(parse_expenses(expenses_string),
                        expected_output)

    def test_parse_expenses_with_empty_input(self):
        expenses_string = ''
        expected_output = []
        self.assertEqual(parse_expenses(expenses_string),
                        expected_output)

if __name__ == '__main__':
    unittest.main()
```

Below the code, a note states: "The code assumes that the datetime module is imported. The test cases cover different scenarios such as valid input, empty input, input with comments, invalid date format, and invalid value format." At the bottom, there's a text input field with placeholder "Ask a question or type '?' for commands" and a "source" button.

history of the  
entire AI field,  
i guess



<https://www.youtube.com/watch?v=b9chqJ2TgzA>