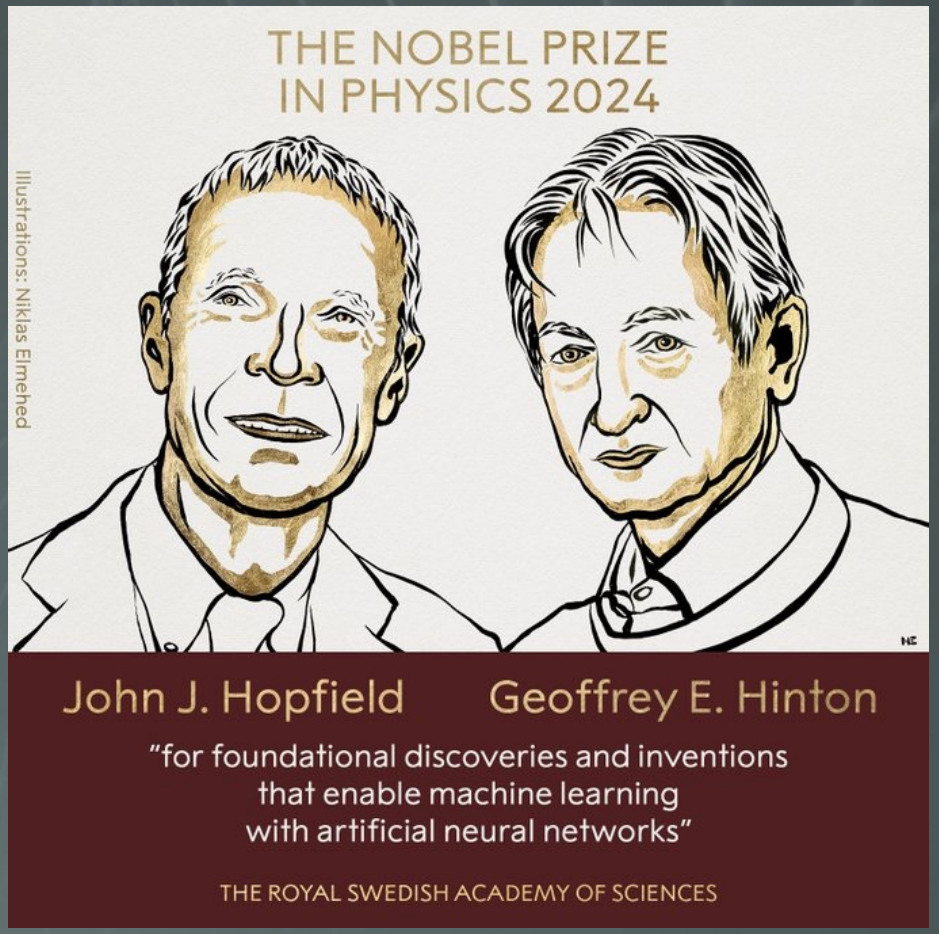
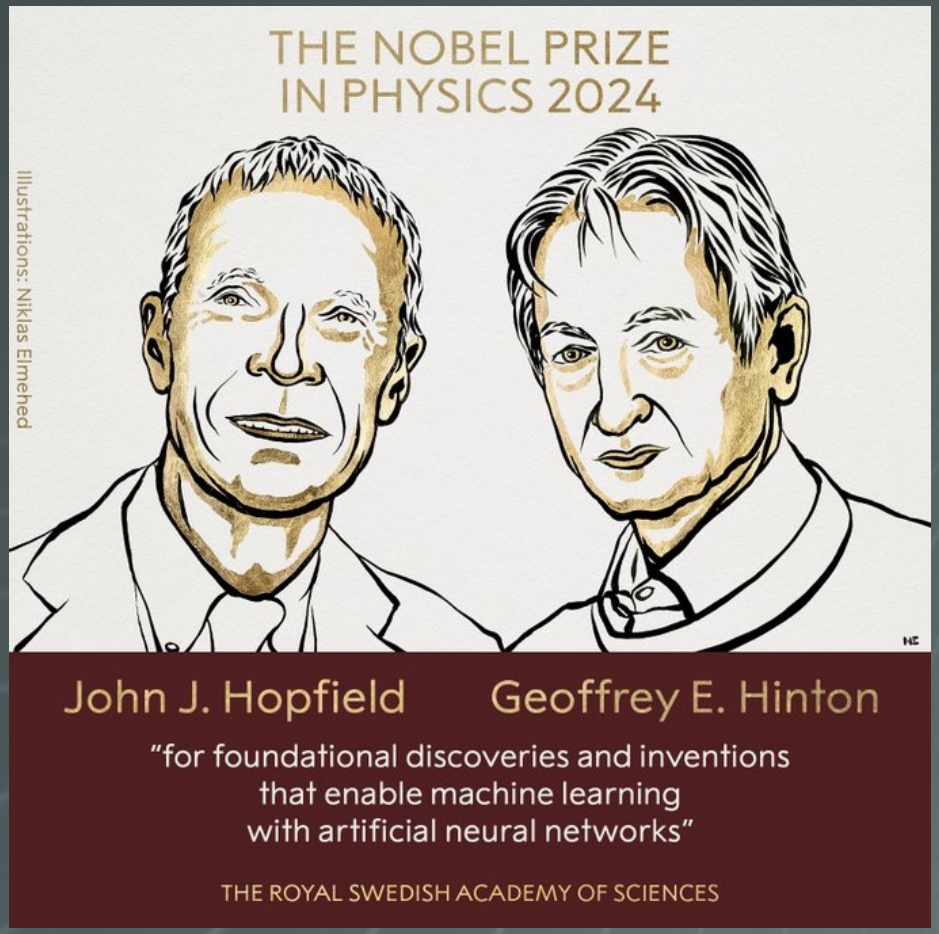
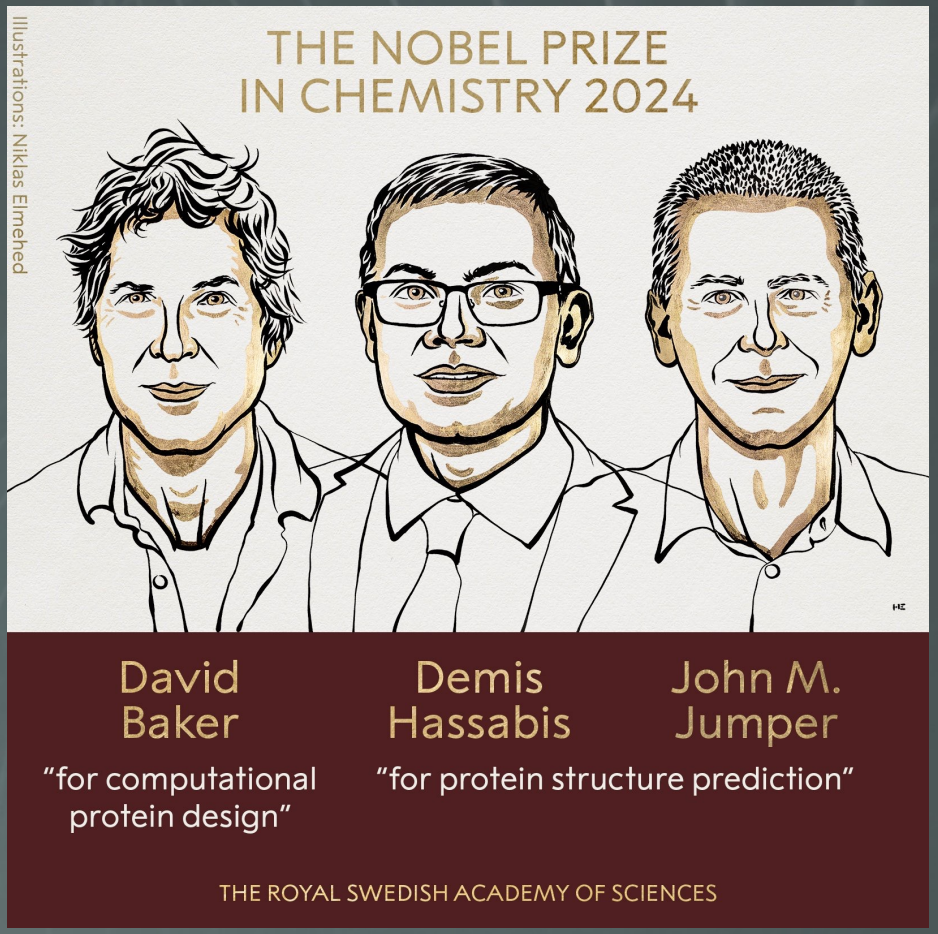
The Physical Foundations of Artificial Intelligence

Gaetano Salina
Istituto Nazionale di Fisica Nucleare
Sezione di Roma Tor Vergata




THE NOBEL PRIZE
IN PHYSICS 2024
Illustrations: Niklas Elmehed
John J. Hopfield
Geoffrey E. Hinton
"for foundational discoveries and inventions
that enable machine learning
with artificial neural networks"
THE ROYAL SWEDISH ACADEMY OF SCIENCES


THE NOBEL PRIZE
IN CHEMISTRY 2024
Illustrations: Niklas Elmehed
David Baker
Demis Hassabis
John M. Jumper
"for computational protein design"
"for protein structure prediction"
THE ROYAL SWEDISH ACADEMY OF SCIENCES

ACCADEMIA NAZIONALE DEI LINCEI

«CONFERENZE ISTITUZIONALI»

GIORGIO PARISI

Presidente emerito dell'Accademia Nazionale dei Lincei
Premio Nobel per la Fisica 2021

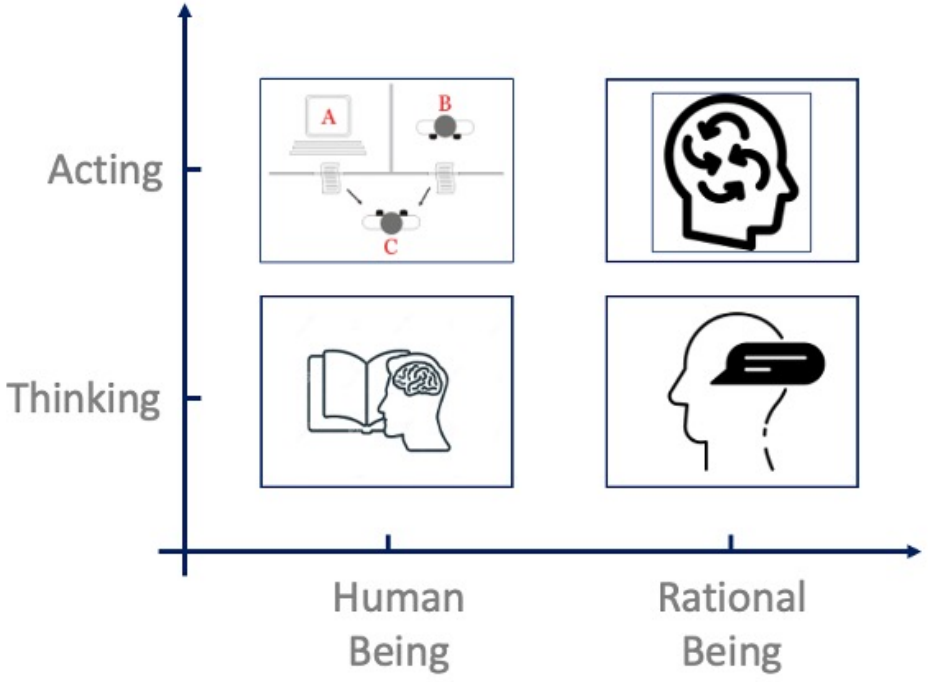
Le radici fisiche della moderna intelligenza artificiale

-46:01

https://www.lincei.it/it/video/14022025-le-radici-fisiche-della-moderna-intelligenza-artificiale

AI & Machine Learning: AI & Proposed Taxonomy

- **Thinking like humans**
  Human Cognitive Processes, focuses on understanding and modeling human thought
- **Acting like humans**
  Human-like Behavioral Test, focuses on replicating human behavior
- **Thinking rationally**
  Formal Logical Reasoning, focuses on correct and systematic logical reasoning
- **Acting rationally**
  Rational Goal-directed Action, focuses on taking the right goal-directed actions

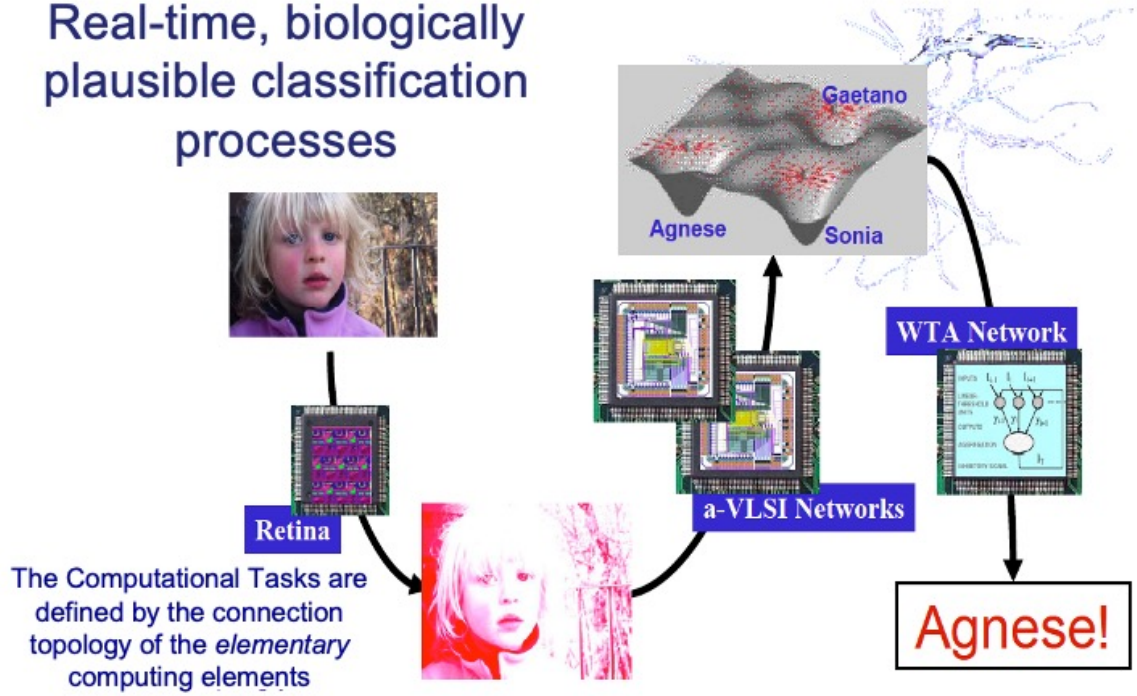


A Modern Approach, S. Russell and P. Norvig, Prentice Hall, 2020

Some notes on Computational Paradigms

A look at the past to guess the future

Real-time, biologically plausible classification processes



Neuromorphic Hardware:

It encompasses any electronic device which **mimics** the natural biological structures of our nervous system.

D.J. Amit and G. Salina 1990-2006

- The goal is to impart cognitive abilities to machines by implementing neurons in silicon.
- Due to its superior energy efficiency and inherent parallelism, this approach is being considered as an alternative to conventional computational architectures

Today
~~Tomorrow's~~ Talk Overview

- *Biological Neural Networks as a Functional Paradigma*
  Brain inspired approach
- *McCulloch-Pitts Formal Neuron*
  Propositional Calculus
- Multilayer Perceptron (MLP)
  Stacked Artificial Neuron Layers
- *Feed Forward and Attractors Neural Network*
  Layers, Dynamics, and Memory
- *Hopfield Model*
  Dynamic Neural Architecture
- *All You Need is Attention… or Hopfield*
  Physics before AI Models

Biological Neural Networks as a Functional Paradigm

- I've seen things you people wouldn't believe. Attack ships on fire off the shoulder of Orion.

- I've seen c-beams glitter in the dark near the Tannhauser gate.

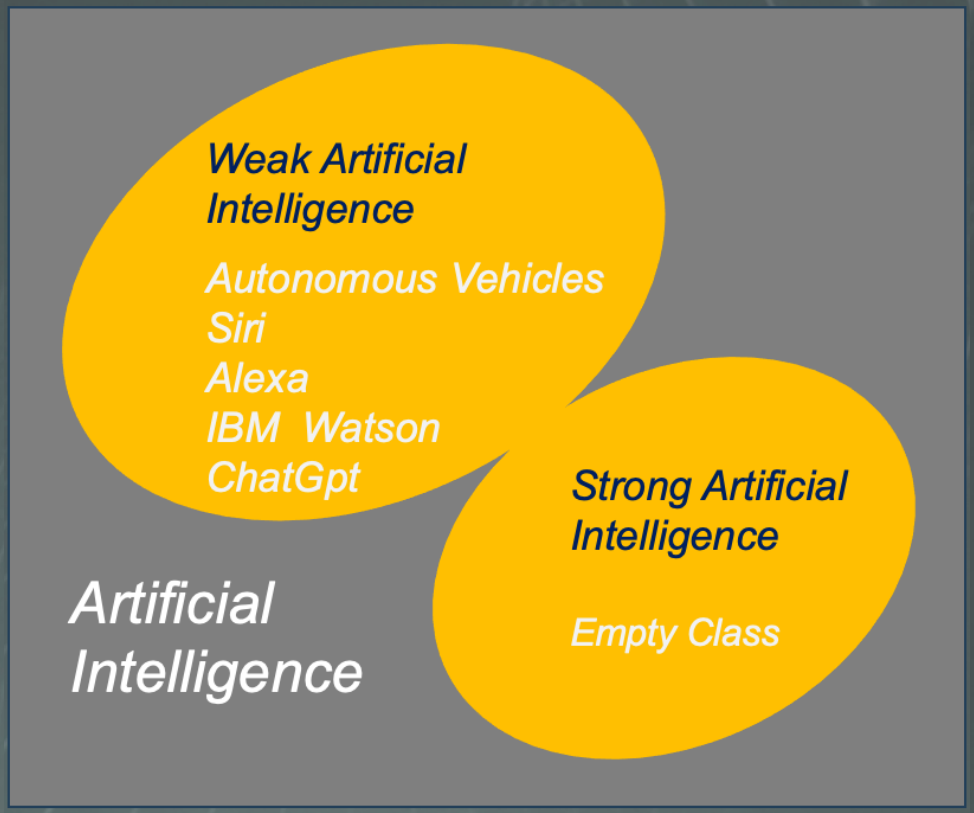
- All those moments will be lost in time… like tears in the rain.

- Time to die.

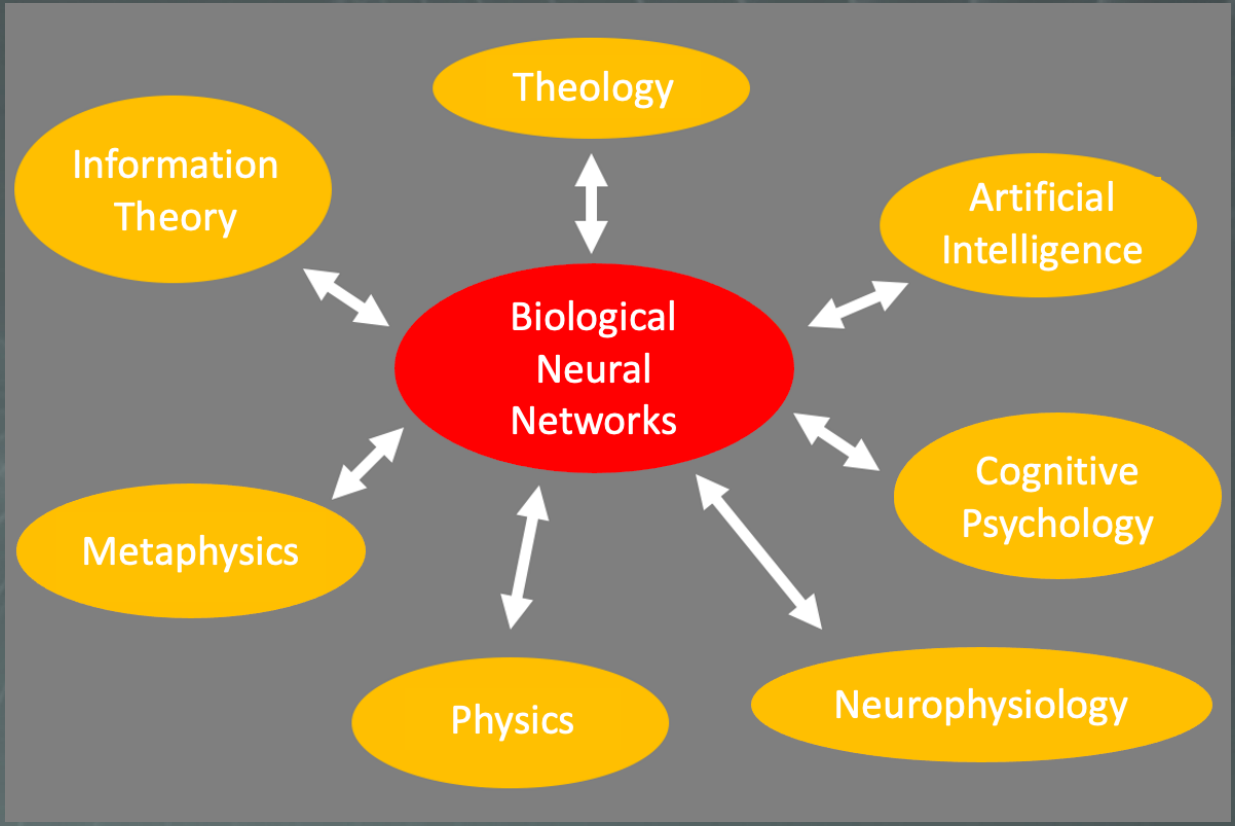
Blade Runner

Can an artifact be capable of analyzing, classifying, and retaining information, of learning continuously from experience without being explicitly programmed?

Can an artifact possess self-consciousness and be capable of thought?



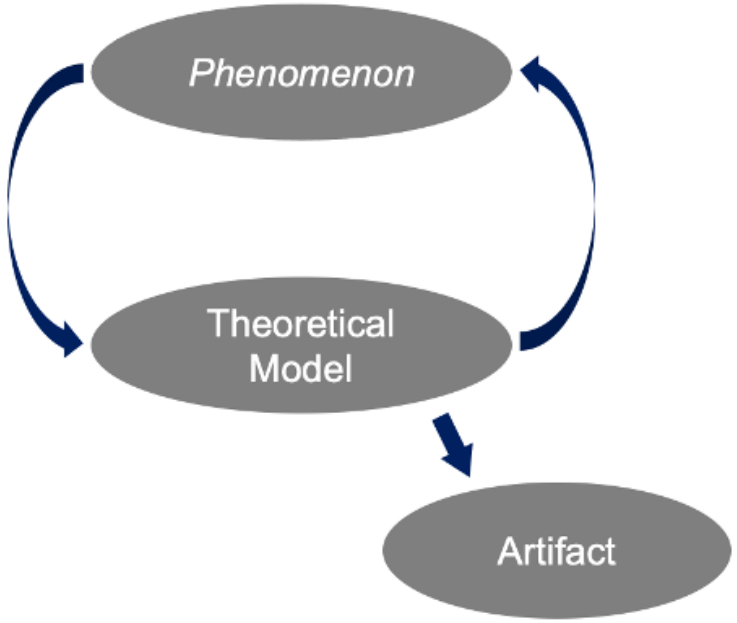
The study of physical and functional characteristics constitutes the starting point for the formulation of an answer, an inquiry that is, by its very nature, interdisciplinary.



Constructing a model of a Natural Phenomenon

The scientific method transforms observation, data, into knowledge

- To integrate apparently disparate empirical findings
- To predict previously unobserved aspects of the system's behavior
- The possibility of constructing an artifact that implements (subsets of) the system's relevant behavioral characteristics



Some Epistemological Considerations

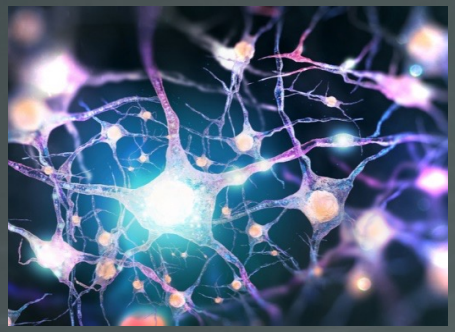
- Reductionism: all natural phenomena are reducible to physical laws, without independent languages or levels of description, but only to more or less detailed levels and languages.
- Reductionism is employed not as an epistemological dogma, but as an extremely productive idea (D.O. Hebb, 1980).
- Mental phenomena are nothing more than the expression of a highly complex structure operating according to fairly simple principles (A. Turing, 1950)

Some Epistemological Considerations

- Identifying a Standard Methodology
- All sciences, …, depend on their philosophical assumptions, …, and it is very easy for philosophical ideas regarding the soul, …, or regarding determinism and free will, to influence the main directives of Theory (D.O. Hebb, 1980).
- As long as these ideas remain implicit, they are dangerous. Make them explicit, and perhaps they can be neutralized.

Biological Neural Networks: Structure

Structurally Highly Complex System



10^9 -10^{10} Neurons

10^4 -10^5 Sinapses/Neurons

The reaction time of a neuron is a few milliseconds

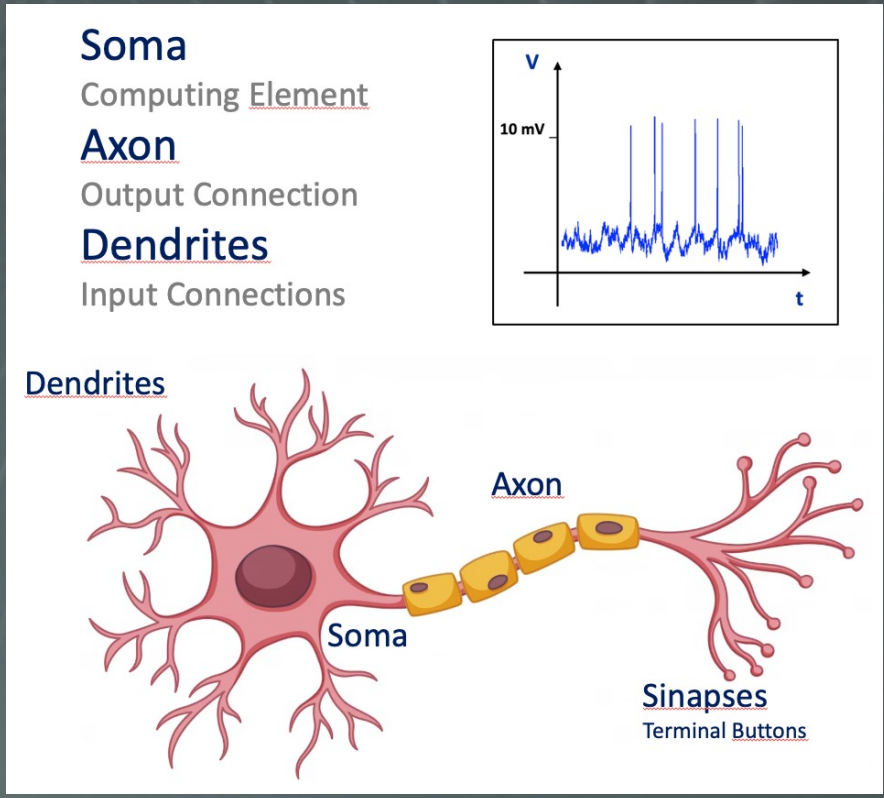
Structure: Input – Computation – Output

Stimulus → Sense → Transducer → Processing → Motor Neurons → Motor System → Spinal Cord → Muscle

Biological Neural Networks: Neuron

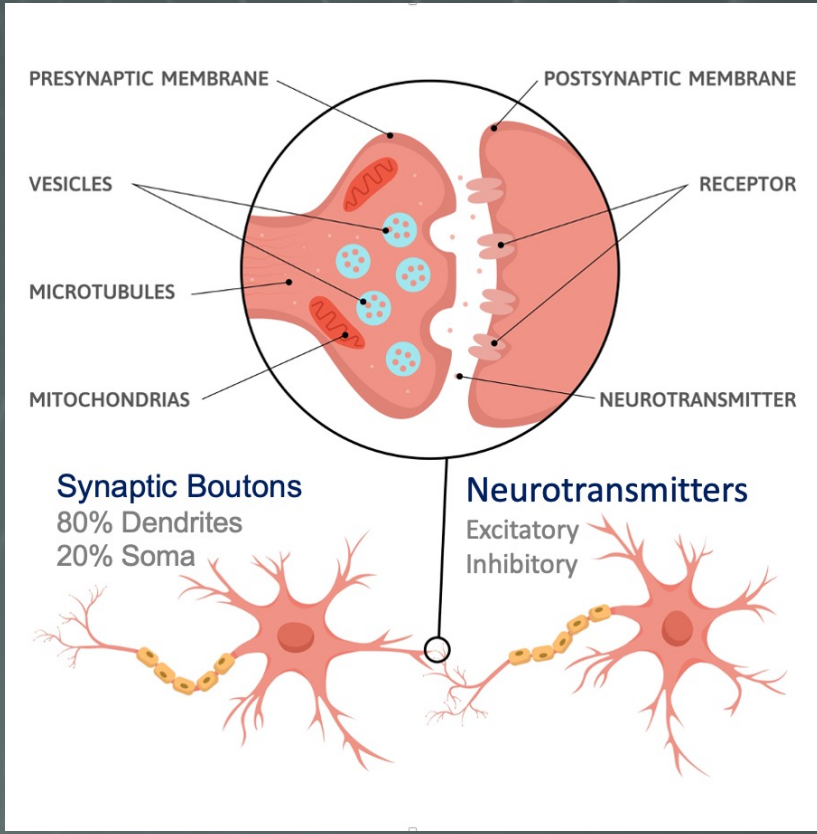
A difference in sodium and potassium ion concentrations creates an electric potential difference (approximately –70 mV) across the soma's cell membrane.

If the electric potential V, resulting from inputs from other neurons, exceeds the membrane potential, the neuron emits, for a few milliseconds, a train of impulses along the axon.



Biological Neural Networks: Sinapse

When an electrical spike propagates along the axon, the vesicles present in the synaptic bouton rupture, releasing neurotransmitters into the synaptic cleft, which tend to excite or inhibit the postsynaptic neuron.



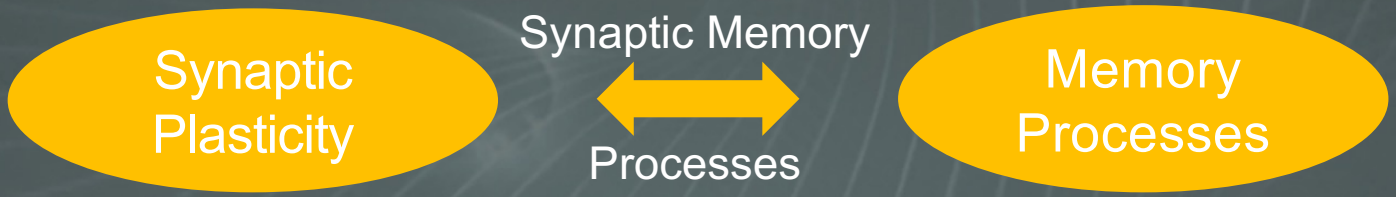
Biological Neural Networks: Sinapse

Learning as an Alteration of Synaptic Physical Properties: changes in the number and/or shape of terminal boutons and variations in dendritic conductivity.

When the axon of Neuron A is sufficiently close to excite Neuron B and participates repeatedly or persistently in the activation of the latter, a growth process or a metabolic change occurs in one or both neurons, increasing the efficiency of A, since it is one of the neurons that activates B (O.D. Hebb, 1949).

Biological Neural Networks: Sinapse

Hebb's proposal remains at a logical-functional level, but it has served as a compass for the analysis of data on synaptic modifications.



Eric R. Kandel (Nobel Prize in Medicine, 2000), using the nervous system of the sea slug Aplysia as an experimental model, demonstrated that changes in synaptic function are central to learning and memory.

Biological Neural Networks: Properties

Dynamic Unsupervised Learning

- Ability to classify and store information
- Information retrieval and error evaluation
- Information processing based on experience

Processing and Storage Viewed as Parallel Processes

- A human sees a lion and runs away (in 0.1 s, 1,000 cycles)
- First love is never forgotten, even after a partial destruction of neurons and synapses

Biological Neural Networks: Properties

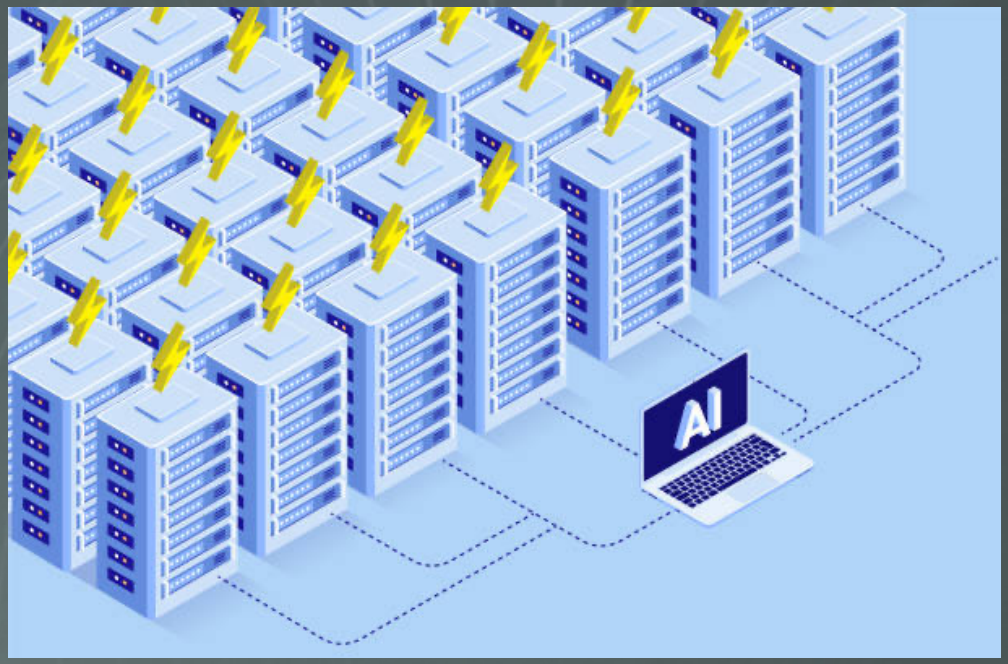
Experience-based Information Processing

Biological Neural Networks: Properties

Extreme redundancy and functional plasticity

Compact size and low weight

Low power consumption



Biological Neural Networks: History

1873: C. Golgi

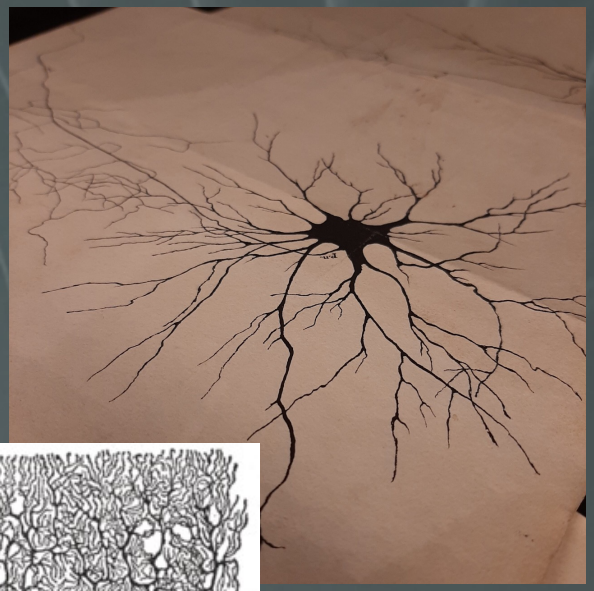
The black reaction, a staining technique using silver nitrate to visualize individual neuron

1888: S. Ramon y Cajal

Demonstrated that the nervous system is composed of discrete units that communicate with each other

1891: H.W. Waldeyer

Introduced the term Neuron

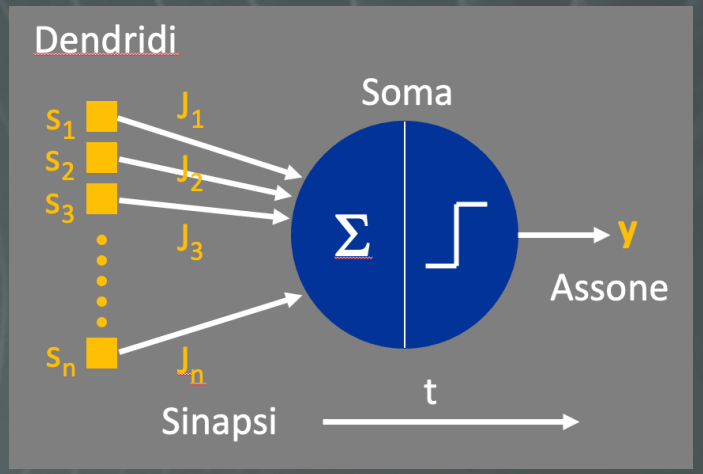


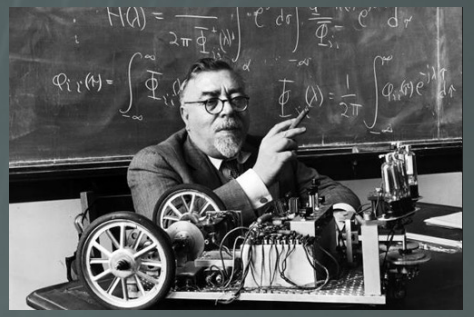
Formal Neural Networks: History

1943: W. McCullogh e W. Pitts
Formal neuron and propositional calculus

1949: D. J. Hebb
Synaptic modifications and memory processes

1950: F. Rosenblatt e N. Wiener
Cybernetics: Study of control and communication mechanisms in living beings and machines

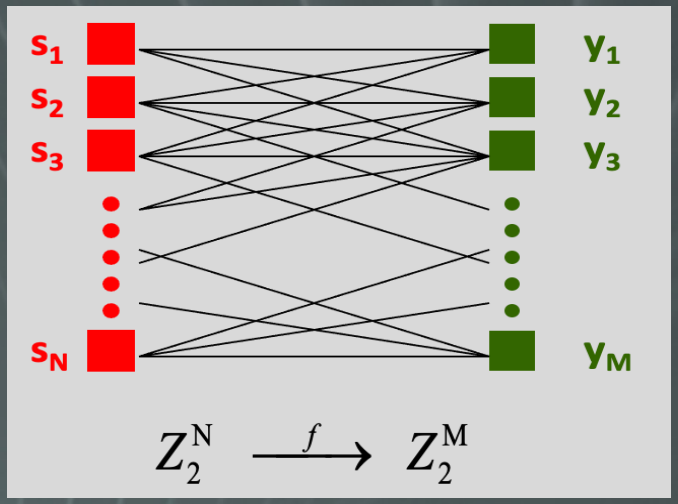




Formal Neural Networks: History

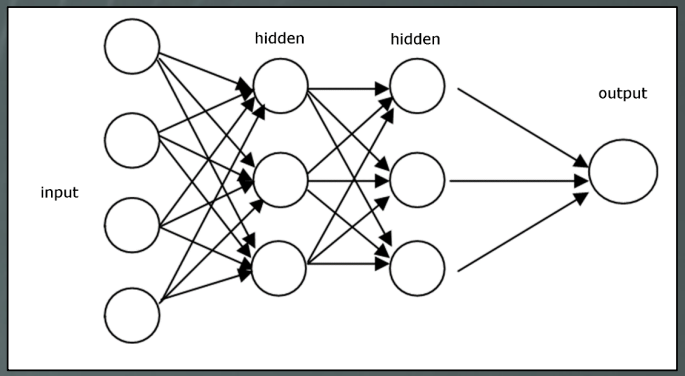
1958: F. Rosenblatt

Generalization of the formal neuron: Perceptron



1973: T. Kohonen e M. Ruohonen

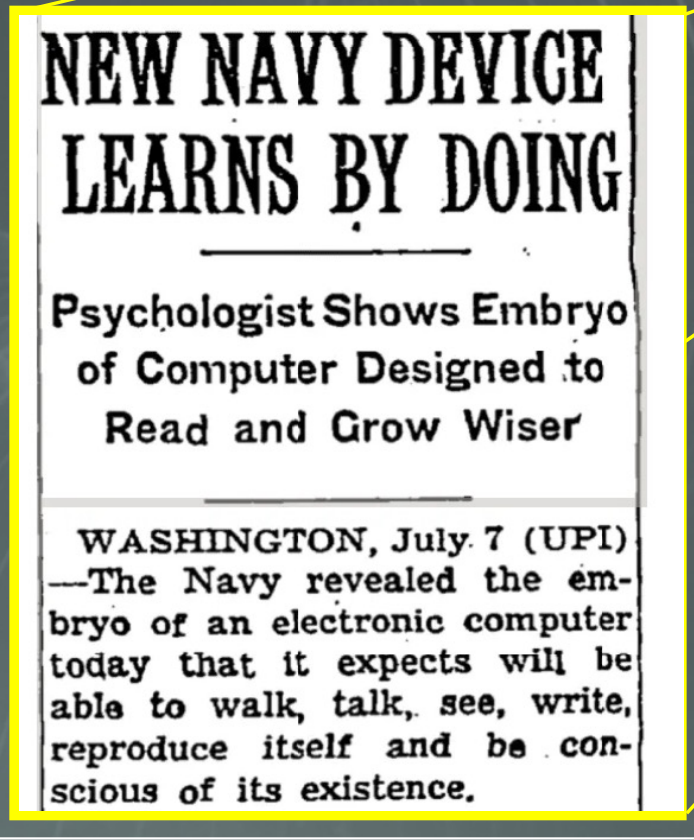
Multiple Perceptron. Deep Feedforward Neural Network



1958: F. Rosenblatt

Generalization of the formal neuron: Perceptron

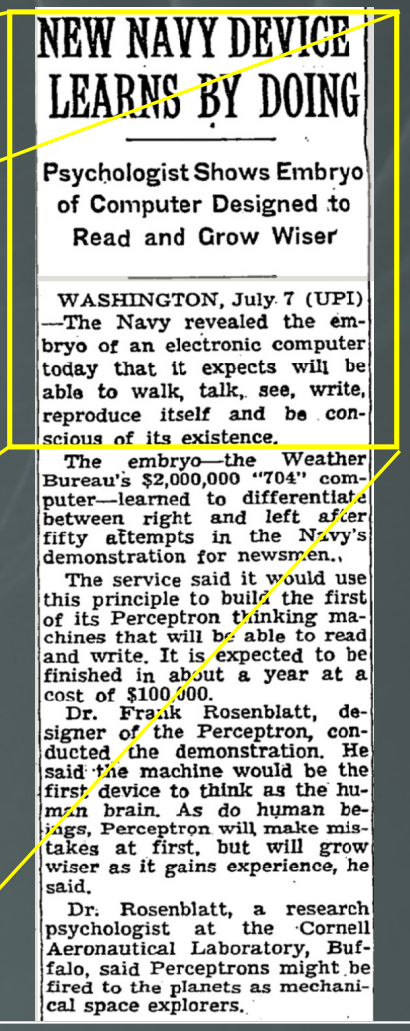
The New York Times
7 July 1958



NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.



NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather Bureau's $2,000,000 "704" computer—learned to differentiate between right and left after fifty attempts in the Navy's demonstration for newsmen.

The service said it would use this principle to build the first of its Perceptron thinking machines that will be able to read and write. It is expected to be finished in about a year at a cost of $100,000.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human beings, Perceptron will make mistakes at first, but will grow wiser as it gains experience, he said.

Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

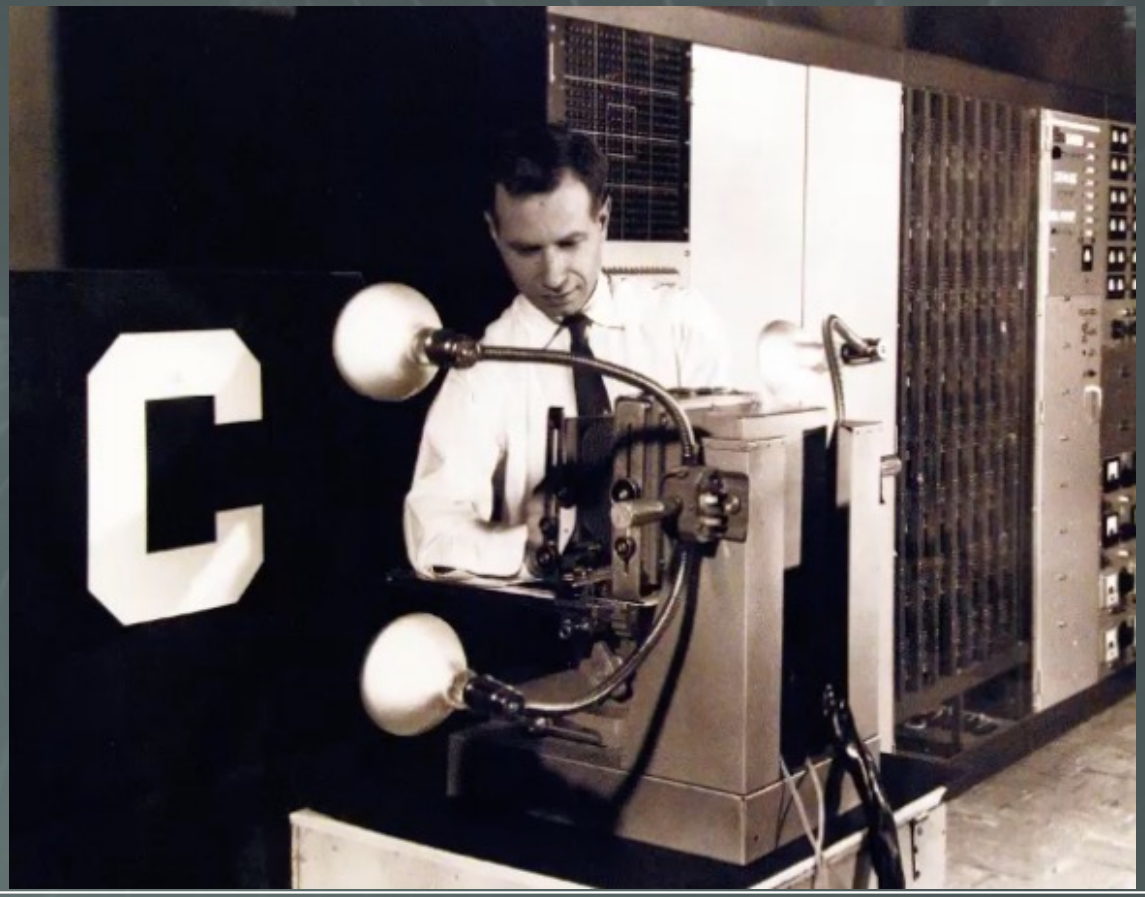
1958: F. Rosenblatt

Generalization of the formal neuron: Perceptron

- 400 photoreceptors
- 512 internal neurons

Each neuron receives input from 40 randomly selected photoreceptors

Eight output neurons, connected to all internal neurons, with weights adjustable via potentiometers



Formal Neural Networks: History

1974: F. Little, J. J. Hopfield e D. J. Amit

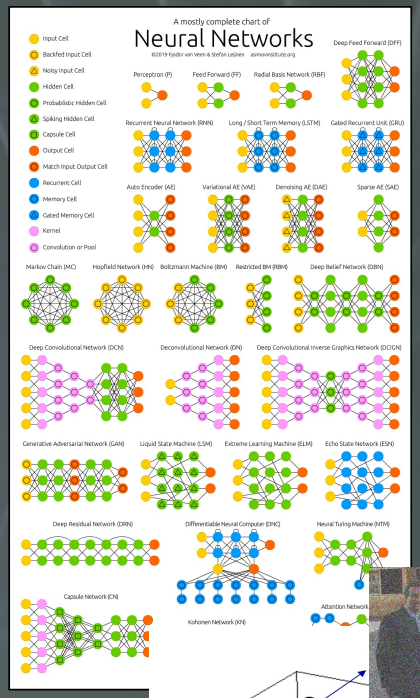
Attractor Neural Networks and associative memory. Dynamical systems and statistical mechanics

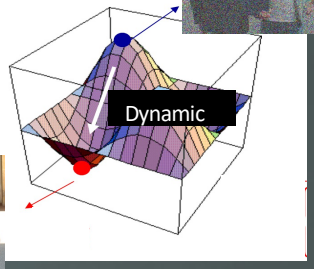
1985: G. E. Hilton

Learning rule for Deep Feedforward Neural Networks: Backpropagation

1990: The Neural Network Zoo

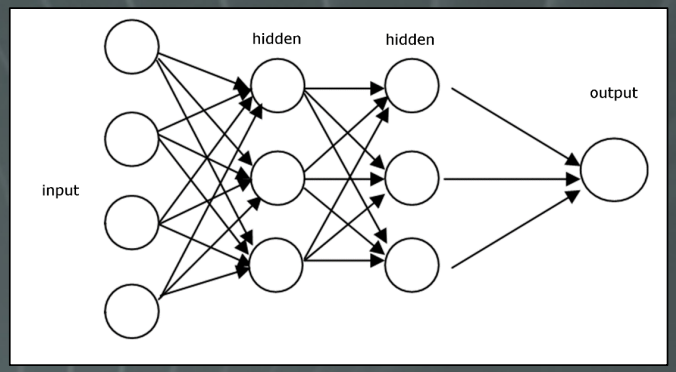
CNN: Convolutional Neural Networks, RNN: Recurrent Neural Networks,Attention Mechanism,LLM: Large Language Mode





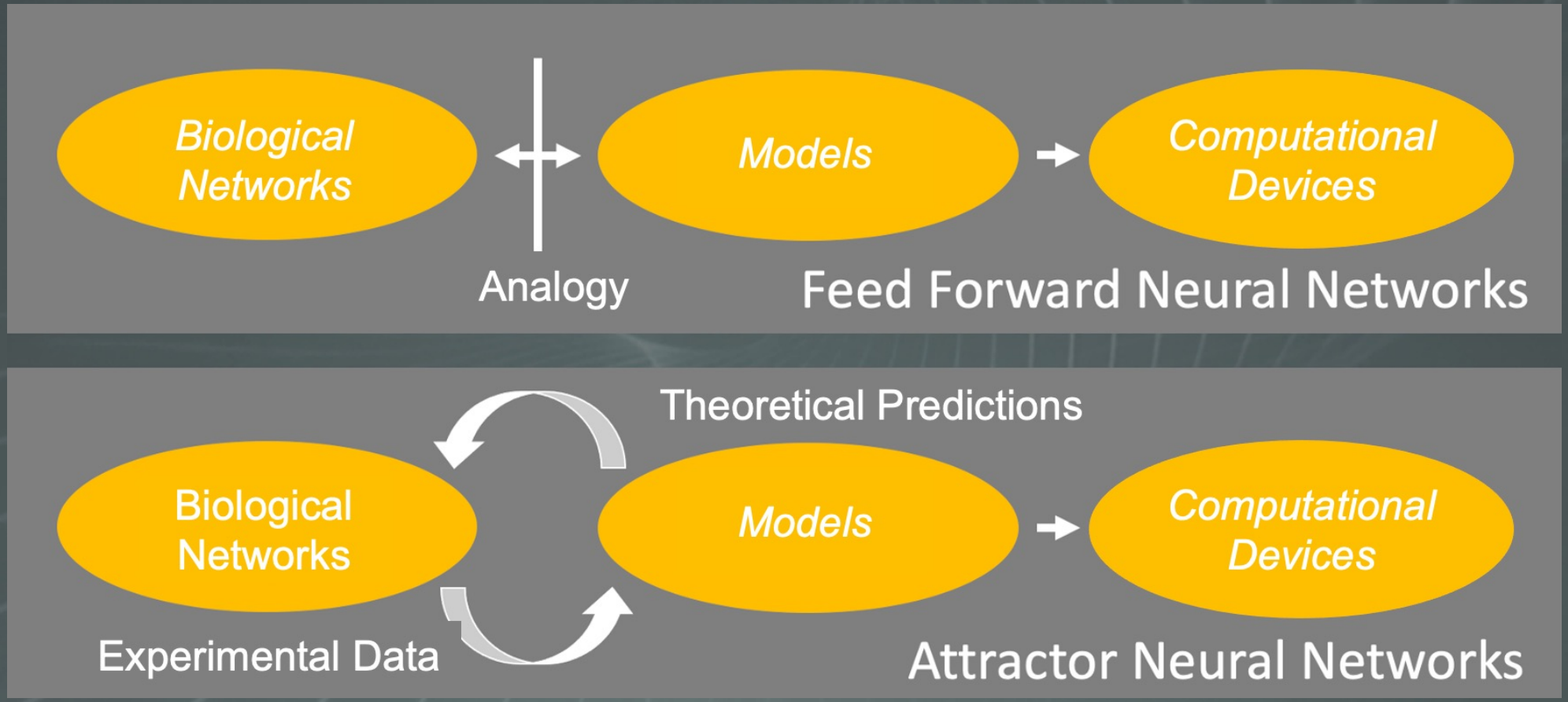
Two Distinct Classes of Artificial Neural Networks

Feedforward Neural Networks have a small number of neurons and low connectivity: self-organizing computational devices



Attractor Neural Networks have a large number of neurons and high connectivity, are considered models of cognitive activity. They are complex dynamical systems, with attractors representing stable states of their dynamics

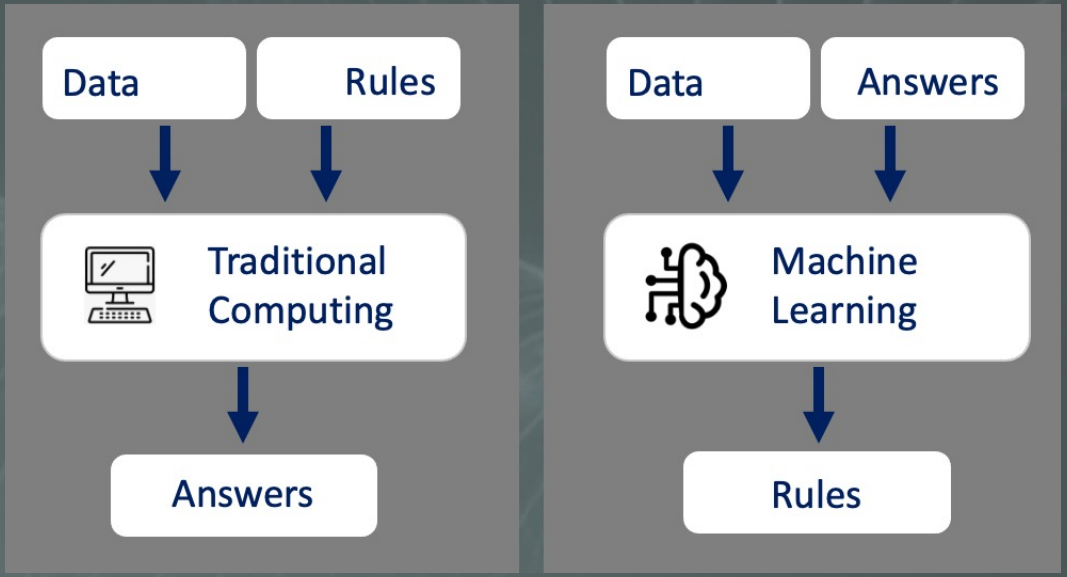
Two Distinct Classes of Artificial Neural Networks



Feed Forward Neural Network: Formal Neuron

Traditionally, a program is written in order to obtain an output

A FFNN model is trained to extract rules from examples

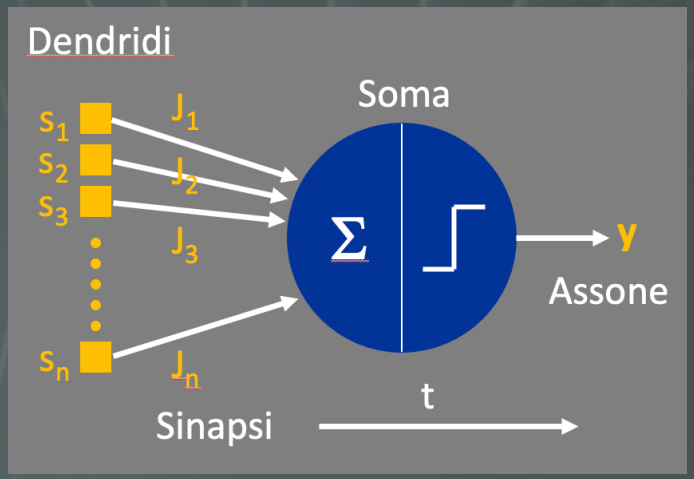


Feed Forward Neural Network: Formal Neuron

$$y = f(\bar{s}, \bar{w}, \theta)$$

$\Sigma = J_1 s_1 + J_2 s_2 + J_3 s_3 + \ldots + J_n s_n$

The soma computes the sum of the contributions of other neurons, weighted by synaptic efficacy



y = 1 f and only if $\Sigma > h$, otherwise *y = 0*

Σ is compared with the threshold *h*.

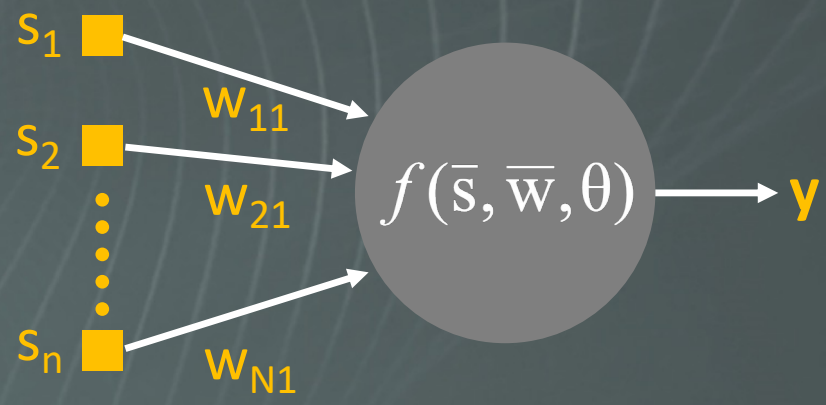
The neuron is active if and only if $\Sigma > h$

$$s_i = (0,1), i = 1,2,3,\ldots,n$$

$$J_i \in \Re, i = 1,2,3,\ldots,n$$

Feed Forward Neural Network: Formal Neuron

How can we choose the synaptic weights and the threshold in such a way as to uniquely determine the desired mapping?



Feed Forward Neural Network: Formal Neuron

Simple Case: *N* = 2. Let us assume:

$$Z_2 \equiv (0,1)$$

$$s_i \in Z_2, i = 1,2; \Rightarrow S^N \equiv Z_2^2$$

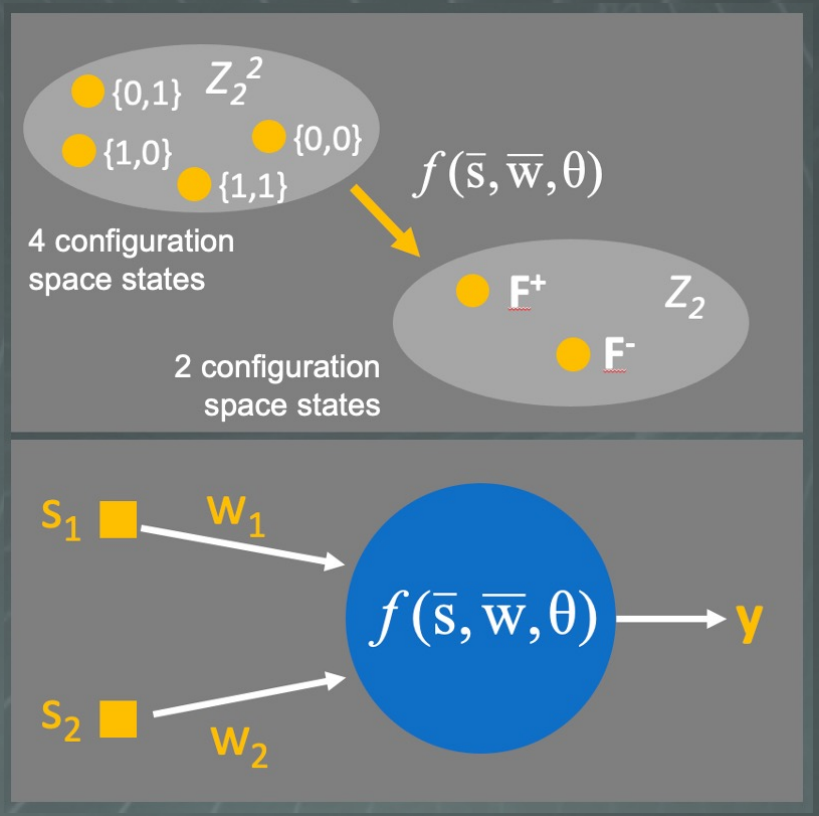
$$y \in Z_2$$

$$w_i \in \Re, i = 1,2; \ \theta \in \Re$$

con

$$f(\bar{s},\bar{w},\theta) = \Theta(s_1 w_1 + s_2 w_2 - \theta)$$

Binary Logic



Heaviside Function

$$Z_2 \equiv (0,1)$$

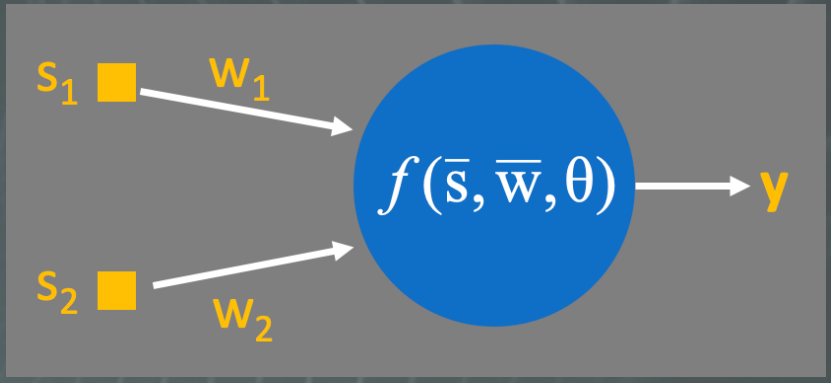
$$s_i \in Z_2, i = 1,2; \Rightarrow S^N \equiv Z_2^2$$

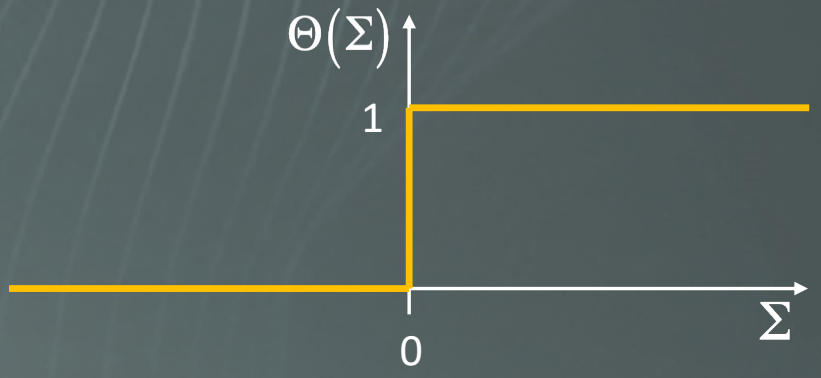
$$f(\bar{s},\bar{w},\theta) = \Theta(s_1 w_1 + s_2 w_2 - \theta)$$

$$\Sigma = s_1 w_1 + s_2 w_2 - \theta$$

$$f(\bar{s},\bar{w},\theta) = \Theta(\Sigma)$$

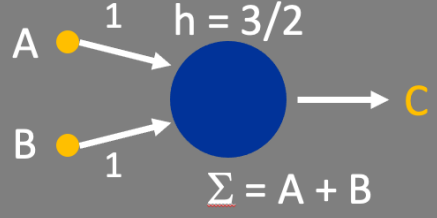
$$\Theta(\Sigma) = \begin{cases} 0 \text{ se } \Sigma \le 0 \\ 1 \text{ se } \Sigma > 0 \end{cases}$$





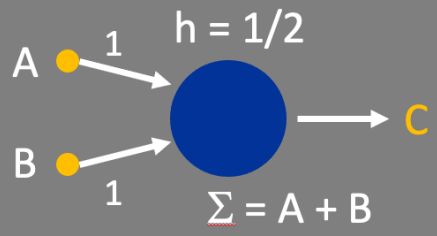
Feed Forward Neural Network: Formal Neuron

Networks of formal neurons capable of performing any propositional calculus operation



C	A	B
0	0	0
0	0	1
0	1	0
1	1	1

AND



C	A	B
0	0	0
1	0	1
1	1	0
1	1	1

OR

Formal Neuron: AND Logic

Let's start from the AND truth table

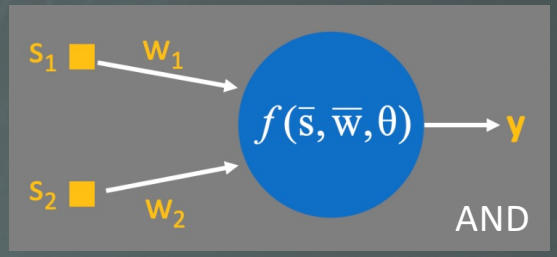
s_1	s_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Evaluate Σ for the four input states

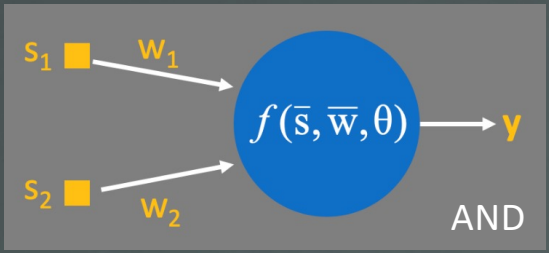
$$\Sigma = s_1 w_1 + s_2 w_2 - \theta$$

Use the Heaviside function.

$$f(\bar{s}, \bar{w}, \theta) = \Theta(s_1 w_1 + s_2 w_2 - \theta)$$



Formal Neuron: AND Logic



$$\Sigma = s_1 w_1 + s_2 w_2 - \theta$$

s_1	s_2	y
0	0	0
0	1	0
1	0	0
1	1	1

$s_1 = 0; s_2 = 0 \rightarrow \Sigma = -\theta$

$s_1 = 0; s_2 = 1 \rightarrow \Sigma = w_2 - \theta$

$s_1 = 1; s_2 = 0 \rightarrow \Sigma = w_1 - \theta$

$s_1 = 1; s_2 = 1 \rightarrow \Sigma = w_1 + w_2 - \theta$

Formal Neuron: AND Logic

$$\Theta(\Sigma) = \begin{cases} 0 \text{ se } \Sigma \leq 0 \\ 1 \text{ se } \Sigma > 0 \end{cases}$$

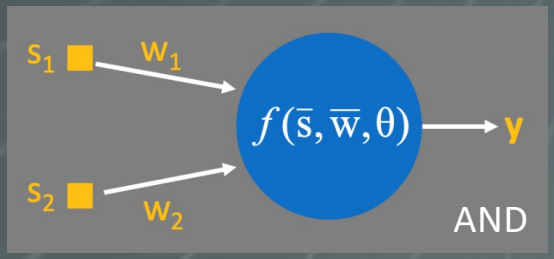
00 → 0; $\Sigma = -\theta$ → $-\theta \leq 0$

01 → 0; $\Sigma = w_2 - \theta$ → $w_2 - \theta \leq 0$

10 → 0; $\Sigma = w_1 - \theta$ → $w_1 - \theta \leq 0$

11 → 1; $\Sigma = w_1 + w_2 - \theta$ → $w_1 + w_2 - \theta > 0$

s_1	s_2	y
0	0	0
0	1	0
1	0	0
1	1	1



Solution

$$w_1^{AND} = w_2^{AND} = 1$$

$$\theta^{AND} = 3/2$$

Feed Forward Neural Network: Multi-Layer Perceptron

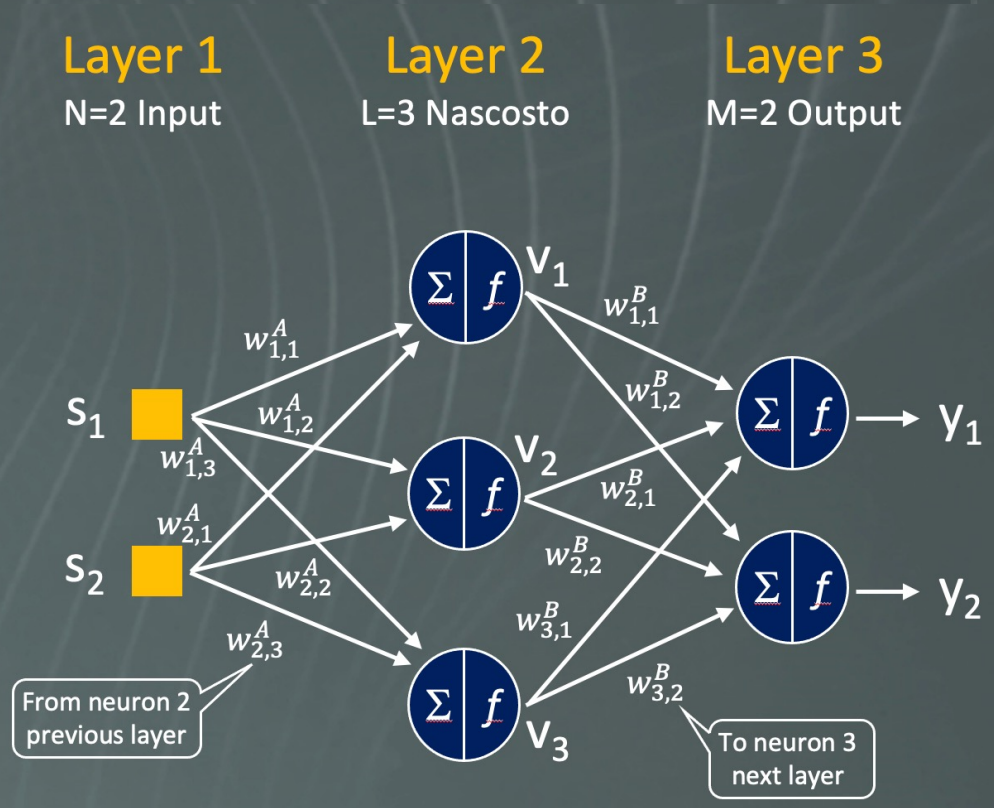
The Multi-Layer Perceptron is a special case of a FFNN with nonlinear activation functions in all layers

Forward pass:

$$v_i = f\left(\sum_{j=1}^{N} w_{i,j}^{A} s_i\right); \quad y_k = f\left(\sum_{l=1}^{L} w_{k,l}^{B} v_l\right)$$

$$y_k = f\left(\sum_{l=1}^{L} w_{k,l}^{B}\, f\left(\sum_{j=1}^{N} w_{l,j}^{A} s_i\right)\right)$$

$f(\,)$ is the activation function



Feed Forward Neural Network: Learning

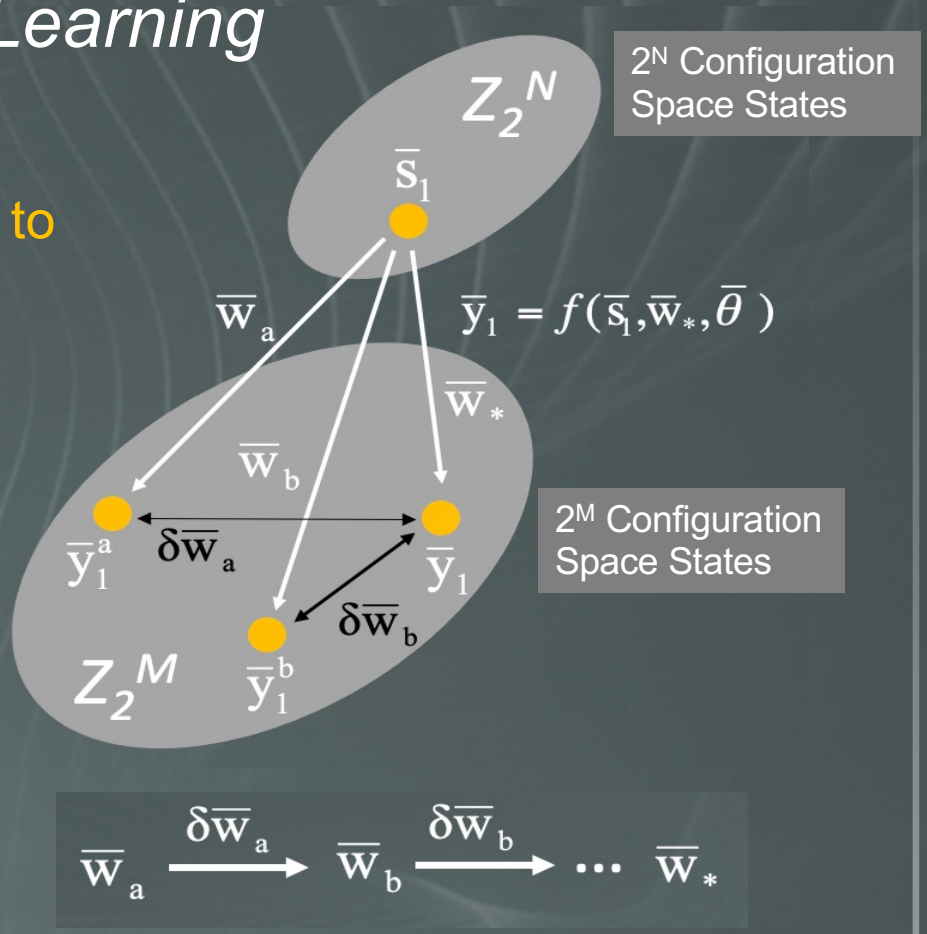
Iterative Algorithm

Iterative process in the weight space, aiming to minimize the quadratic function:

$$D = \frac{1}{2}\sum_{\mu}\sum_{i}\left(y_i^{\mu} - Y_i^{\mu}\right)^2 \Rightarrow D\left(\{w_{ij}\}\right)$$

i.e.

$$D\left(\{w_{ij}\}\right) \xrightarrow{\{w_{ij}\}\to\{w_{ij}^{*}\}} 0$$



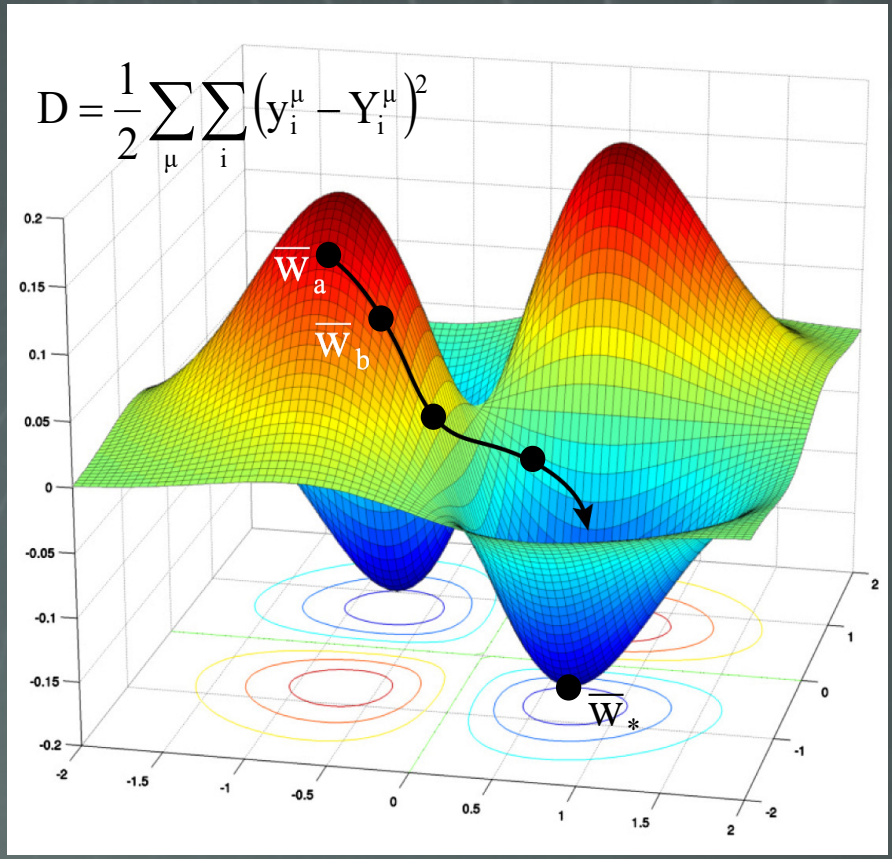
Feed Forward Neural Network: Learning

Iterative Algorithm

Move along the path of steepest descent: Gradient-Based Update

$$\overline{w}_a \xrightarrow{\delta\overline{w}_a} \overline{w}_b \xrightarrow{\delta\overline{w}_b} \cdots \overline{w}_*$$

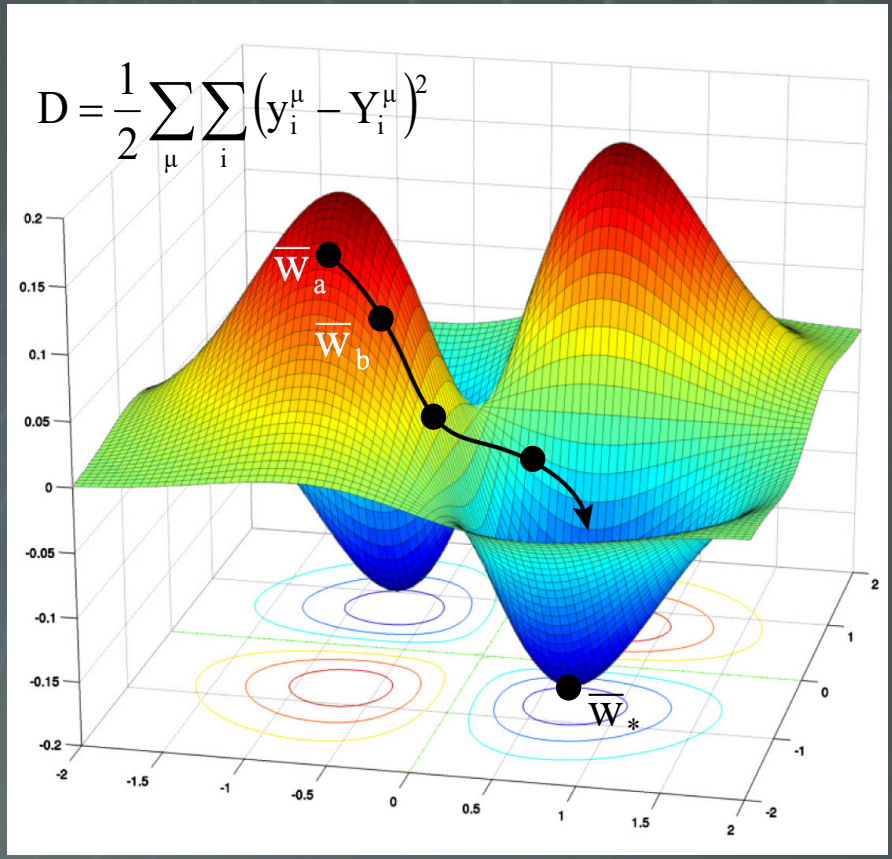
$$D(\overline{w}_a) > D(\overline{w}_b) \ldots > 0$$



Feed Forward Neural Network: Back Propagation

Backpropagation Algorithm: Gradient-based algorithm used for training in Deep Learning (DL)

Reduces the cost function and computational complexity of the problem



Feed Forward Neural Network: Back Propagation

Backpropagation involves two phases:

Forward Phase:

Present an input example

Compute the output

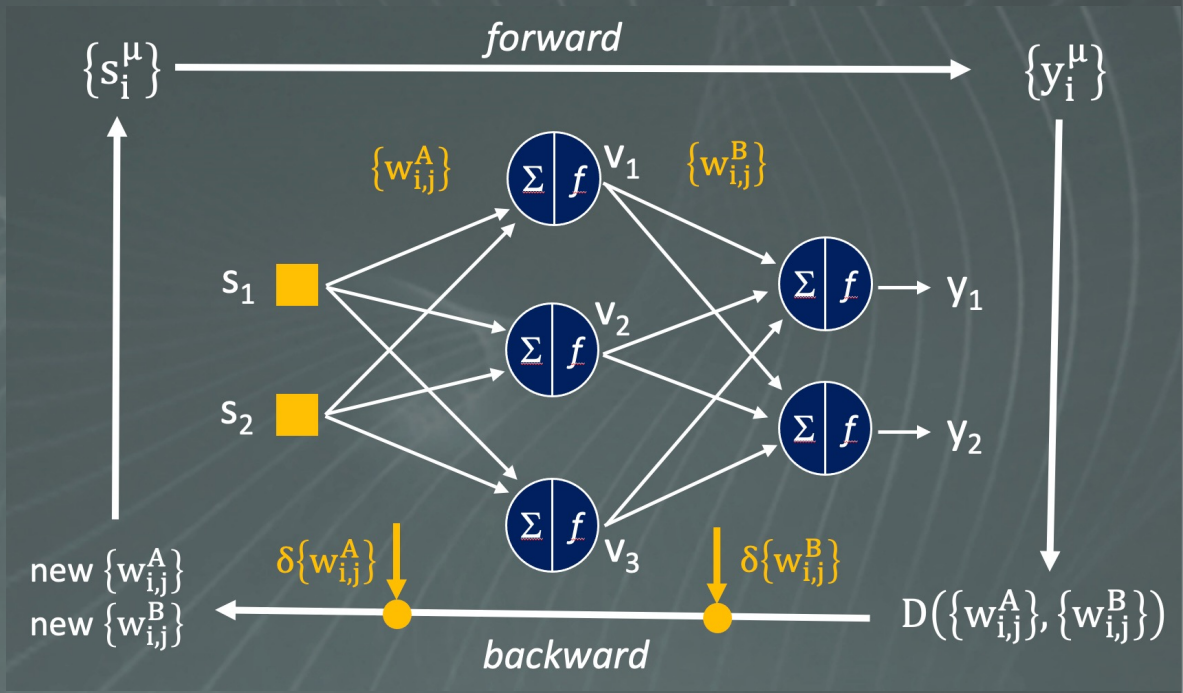
Calculate the error

Backward Phase:

Propagate the error backward

Adjust the weights of each layer

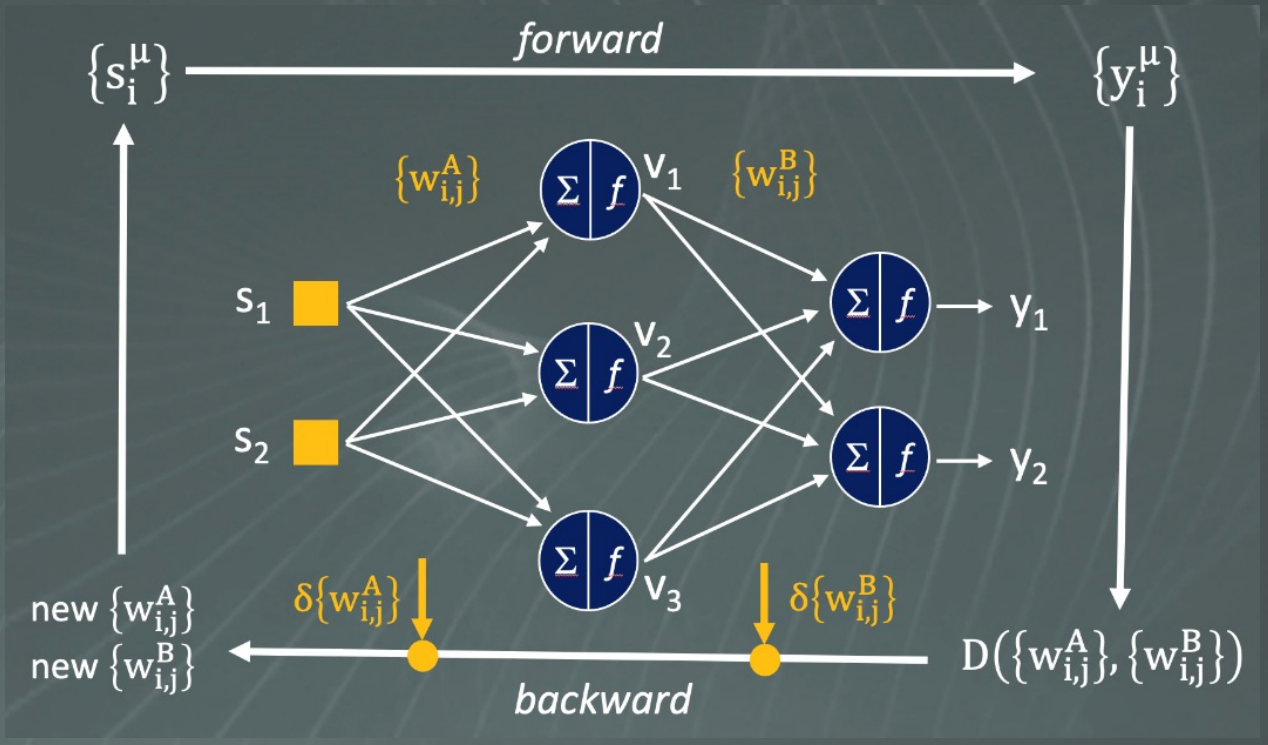
Update the network's weights



Feed Forward Neural Network: Back Propagation

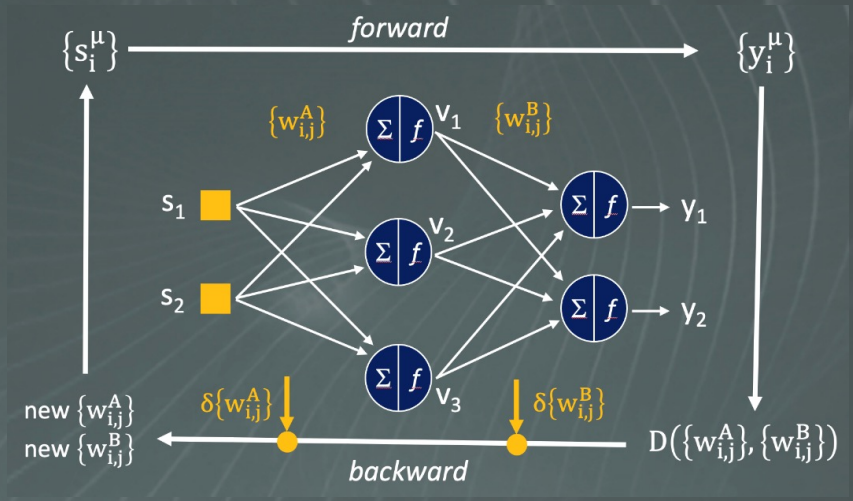
Minimize D by adjusting the weights $\{w_{i,j}^{B}\}$ while keeping $\{w_{i,j}^{A}\}$ fixed

Once the new weights $\{w_{i,j}^{B}\}$ are determined, adjust the weights $\{w_{i,j}^{A}\}$



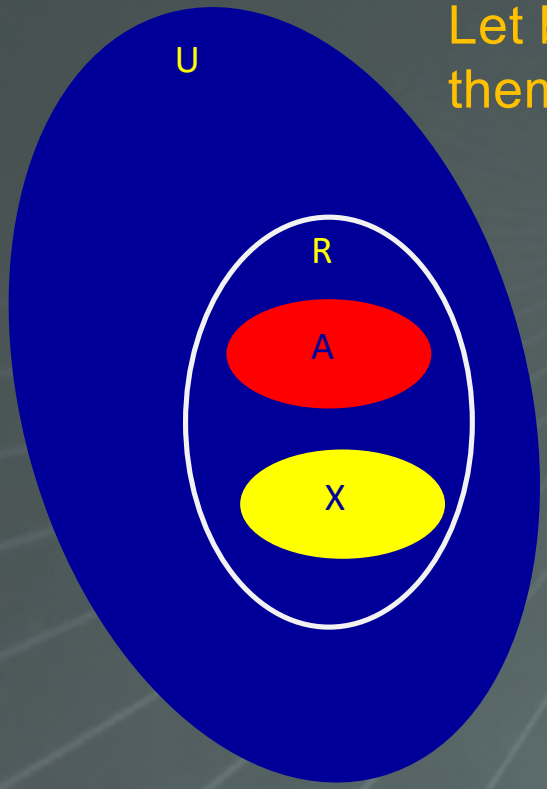
Feed Forward Neural Network: Back Propagation

G. E. Hilton



Prize motivation: "for foundational discoveries and inventions that enable machine learning with artificial neural networks"

Feed Forward Neural Network: Learning & Generalization



Let be U the set of all possible input-output rules, some of them are compatible with a given rule R

Let be A the set used for the learning (p examples)

Let be X the set used for the model validation.

A e X are random choosen and are representative of R

The network has *learned* A and has *no knowledge* of X or R

Feed Forward Neural Network: Learning & Generalization

Finite *p* Training Examples → Correct Generalization

N input neurons

↓

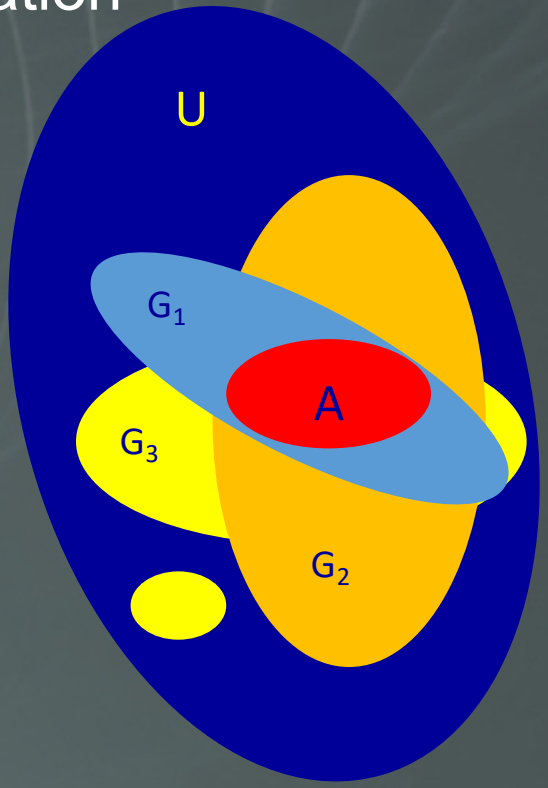
2^N input patterns

↓

2^{2^N} possible functions → *p* examples in A

↓

2^{2^N-p} compatible generalizations



All 2^{2^N-p} possible generalizations are valid !

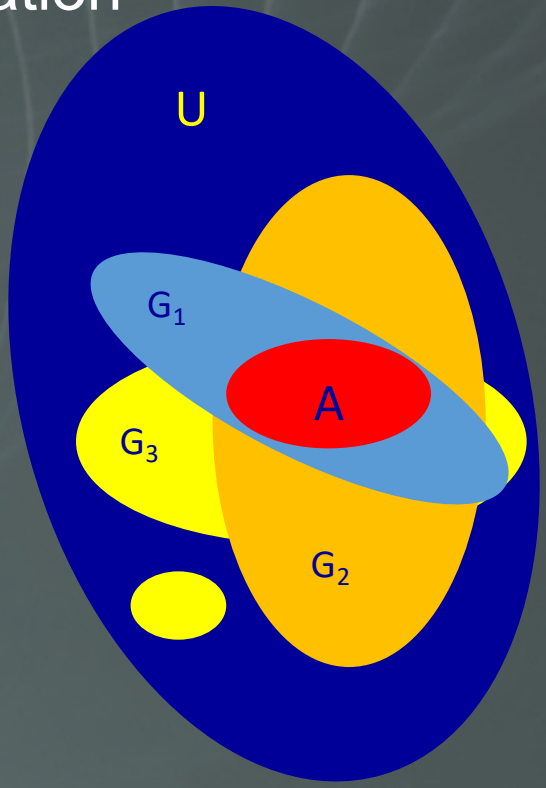
Feed Forward Neural Network: Learning & Generalization

Finite p Training Examples → Correct Generalization

Generalization in Practice

- Theory predicts a critical number of examples $p_* \ll 2^N$
- With p_*, correct generalization is possible
- In practice, when choosing p
  - We do not know whether $p < p_*$ or $p \geq p_*$
  - We do not know if the examples are informative enough

Therefore, generalization is inferred from empirical indicators and inductive bias rather than guaranteed; no recipe exists that ensures correct generalization from a given set of examples



Attractor Neural Network: Complex Dynamical Systems

Large number of neurons and sinapses, with a complex synaptic connectivity topology

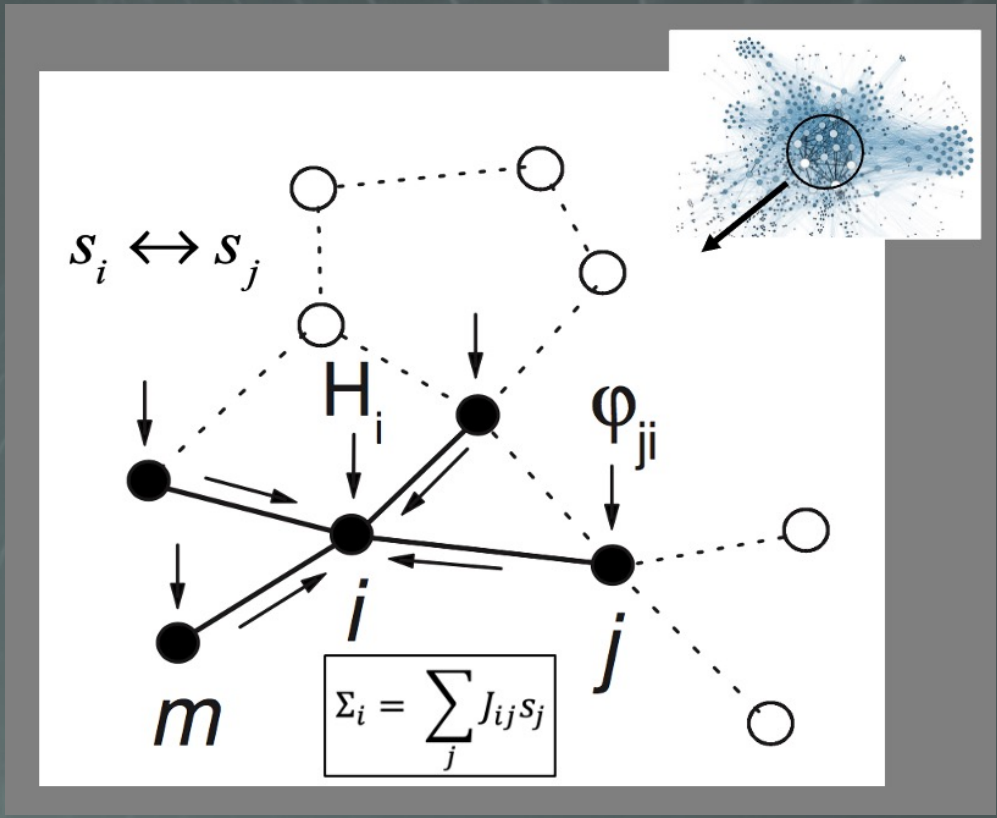
Dual dynamics

Neurons

$$s_i = F(\Sigma_i + H_i - h_i)$$

Sinapses

$$\dot{J}_{ij} = -aJ_{ij} + f(s_i s_j)$$



Attractor Neural Network: Learning

Changes in synaptic weights modify the energy function landscape, creating (or destroying) new minima or reinforcing existing ones

Sinapses

$$\dot{J}_{ij} = -aJ_{ij} + f(s_i s_j)$$

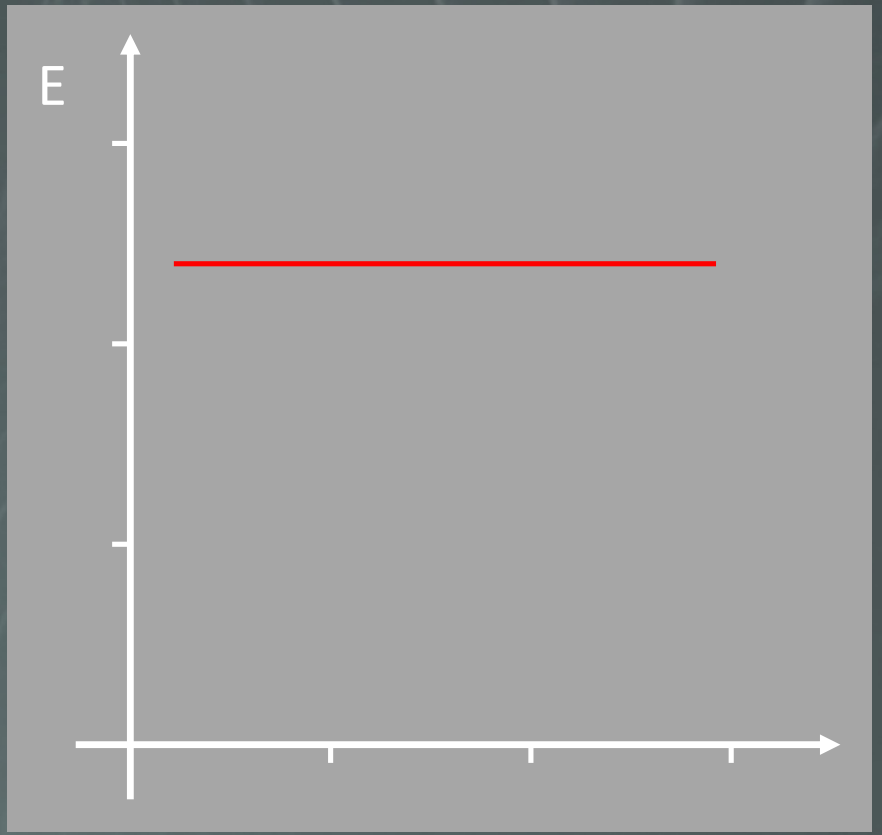
Attractor Neural Network: *Learning*

The network is trained to learn the following classes:

A = (𝗔, A, 𝒜, ... , A)

B = (𝗕, B, ℬ, ... , B)

C = (𝗖, C, 𝒞, ... , C)



Attractor Neural Network: *Learning*

The network is trained to learn the following classes:

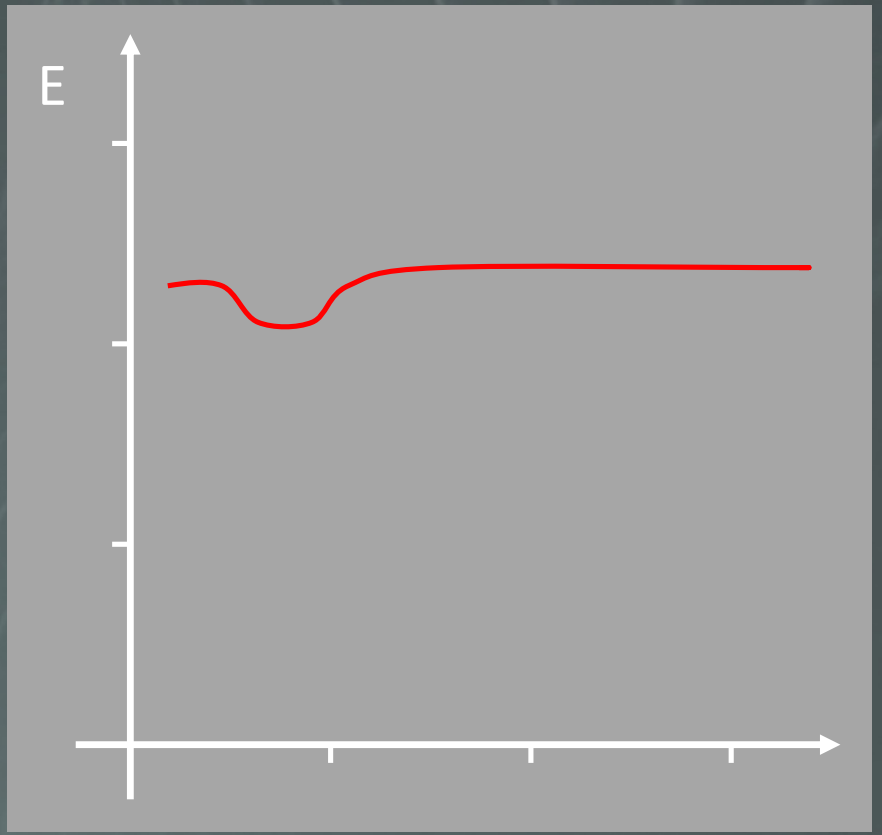
A = (𝗔, A, 𝒜, ... , A)

B = (𝗕, B, ℬ, ... , B)

C = (𝗖, C, 𝒞, ... , C)

Presenting the letter: A

P = 1



Attractor Neural Network: *Learning*

The network is trained to learn the following classes:

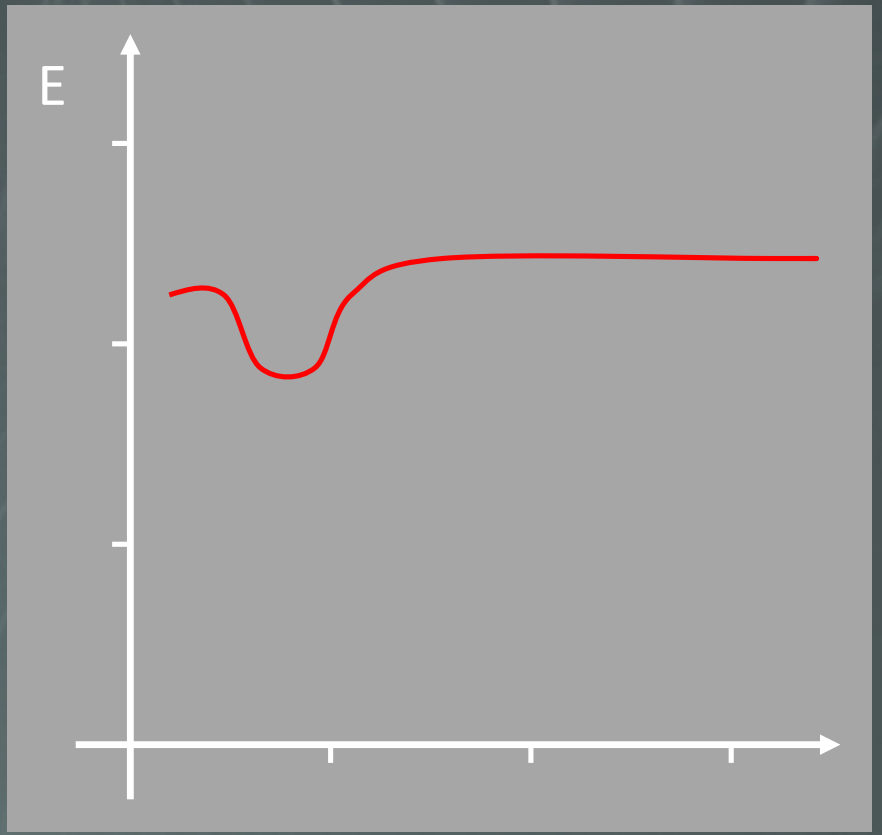
A = (𝗔, A, 𝒜, ... , A)

B = (𝗕, B, ℬ, ... , B)

C = (𝗖, C, 𝒞, ... , C)

Presenting the letter: A

P = 2



Attractor Neural Network: *Learning*

The network is trained to learn the following classes:

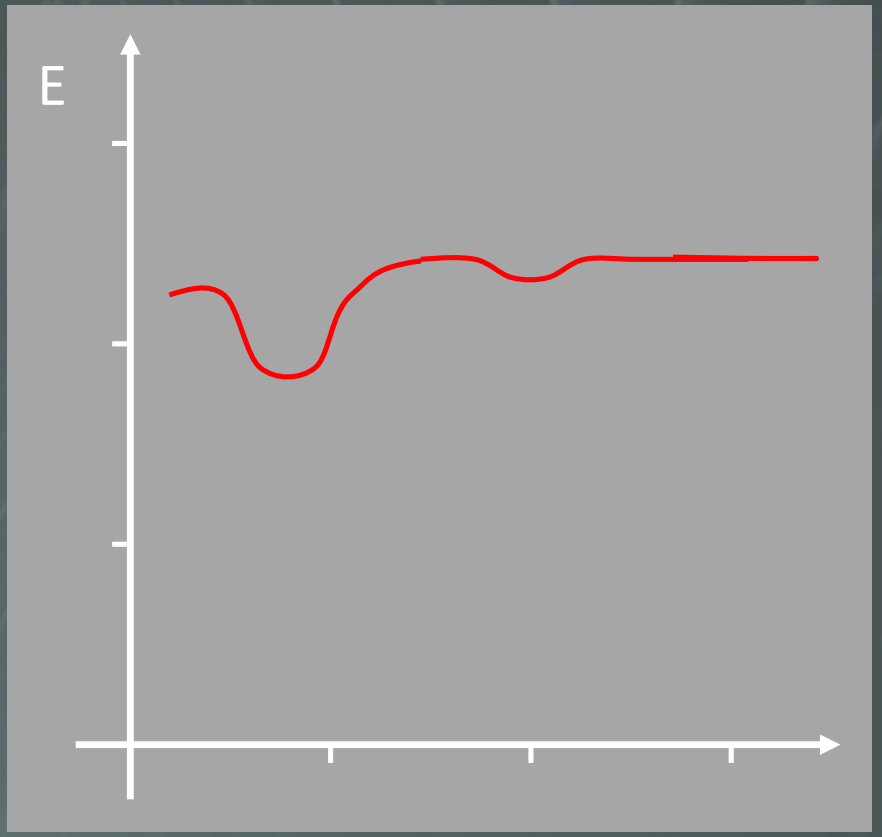
A = (𝐀, A, $\mathcal{A}$, ... , A)

B = (𝐁, B, $\mathcal{B}$, ... , B)

C = (𝐂, C, $\mathcal{C}$, ... , C)

Presenting the letter: $\mathcal{B}$

P = 3



Attractor Neural Network: *Learning*

The network is trained to learn the following classes:

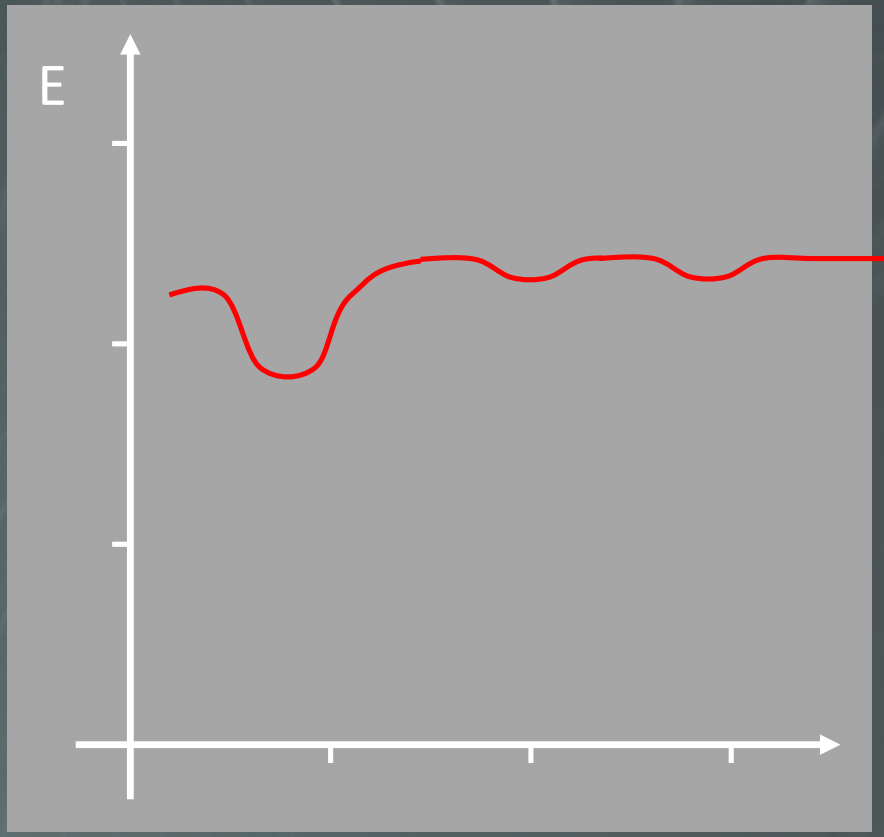
A = (𝗔, 𝐴, 𝒜, ... , A)

B = (𝗕, 𝐵, ℬ, ... , B)

C = (𝗖, 𝐶, 𝒞, ... , C)

Presenting the letter: 𝐂

P = 4



Attractor Neural Network:
Learning

The network is trained to learn the following classes:

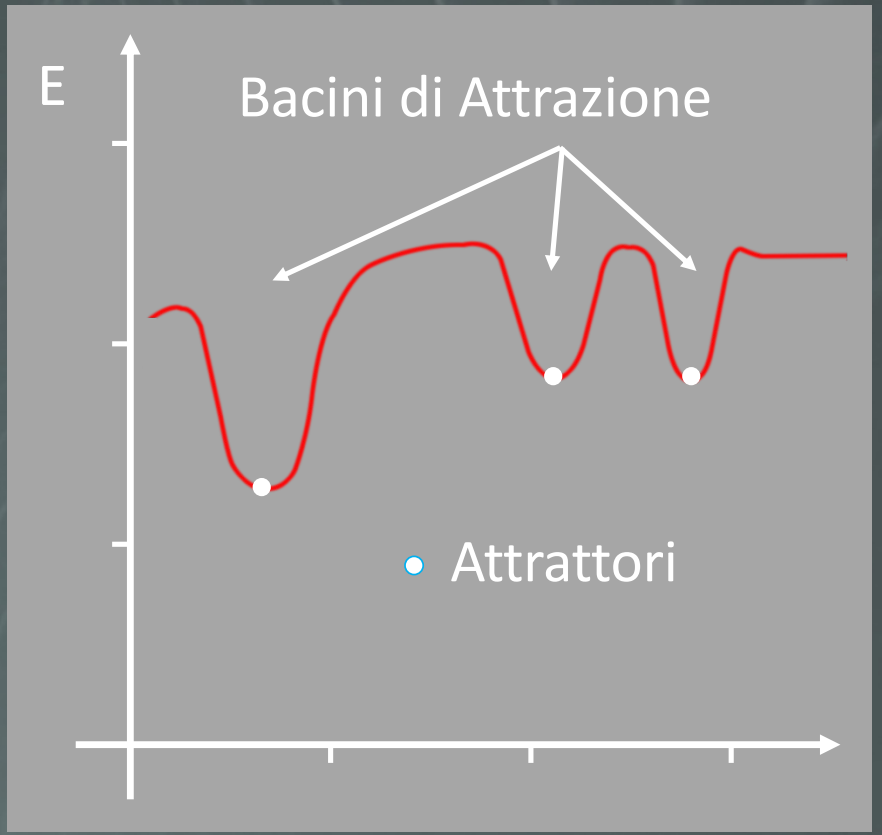
A = (𝗔, A, 𝒜, ... , A)

B = (𝗕, B, ℬ, ... , B)

C = (𝗖, C, 𝒞, ... , C)

Presenting the letter: B

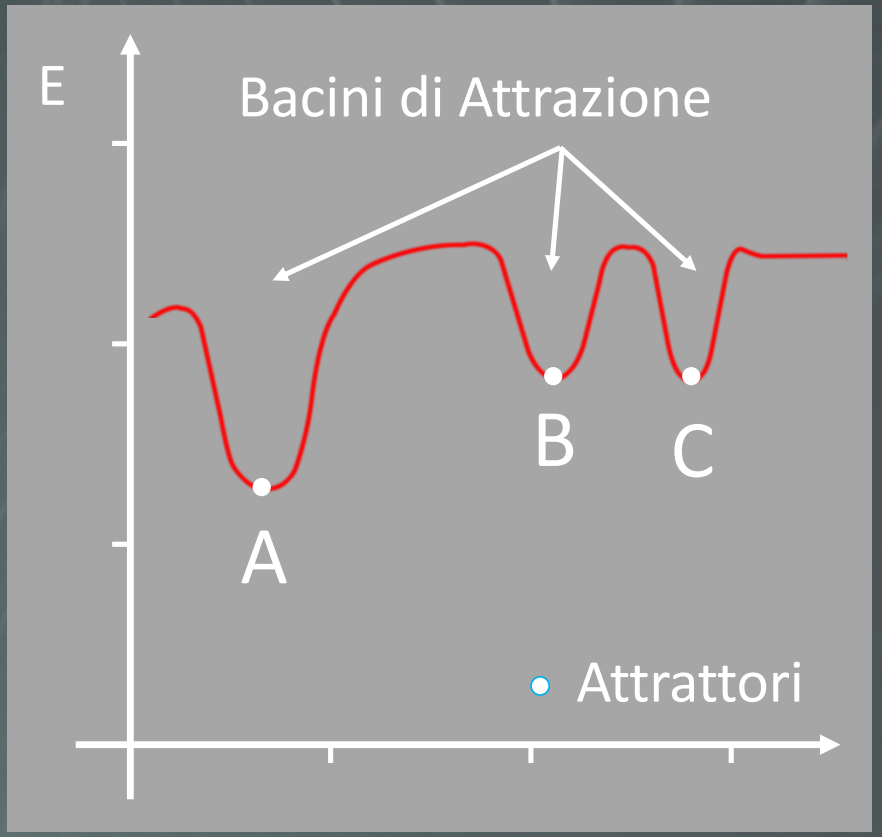
P = 1000



Attractor Neural Network: *Learning*

Unsupervised Learning

The network's state at the attractors represents the prototype of the class

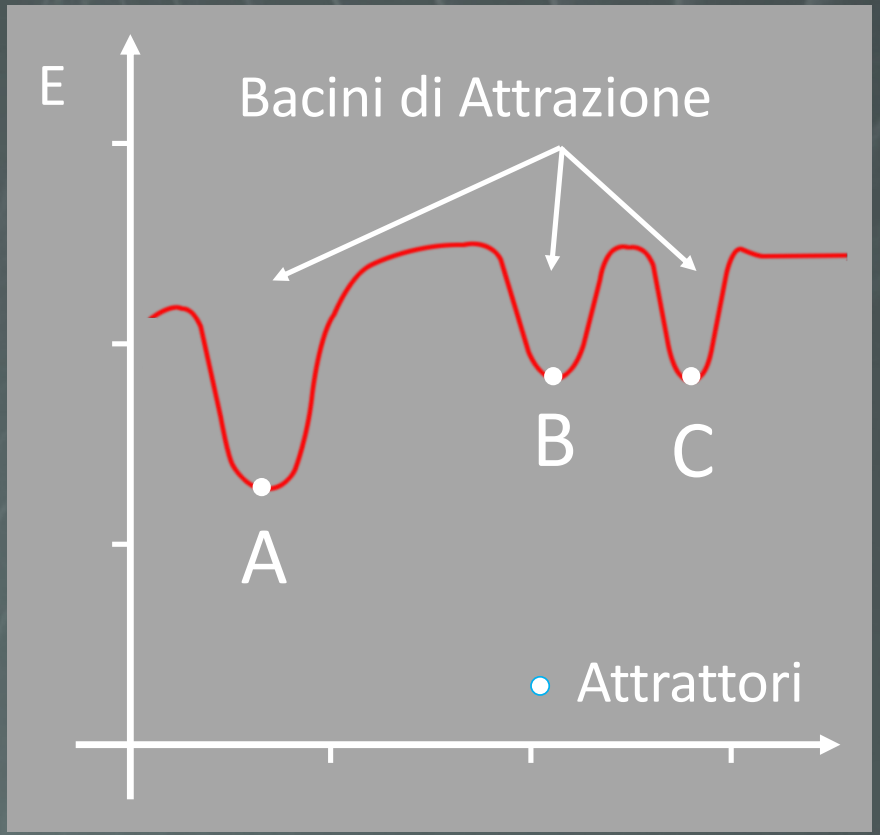


Attractor Neural Network: *Learning*

Unsupervised Learning

Presenting a new class element creates a new attractor.

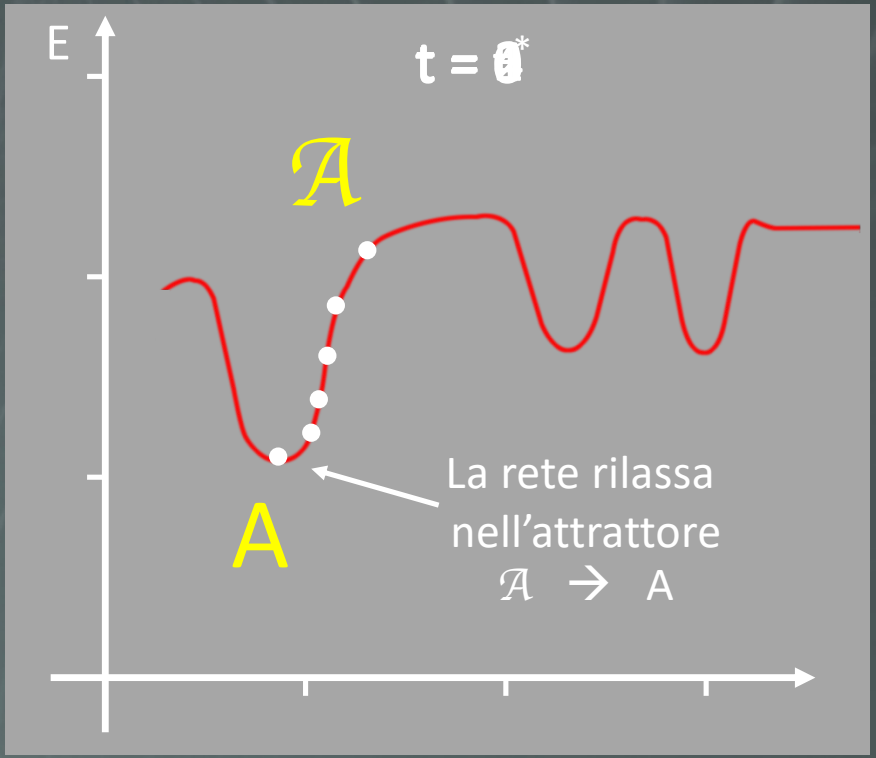
If elements of a class are no longer presented, the network forgets by removing the corresponding attractor.



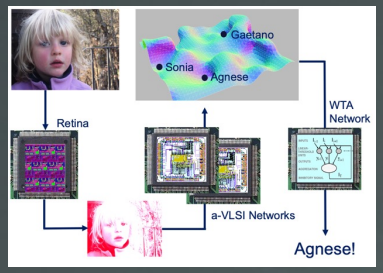
Attractor Neural Network:
Memory retrieval

Class Recognition

- Present to the network: $\mathcal{A}$
- Dynamic relaxation toward a local energy minimum
- Learning and recall are not separate processes



Attractor Neural Network: Memory retrieval



Learning and classification processes in real time and biologically plausible

Unsupervised Learning

Supervised Learning

Name Retrieval

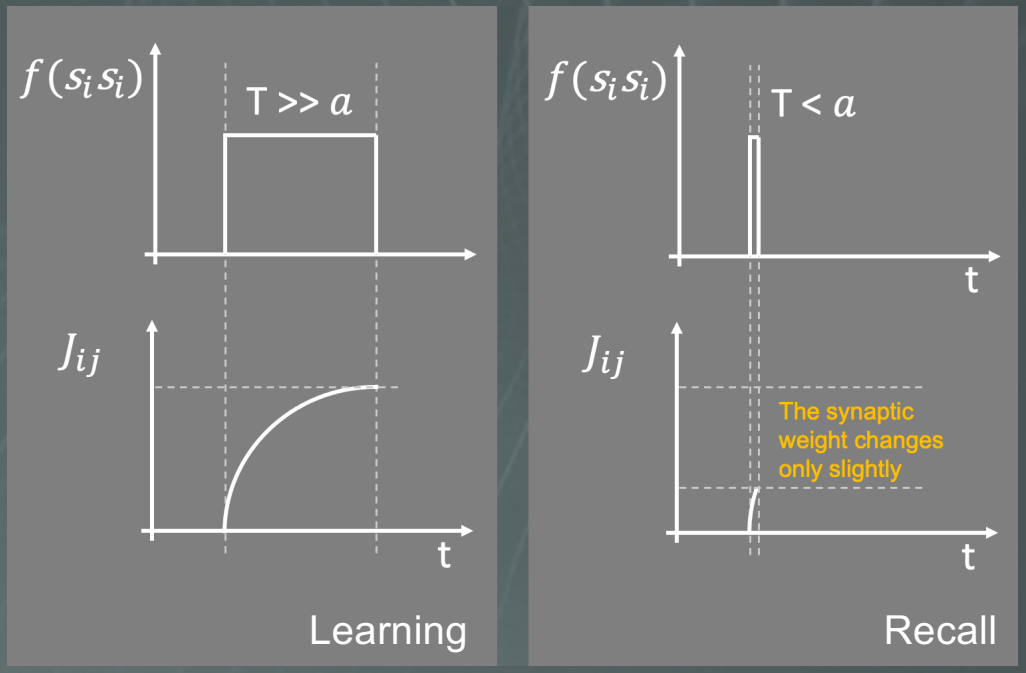
→ Agnese

Attractor Neural Network: Synaptic dynamics

Learning and recall are not separate processes

Synaptic weight dynamics

$$\dot{J}_{ij} = -aJ_{ij} + f(s_i s_j)$$



Associative memory and Hopfield Model

Associative Memory: Definition

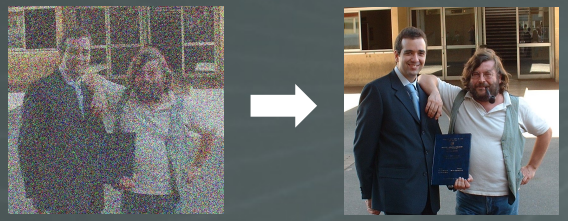
Associative memory is also known as content-addressable memory (CAM), associative storage, or associative array.

It is a type of memory designed to search through data, instead of providing direct access via addresses.

Associative Memory: Definition

Associative memory can be classified into two types:

- Auto-associative: recall patterns from partial or noisy inputs



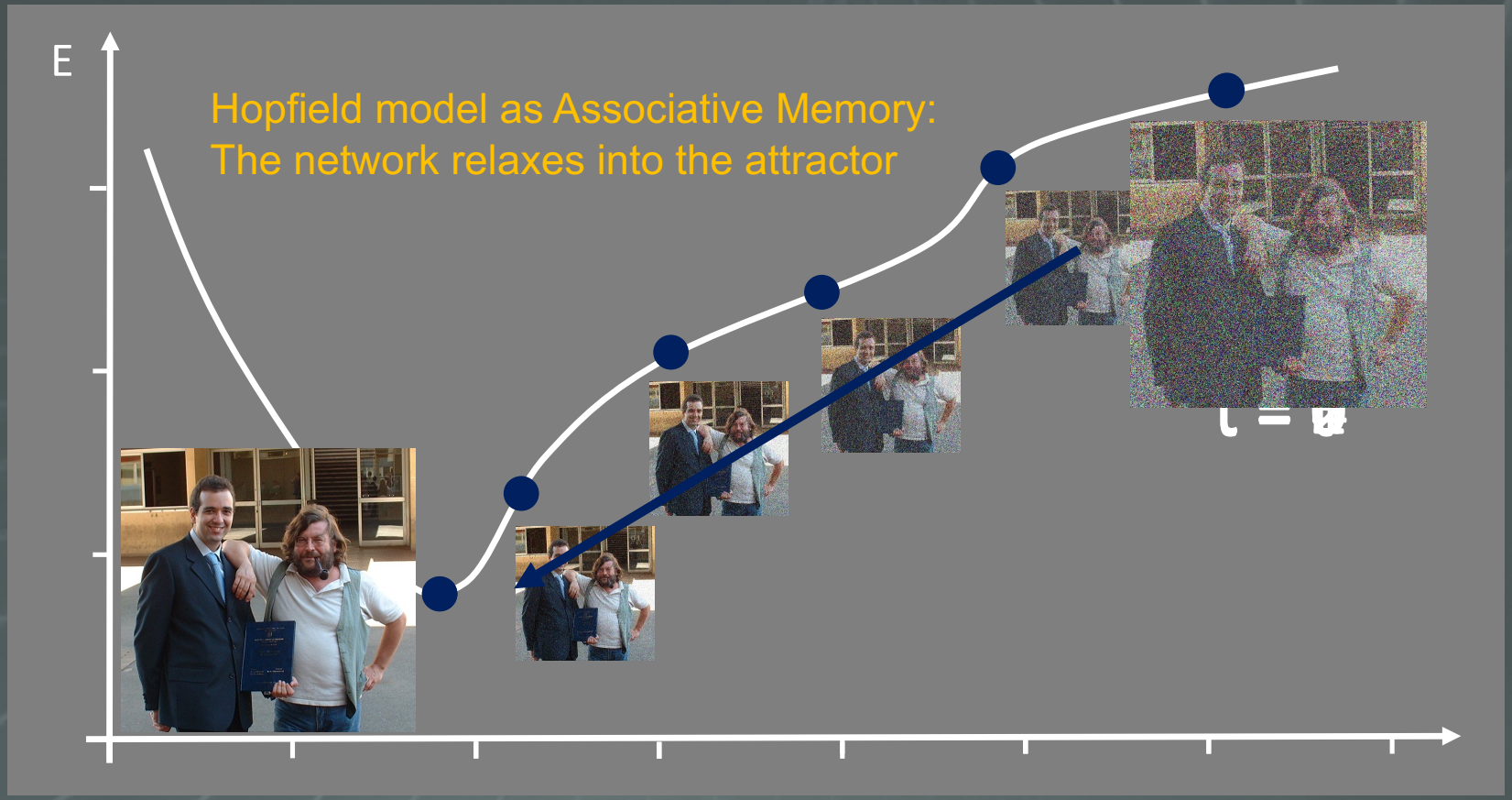
- Hetero-associative: associate one set of patterns with another

Photo of Andrea's graduation

Associative Memory: Definition

The retrieval of information is triggered by partial knowledge of the information

Associative Memory



Attractor Neural Network: Complex Dynamical Systems

Large number of neurons and sinapses, with a complex synaptic connectivity topology

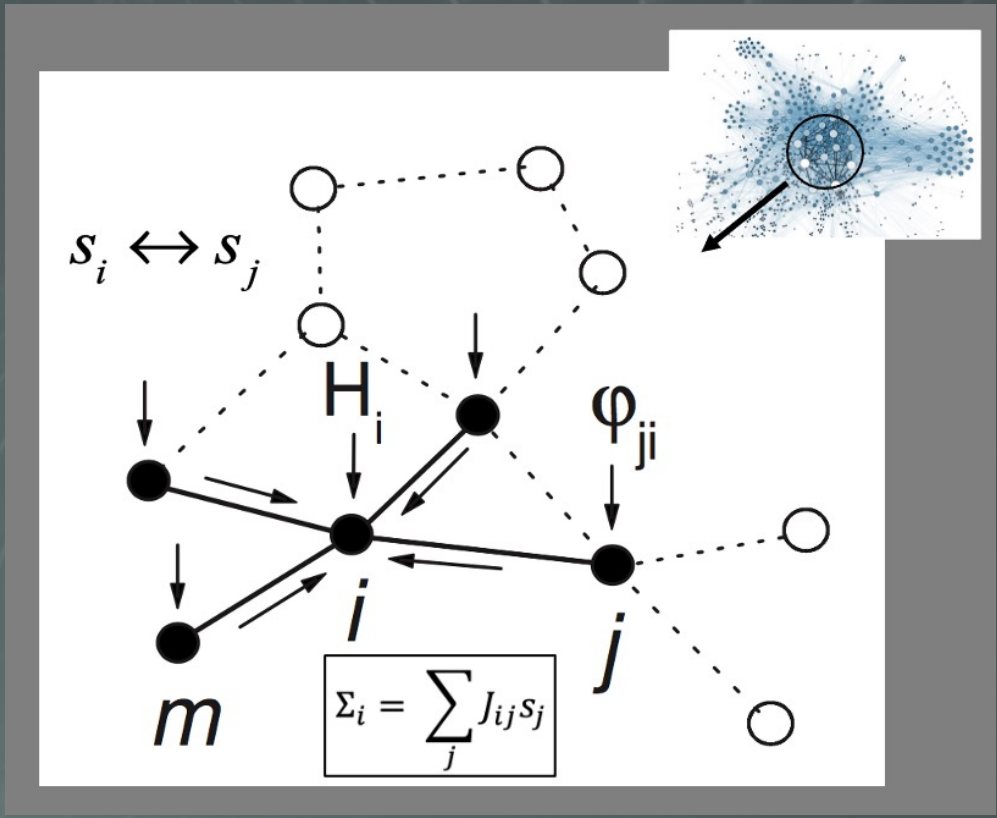
Dual dynamics

Neurons

$$s_i = F(\Sigma_i + H_i - h_i)$$

Sinapses

$$\dot{J}_{ij} = -aJ_{ij} + f(s_i s_j)$$



Hopfield Model: Synaptic Network

Let assume a system with N neurons

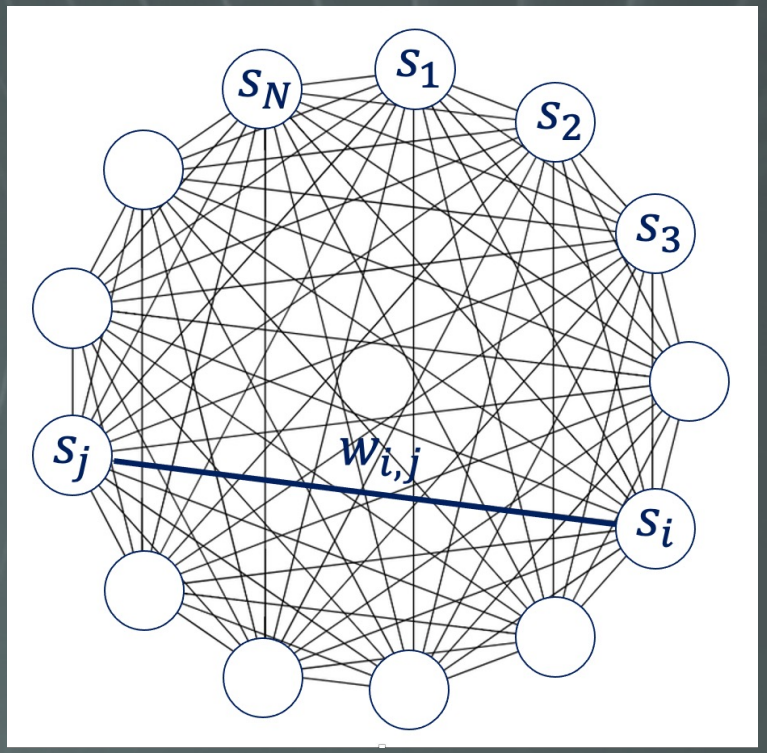
$s_i; i = 1,2,3, \dots, N;\ N \gg 1$

and a synaptic network that is fully connected and symmetric.

$w_{i,j} = w_{j,i}$

$w_{i,i} = 0$

$i, j = 1,2,3, \dots, N$



Hopfield Model: Dynamics

The dynamics of the neurons are given by

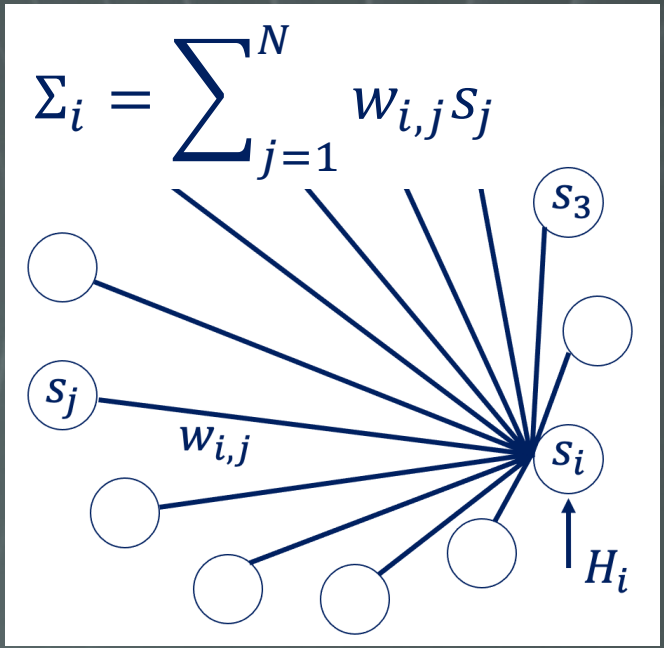
$$s_i = sign(\Sigma_i + H_i)$$

$$\Sigma_i = \sum_{j=1}^{N} w_{i,j} s_j$$

$$s_i \in (-1,1)$$

No dynamics on the synapses:

$$\dot{w}_{ij} = 0$$



Hopfield Model: Stored Patterns

Let be:

$\bar{s} = \{s_i\}; i = 1,2, \dots N$

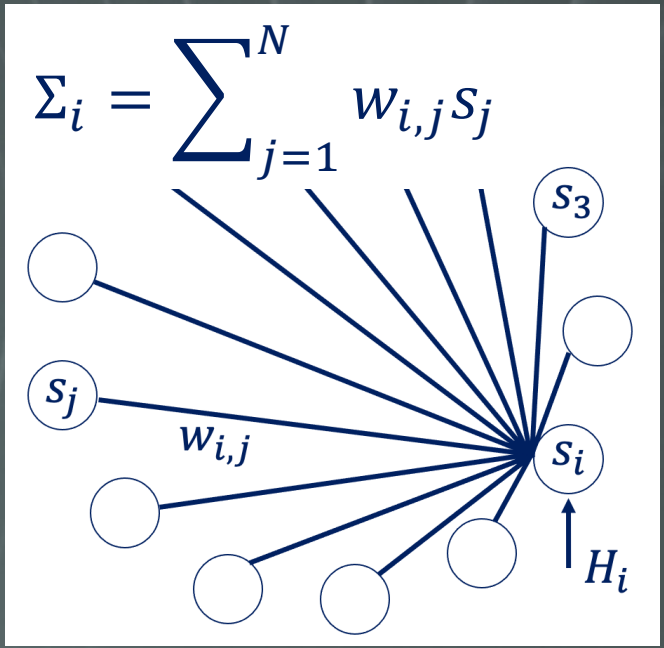
the state vector of the neurons. Let be

$\bar{w} = \{w_{i,j}\}; i,j = 1,2, \dots N$

the synaptic matrix, and let there be

$\overline{\xi^{\mu}} = \{\xi^{\mu}{}_i\}; \begin{cases} i = 1,2, \dots N \\ \mu = 1,2, \dots p \end{cases}$

p memory patterns

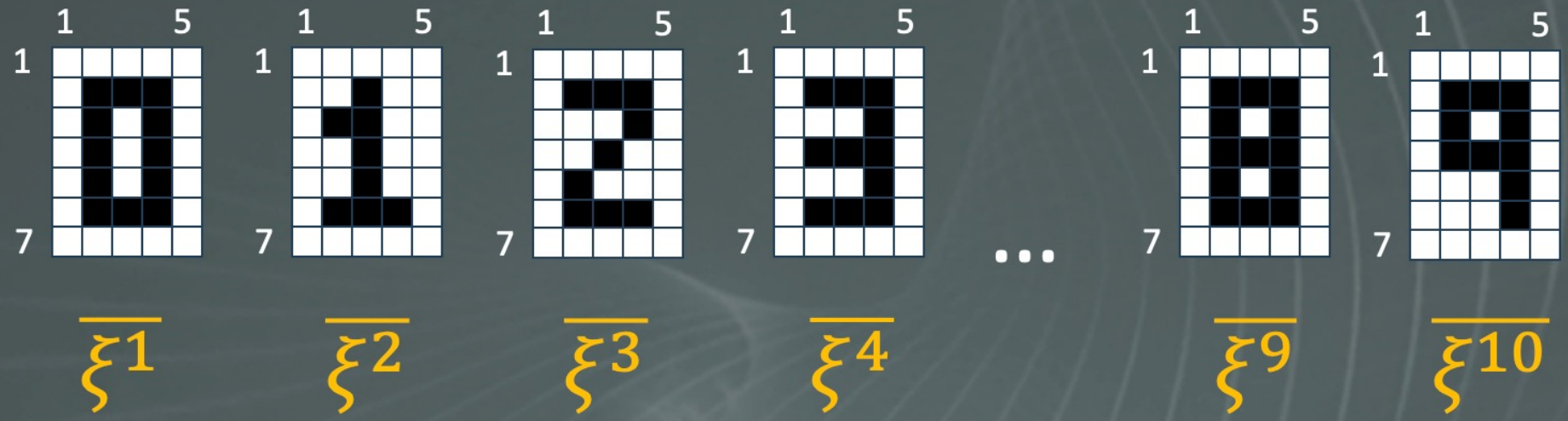


Hopfield Model: Stored Patterns

After defining the system, how should we design the synaptic matrix so that the network's dynamics produce attractors corresponding to the *p* stored memory patterns?

$$\overline{\xi^{\mu}} = \{\xi^{\mu}{}_{i}\}; \begin{cases} i = 1,2, \dots N \\ \mu = 1,2, \dots p \end{cases} \xrightarrow{??} \bar{w} = \{w_{i,j}\};$$

Hopfield Model: Stored Patterns



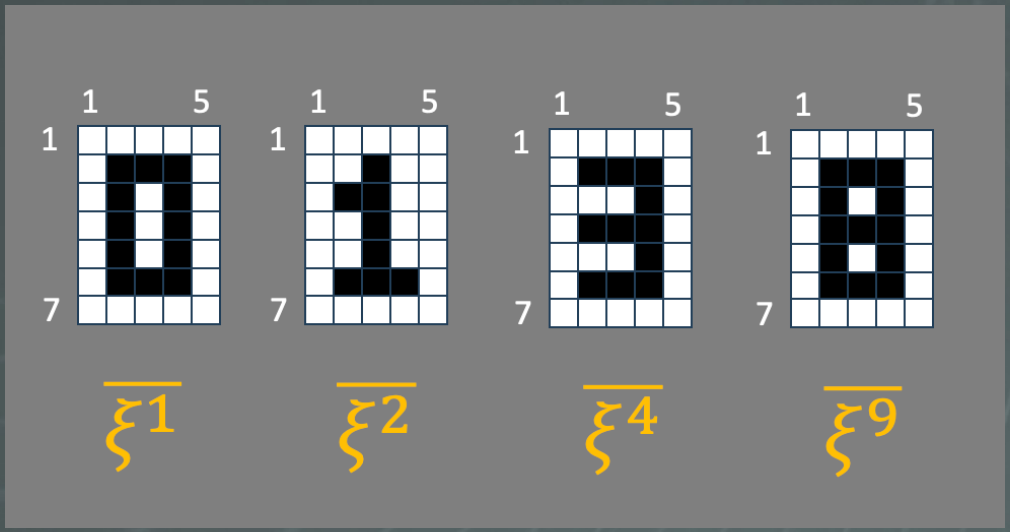
$\overline{\xi 1}$ $\overline{\xi 2}$ $\overline{\xi 3}$ $\overline{\xi 4}$ ... $\overline{\xi 9}$ $\overline{\xi 10}$

$$\overline{\xi^{\mu}} = \{\xi^{\mu}{}_i\}; \begin{cases} i = 1,2, \dots N \\ \mu = 1,2, \dots p \end{cases}$$

$\xi^{\mu}{}_i = +1$

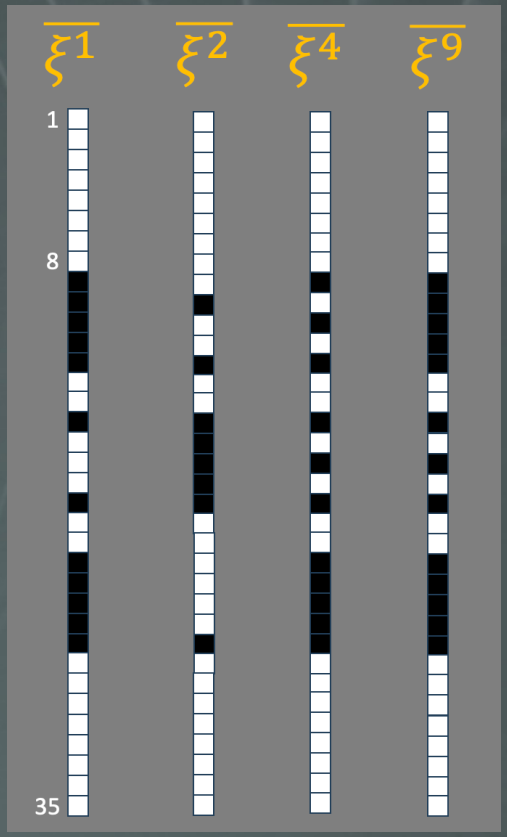
$\xi^{\mu}{}_i = -1$

Hopfield Model: Stored Patterns



$\xi^{\mu}{}_{i} = +1$

$\xi^{\mu}{}_{i} = -1$



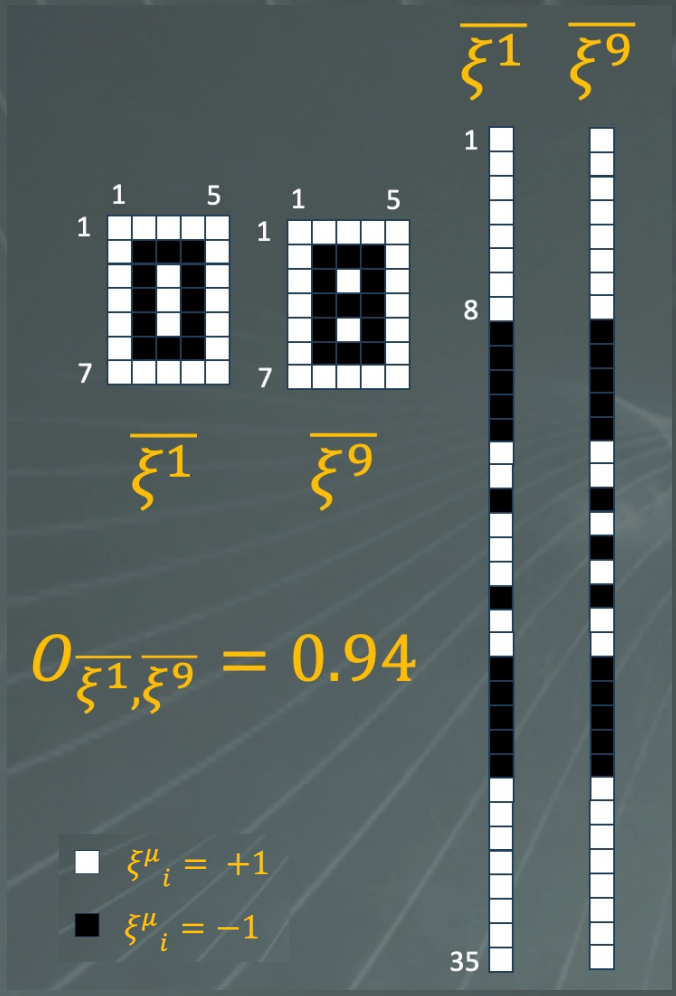
Hopfield Model: Overlap $O_{\bar{a},\bar{b}}$

The overlap $O_{\bar{a},\bar{b}}$ between two vectors

$$\bar{a} = \{a_i\}; i = 1,2, \dots N$$
$$\bar{b} = \{b_i\}; i = 1,2, \dots N$$

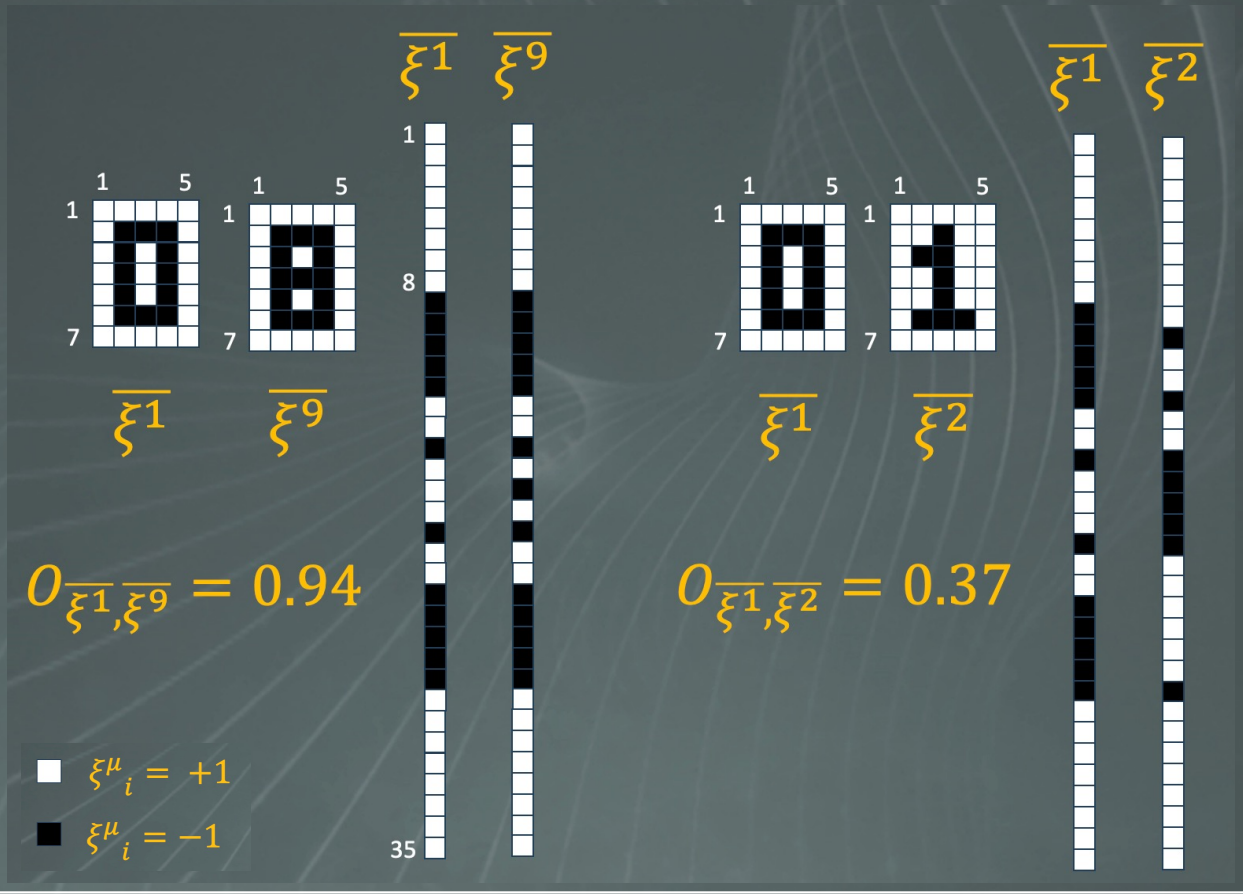
with $a_i \in (-1,1)$ and $b_i \in (-1,1)$ is

$$O_{\bar{a},\bar{b}} = \frac{1}{N} \sum_{i=1}^{N} a_i \, b_i$$



Hopfield Model: Overlap $O_{\bar{a},\bar{b}}$

$$O_{\bar{a},\bar{b}} = \frac{1}{N}\sum_{i=1}^{N} a_i\, b_i$$




HOPFIELD MODELL:
ASSOCIATIVE RECALL $p = 1$

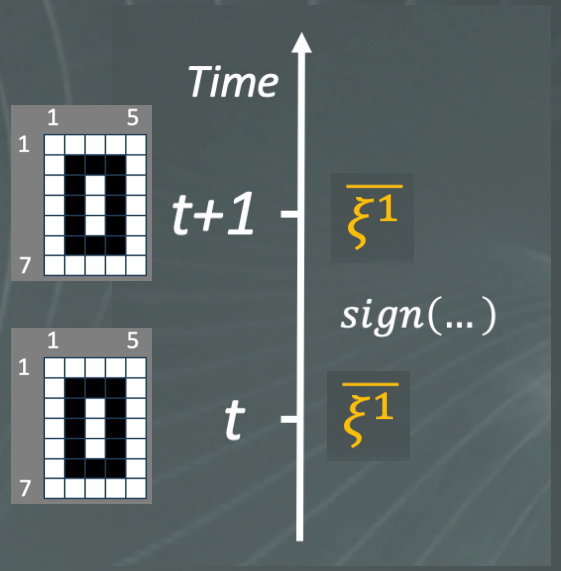
Hopfield Model: Associative Recall *p = 1*

Single stored pattern

$$\overline{\xi^1} = \{\xi^1{}_i\}; i = 1,2, \dots N$$

At time step t, the pattern $\overline{\xi^1}$ is presented to the network. We have:

$$s_i(t+1) = sign\left(\sum_{j=1}^{N} w_{i,j}\xi^1{}_j\right)$$



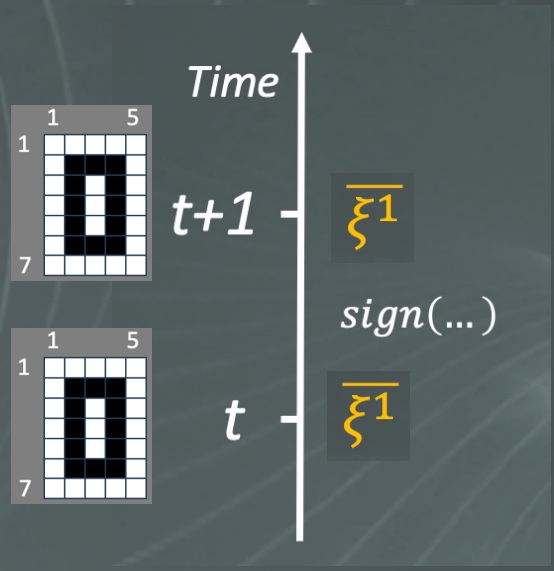
Hopfield Model: Associative Recall *p = 1*

if $\overline{\xi^1}$ is a dynamical attractor:

$$s_i(t+1) = sign\left(\sum_{j=1}^{N} w_{i,j}\xi^1{}_j\right) = \xi^1{}_j \qquad \forall i$$

This suggests defining $w_{i,j}$ as

$$w_{i,j} = \frac{1}{N}\xi^1{}_i\xi^1{}_j$$



Hopfield Model: Associative Recall *p = 1*

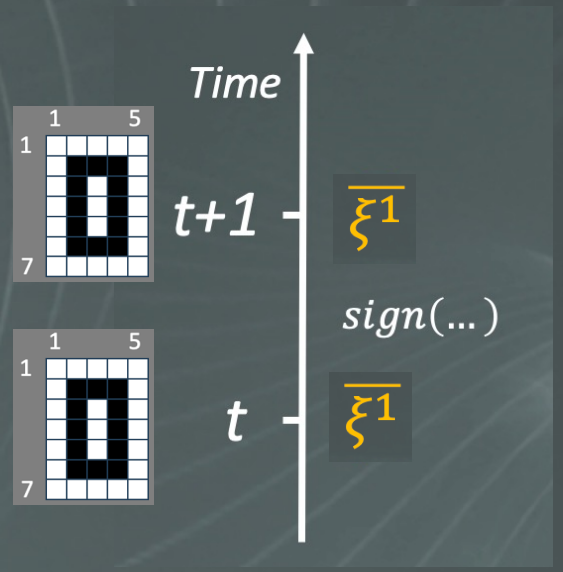
By substituting

$$s_i(t+1) = sign\left(\sum_{j=1}^{N} w_{i,j}\xi^1{}_j\right)$$

$$= sign\left(\frac{1}{N}\sum_{j=1}^{N} \xi^1{}_i\xi^1{}_j\xi^1{}_j\right)$$

$$= sign\left(\frac{1}{N}\xi^1{}_i\sum_{j=1}^{N} \xi^1{}_j\xi^1{}_j\right)$$

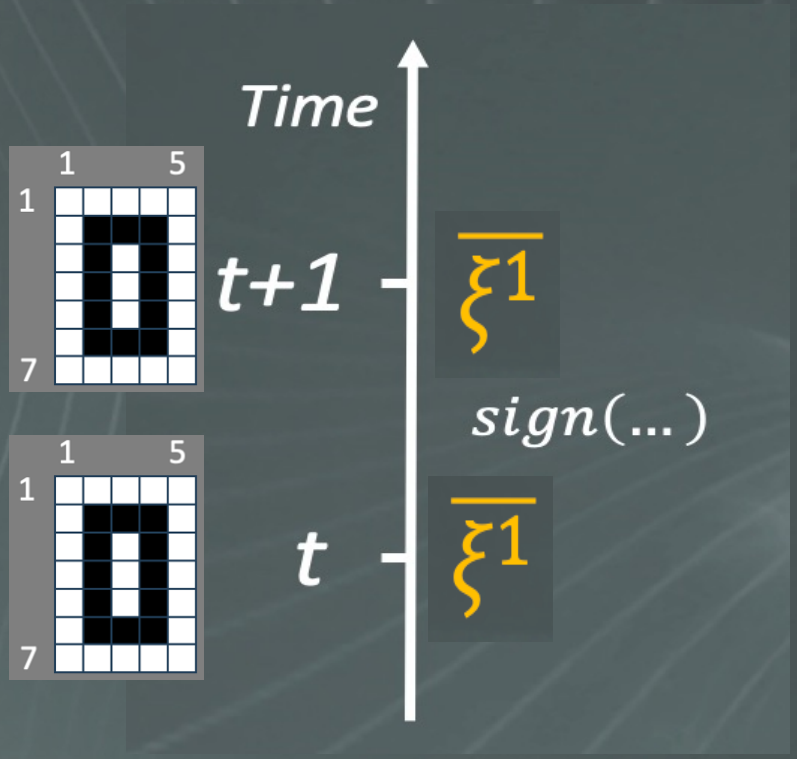
$$= sign\left(\xi^1{}_i\right) = \xi^1{}_i$$



Hopfield Model: Associative Recall *p = 1*

In summary, $\overline{\xi^1}$ is an attractor of the dynamics of the Hopfield model if and only if:

$$w_{i,j} = \frac{1}{N} \xi^1{}_i \xi^1{}_j$$



Hopfield Model: Associative Recall *p = 1*

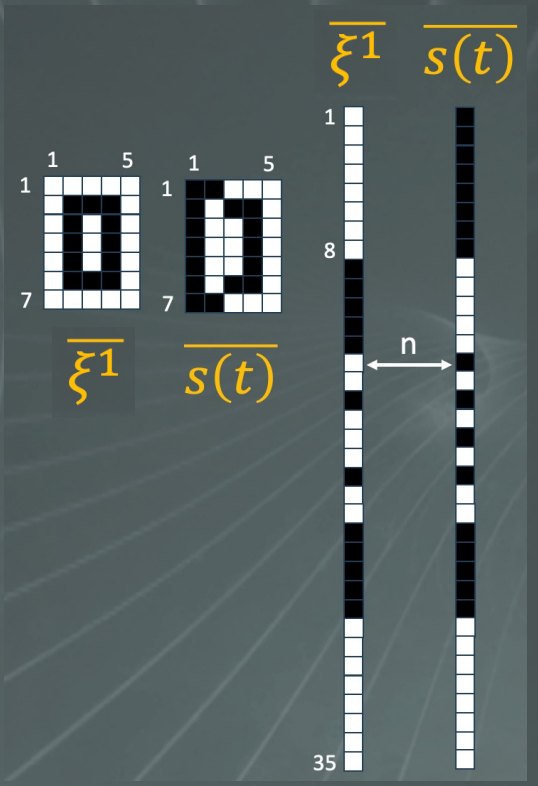
Single stored pattern

$$\overline{\xi^1} = \{\xi^1{}_i\}; i = 1,2, \dots N$$

At time step t, the pattern $\bar{s}(t)$ is presented to the network. We have:

$$\bar{s}(t) = \{s_i(t)\}$$

$$s_i(t) = \begin{cases} -\xi^1{}_i & i = 1,2, \dots, n \\ +\xi^1{}_i & i = n+1, \dots, N \end{cases}$$



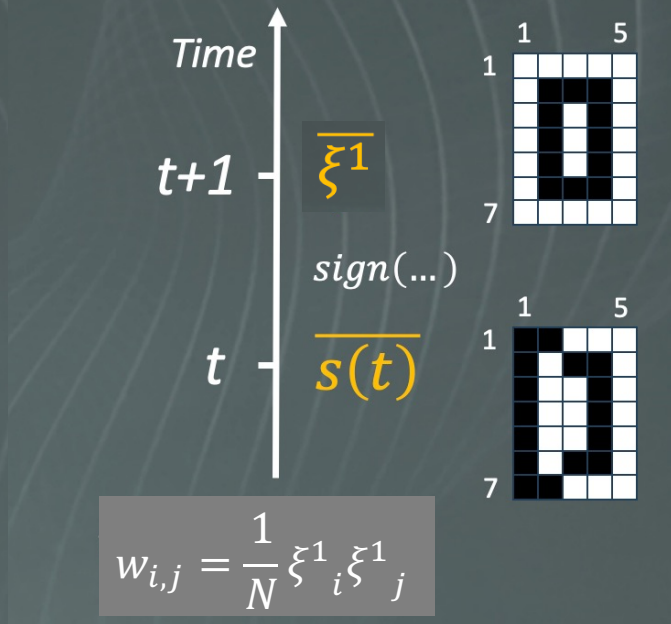
Hopfield Model: Associative Recall *p = 1*

We have

$$s_i(t+1) = sign\left(\sum_{j=1}^{N} w_{i,j} s_j(t)\right)$$

$$= sign\left(\frac{1}{N}\sum_{j=1}^{N} \xi^1{}_i \xi^1{}_j s_j(t)\right)$$

$$= sign\left(\frac{1}{N}\xi^1{}_i \sum_{j=1}^{N} \xi^1{}_j s_j(t)\right)$$

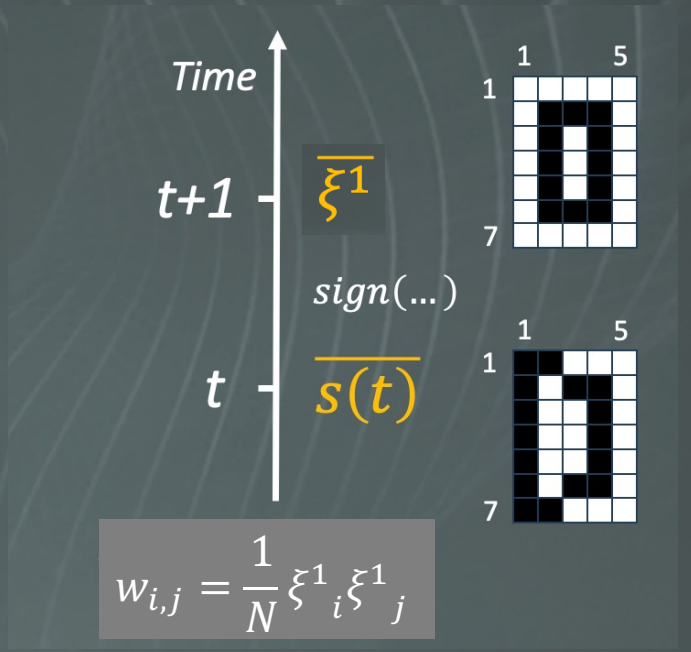


Hopfield Model: Associative Recall *p = 1*

$$s_i(t+1) = sign\left(\frac{1}{N}\xi^1{}_i \sum_{j=1}^{N} \xi^1{}_j s_j(t)\right)$$

The key idea is to split the summation into two terms

$$\sum_{j=1}^{N} \xi^1{}_j s_j(t) = \sum_{j=1}^{n} \xi^1{}_j s_j(t) + \sum_{j=n+1}^{N} \xi^1{}_j s_j(t)$$



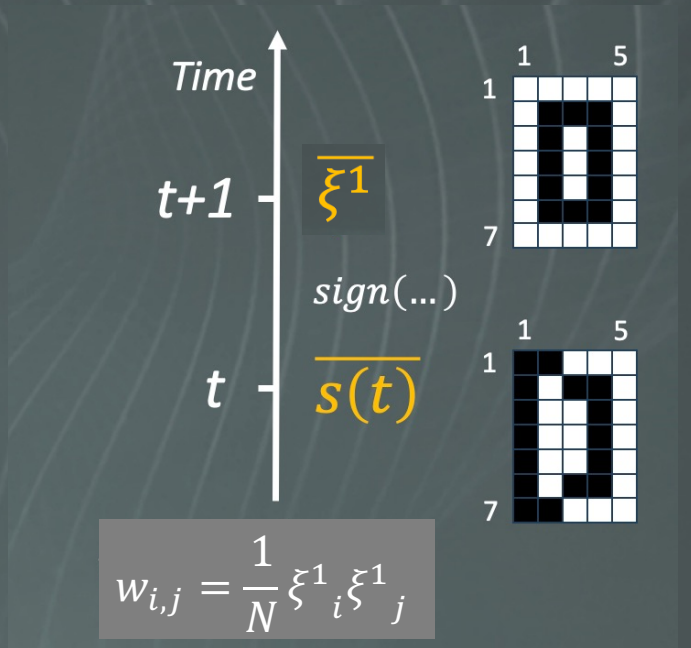
Hopfield Model: Associative Recall *p = 1*

$$s_i(t+1) = sign\left(\frac{1}{N}\xi^1{}_i \sum_{j=1}^{N} \xi^1{}_j s_j(t)\right)$$

Given the definition of $\bar{s}(t)$

$$\sum_{j=1}^{N} \xi^1{}_j s_j(t) = \sum_{j=1}^{n} \xi^1{}_j \left(-\xi^1{}_j\right)$$

$$+\sum_{j=n+1}^{N} \xi^1{}_j \xi^1{}_j = -n + (N-n)$$



Hopfield Model: Associative Recall *p = 1*

then

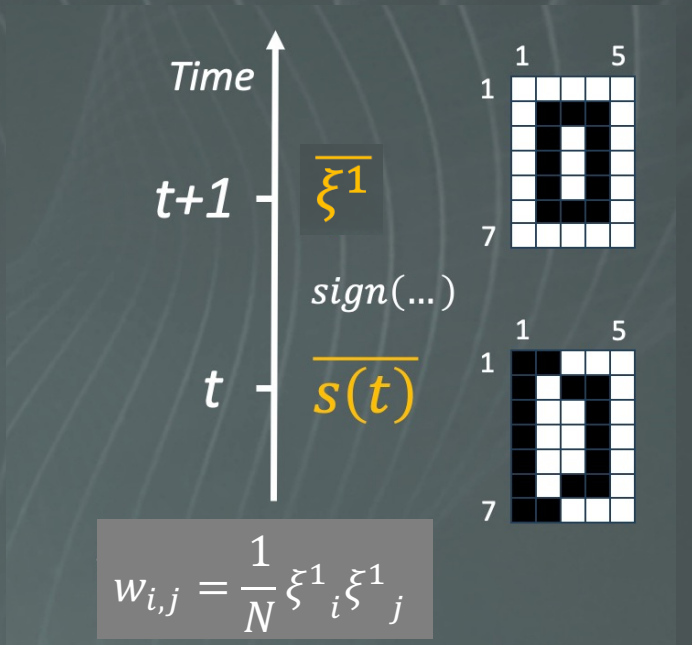
$$\sum_{j=1}^{N} \xi^1{}_j s_j(t) = -n + (N-n) = N - 2n$$

Sostituendo in

$$s_i(t+1) = sign\left(\frac{1}{N}\xi^1{}_i \sum_{j=1}^{N} \xi^1{}_j s_j(t)\right)$$

we have

$$s_i(t+1) = sign\left(\left(1 - \frac{2n}{N}\right)\xi^1{}_i\right)$$



Hopfield Model: Associative Recall *p = 1*

We have

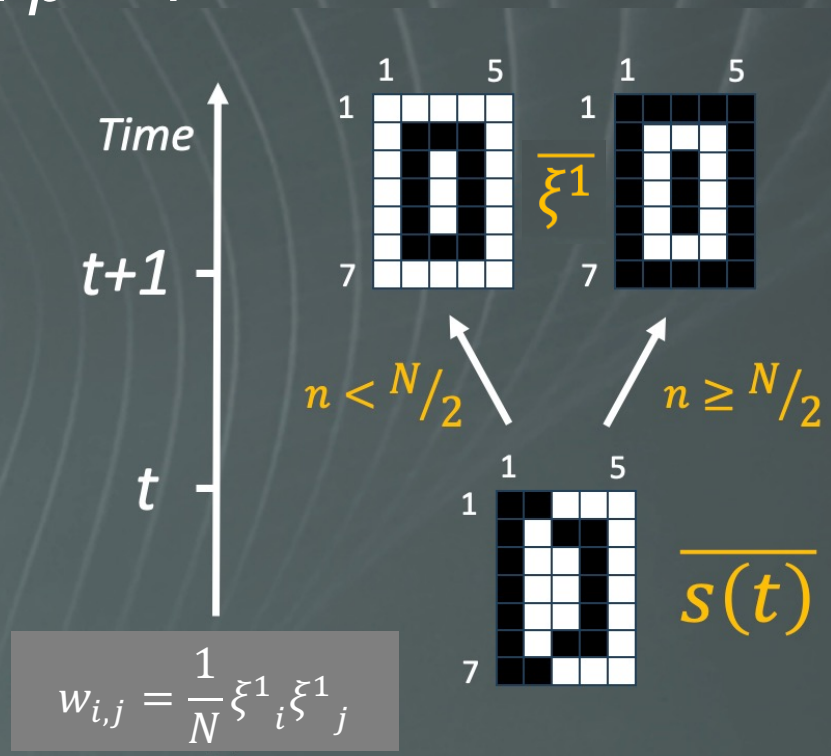
$$s_i(t+1) = sign\left(\left(1 - \frac{2n}{N}\right)\xi^1{}_i\right)$$

if $n < {}^N/_2$ we obtain

$$s_i(t+1) = sign\left(\xi^1{}_i\right) = \xi^1{}_i$$

and if $n \geq {}^N/_2$ we obtain

$$s_i(t+1) = sign\left(-\xi^1{}_i\right) = -\xi^1{}_i$$

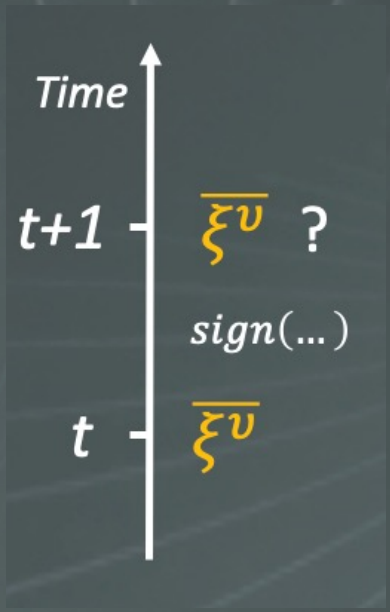


HOPFIELD MODELL: ASSOCIATIVE RECALL $p > 1$

Hopfield Model: Associative Recall *p > 1*

p stored patterns

$$\overline{\xi^{\mu}} = \{\xi^{\mu}{}_{i}\}; \begin{cases} i = 1,2, \dots N \\ \mu = 1,2, \dots p \end{cases}$$



At time step t , the pattern $\overline{\xi^{v}}$ is presented to the network. We have

$$s_i(t+1) = sign\left(\sum_{j=1}^{N} w_{i,j}\xi_j^v\right)$$

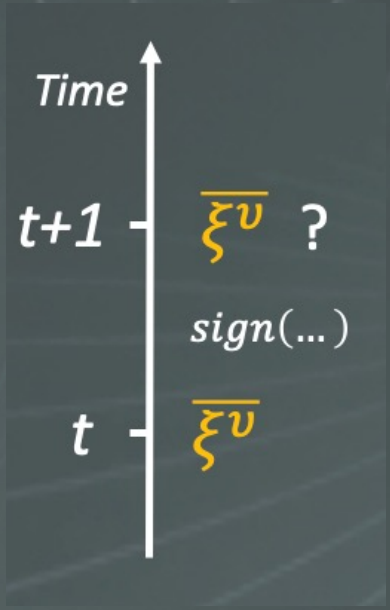
Hopfield Model: Associative Recall *p > 1*

For *p = 1* we had defined

$$w_{i,j} = \frac{1}{N} \xi^1{}_i \xi^1{}_j$$

For *p > 1* this can be generalized as

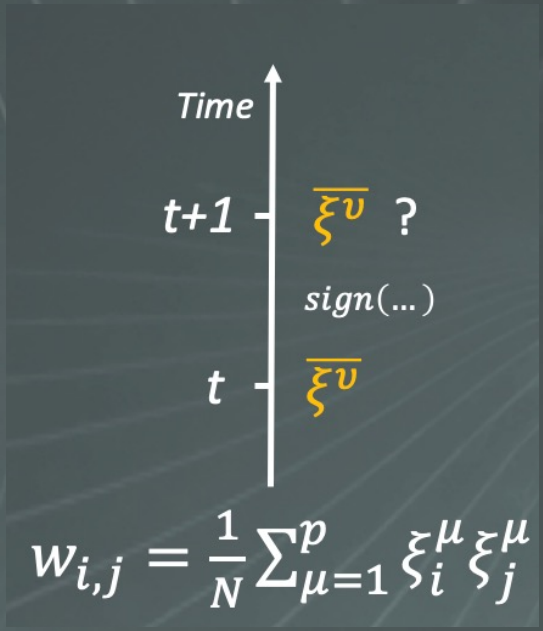
$$w_{i,j} = \frac{1}{N} \sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu$$



Hopfield Model: Associative Recall *p > 1*

We have

$$s_i(t+1) = sign\left(\sum_{j=1}^{N} w_{i,j}\xi_j^v\right) = sign\left(\frac{1}{N}\sum_{\mu=1}^{p}\sum_{j=1}^{N}\xi_i^\mu\xi_j^\mu\xi_j^v\right)$$



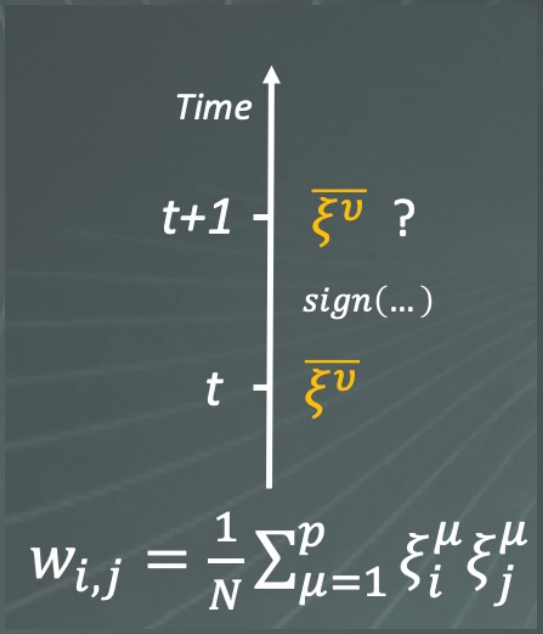
Hopfield Model: Associative Recall $p > 1$

$$sign\left(\frac{1}{N}\sum_{\mu=1}^{p}\sum_{j=1}^{N}\xi_i^{\mu}\xi_j^{\mu}\xi_j^{v}\right)$$

Let us consider the argument of the sign function

$$\frac{1}{N}\sum_{\mu=1}^{p}\sum_{j=1}^{N}\xi_i^{\mu}\xi_j^{\mu}\xi_j^{v} =$$

$$\frac{1}{N}\sum_{\mu=1}^{p}\xi_i^{\mu}\sum_{j=1}^{N}\xi_j^{\mu}\xi_j^{v}$$



Hopfield Model: Associative Recall $p > 1$

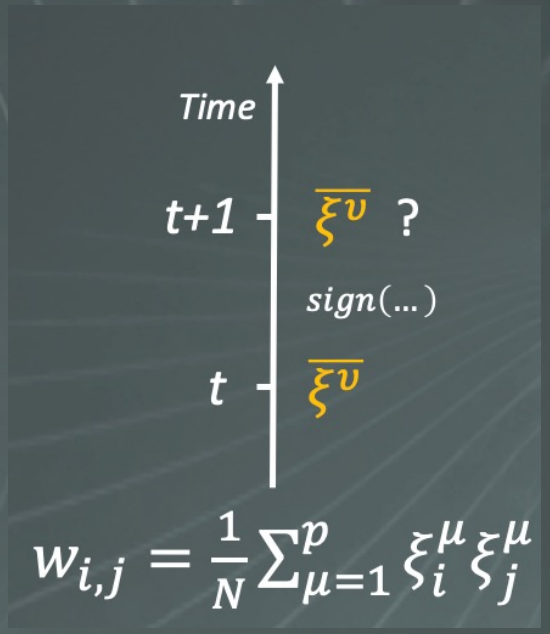
Split the summation over μ into two terms:

$$\frac{1}{N}\sum_{\mu=1}^{p}\xi_i^{\mu}\sum_{j=1}^{N}\xi_j^{\mu}\xi_j^{\upsilon} =$$

$$\frac{1}{N}\xi_i^{\upsilon}\sum_{j=1}^{N}\xi_j^{\upsilon}\xi_j^{\upsilon} +$$

$$\frac{1}{N}\sum_{\substack{\mu=1\\ \mu\neq\upsilon}}^{p}\xi_i^{\mu}\sum_{j=1}^{N}\xi_j^{\mu}\xi_j^{\upsilon} = \xi_i^{\upsilon} + \sum_{\substack{\mu=1\\ \mu\neq\upsilon}}^{p}\xi_i^{\mu}O_{\overline{\xi^{\mu}},\overline{\xi^{\upsilon}}}$$

Crossover Term



$$O_{\bar{a},\bar{b}} = \frac{1}{N}\sum_{i=1}^{N}a_i\,b_i$$

Hopfield Model: Associative Recall *p* > *1*

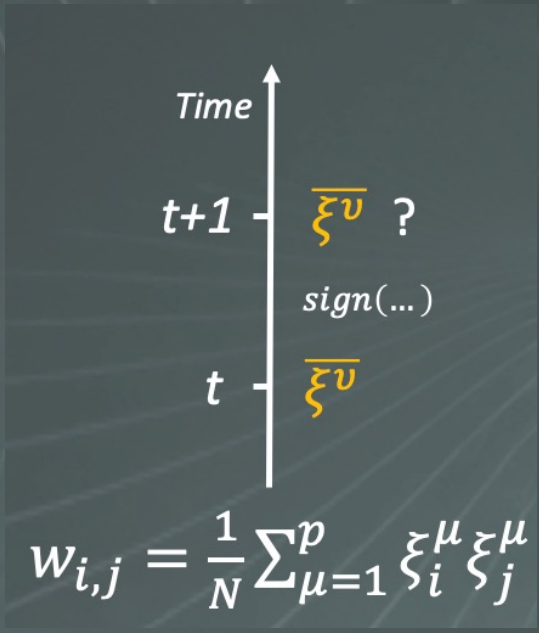
We obtain

$$s_i(t+1) = sign\left(\xi_i^{\upsilon} + \sum_{\substack{\mu=1 \\ \mu\neq\upsilon}}^{p} \xi_i^{\mu} O_{\overline{\xi^{\mu}},\overline{\xi^{\upsilon}}}\right)$$

There is an Associative Recall if

$$\left|\sum_{\substack{\mu=1 \\ \mu\neq\upsilon}}^{p} \xi_i^{\mu} O_{\overline{\xi^{\mu}},\overline{\xi^{\upsilon}}}\right| \ll 1 \quad \Longrightarrow \quad s_i(t+1) = \xi_i^{\upsilon}$$

Or if $O_{\overline{\xi^{\mu}},\overline{\xi^{\upsilon}}} = 0$



Hopfield Model: Associative Recall $p > 1$

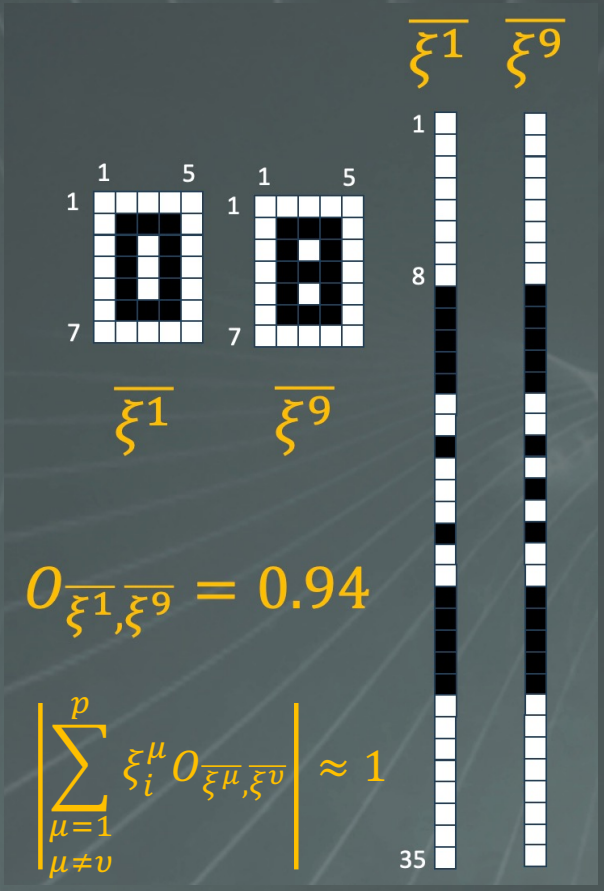
We obtain

$$s_i(t+1) = sign\left(\xi_i^v + \sum_{\substack{\mu=1 \\ \mu \neq v}}^{p} \xi_i^\mu O_{\overline{\xi^\mu},\overline{\xi^v}}\right)$$

there is an Associative Recall if

$$\left|\sum_{\substack{\mu=1 \\ \mu \neq v}}^{p} \xi_i^\mu O_{\overline{\xi^\mu},\overline{\xi^v}}\right| \ll 1 \quad \Longrightarrow \quad s_i(t+1) = \xi_i^v$$

or if $O_{\overline{\xi^\mu},\overline{\xi^v}} = 0$



Hopfield Model: Associative Recall $p > 1$

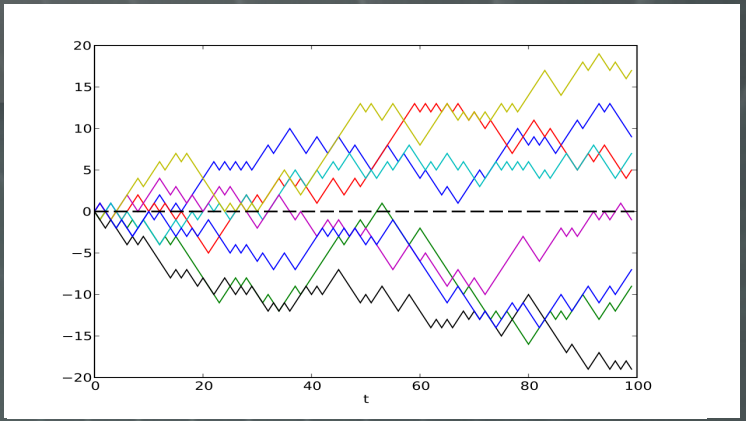
How can we estimate the crossover term?

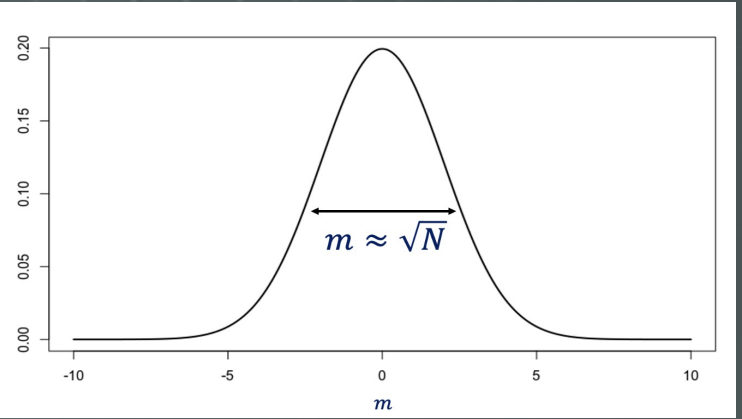
$$\frac{1}{N}\sum_{\substack{\mu=1\\ \mu\neq\upsilon}}^{p}\xi_i^{\mu}\sum_{j=1}^{N}\xi_j^{\mu}\xi_j^{\upsilon}$$

Random Walk

- We flip a coin
- Step right for heads, left for tails
- The probability of going right or left is ½
- After N flips, what's the probability of being m steps from the origin $\mathrm{P}(m, N)$?

$$m = \sum_{i=1}^{N} \xi_i \qquad \xi_i = \begin{cases} +1; p_T = 0.5 \\ -1; p_C = 0.5 \end{cases}$$





$$\lim_{N\to\infty} P\,(m, N) = \left(\frac{2}{\pi N}\right)^{1/2} e^{-m^2/2N}$$

Hopfield Model: Associative Recall *p > 1*

How can we estimate the crossover term?

$$\frac{1}{N}\sum_{\substack{\mu=1\\ \mu\neq\upsilon}}^{p}\xi_i^{\mu}\sum_{j=1}^{N}\xi_j^{\mu}\xi_j^{\upsilon} \approx O\left(\frac{1}{N}\sqrt{(p-1)N}\right) \approx O\left(\sqrt{\frac{(p-1)}{N}}\right)$$

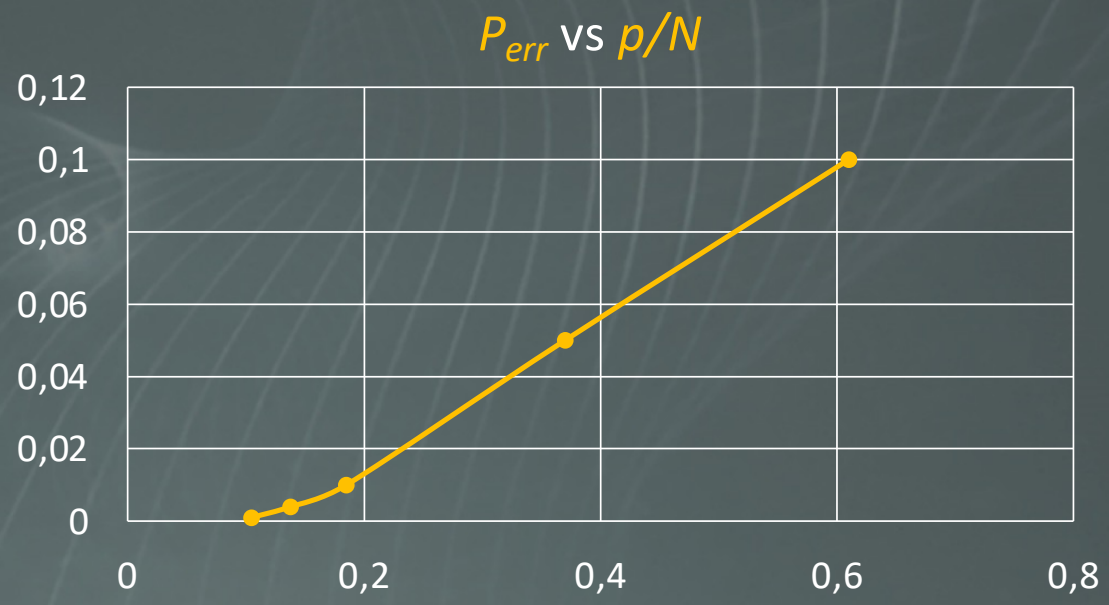
Associative Recall

$$s_i(t+1) = \xi_i^{\upsilon} \quad if \quad \frac{(p-1)}{N} \ll 1$$

Hopfield Model: Associative Recall $p > 1$

For p < 0.185 N, less than 1% of the bits in the patterns are unstable

P_{err}	p/N
0.001	0.105
0.004	0.138
0.010	0.185
0.050	0.370
0.100	0.610



Hopfield Model: Associative Recall *p* > *1*

PHYSICAL REVIEW A VOLUME 32, NUMBER 2 AUGUST 1985

Spin-glass models of neural networks

Daniel J. Amit and Hanoch Gutfreund
Racah Institute of Physics, Hebrew University, 91904 Jerusalem, Israel

H. Sompolinsky
Department of Physics, Bar-Ilan University, 52100 Ramat-Gan, Israel

(Received 22 March 1985)

Two dynamical models, proposed by Hopfield and Little to account for the collective behavior of neural networks, are analyzed. The long-time behavior of these models is governed by the statistical mechanics of infinite-range Ising spin-glass Hamiltonians. Certain configurations of the spin system, chosen at random, which serve as memories, are stored in the quenched random couplings. The present analysis is restricted to the case of a finite number p of memorized spin configurations, in the thermodynamic limit. We show that the long-time behavior of the two models is identical, for all temperatures below a transition temperature T_c. The structure of the stable and metastable states is displayed. Below T_c, these systems have $2p$ ground states of the Mattis type: Each one of them is fully correlated with one of the stored patterns. Below $T \sim 0.46 T_c$, additional dynamically stable states appear. These metastable states correspond to specific mixings of the embedded patterns. The thermodynamic and dynamic properties of the system in the cases of more general distributions of random memories are discussed.

I. INTRODUCTION

A. Models of neural networks

Recently, a number of models have been proposed which view the human memory as a collective property of large interconnected neural networks. Two such models, one proposed recently by Hopfield[1] and a closely related model, proposed some ten years ago by Little,[2] are the focus of this paper. We briefly review the physiological background of these models. For more details the reader is referred to Refs. 1 and 2 and to a recent study of these models by Peretto.[3] In both models each neuron is viewed as an Ising spin with two possible states: an "up" position or a "down" position depending on whether the neuron has, or has not, fired an electrochemical signal (within an interval of time of the order of a millisecond). The state of the network of N such neurons at time t is defined as the instantaneous configuration of all the spin variables at time t:

$$|\alpha,t\rangle = |S_1^\alpha, S_2^\alpha, \ldots, S_N^\alpha; t\rangle \ . \tag{1.1}$$

The dynamic evolution of these states, in the phase space of 2^N states, is determined by the interactions among the neurons. The neurons are interconnected by synaptic junctions of strength J_{ij}, which determine the contribution of a signal fired by the jth neuron to the postsynaptic potential which acts on the ith neuron. This contribution can be either positive (excitatory synapse) or negative (inhibitory synapse). The potential V_i on each neuron is the sum of all postsynaptic potentials delivered to it in an integrating period of time, of the order of a few milliseconds, i.e.,

$$V_i = \sum_j J_{ij}(S_j + 1) \ . \tag{1.2}$$

In the absence of noise, or external perturbation, each neuron fires a signal if its potential V_i exceeds a threshold value U_i. Thus the stable states of the network will be those configurations in which each of the spin variables S_i is aligned with its molecular field $h_i = V_i - U_i$, i.e.,

$$S_i h_i = S_i(V_i - U_i) > 0 \ . \tag{1.3}$$

It will be assumed throughout the paper that the J_{ij}'s are symmetric, i.e., $J_{ij} = J_{ji}$. In such a case Eq. (1.3) is equivalent to the requirement that the configurations $\{S_i\}$ be local minima (i.e., stable to all single-spin flips) of the Hamiltonian

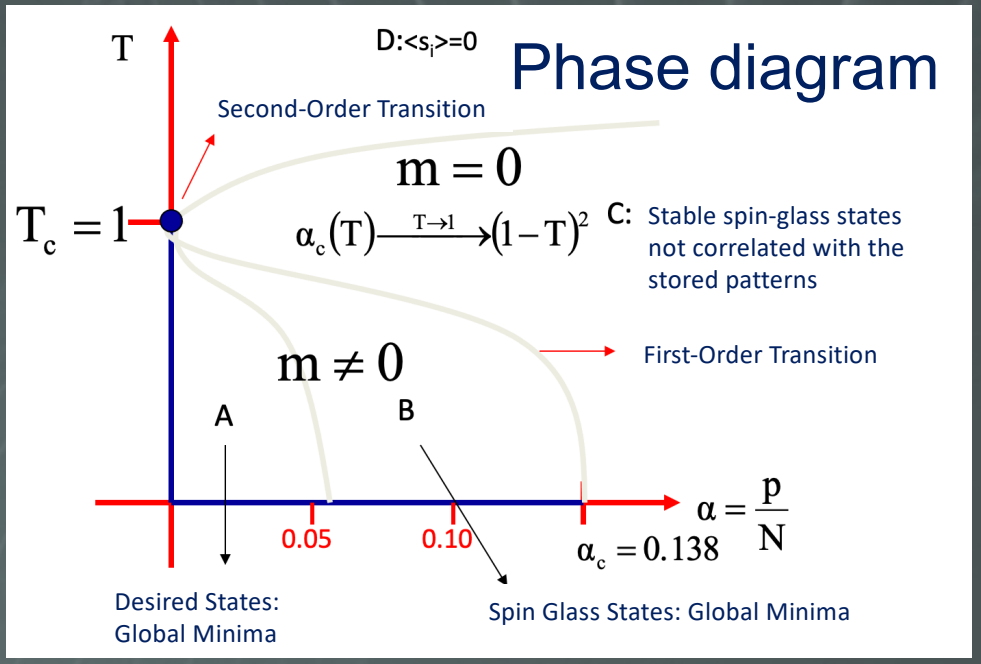
$$H = -\tfrac{1}{2}\sum_i h_i S_i = -\tfrac{1}{2}\sum_{i,j} J_{ij} S_i S_j \ , \tag{1.4}$$

where it is usually assumed that the threshold potentials satisfy $U_i \simeq \sum_j J_{ij}$. Thus there is no external field term in H. In the presence of noise there is a finite probability of having configurations other than those given by Eq. (1.3). This can be taken into account by introducing an effective temperature $1/\beta$, characterizing the level of noise in the system,[4] as will be described below.

For the network to have a capacity for learning and memory its stable configurations must be correlated with certain configurations, which are determined by the learning process. This is achieved by choosing the interactions J_{ij} to be given by

$$J_{ij} = \frac{1}{N}\sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu, \quad i \neq j \ . \tag{1.5}$$

The p sets of $\{\xi_i^\mu\}$ are certain configurations of the network which were fixed by the learning process. The ξ_i^μ are taken to be quenched random variables, assuming the values $+1$ and -1 with equal probabilities. Note that according to Eq. (1.5) every pair of neurons is connected.

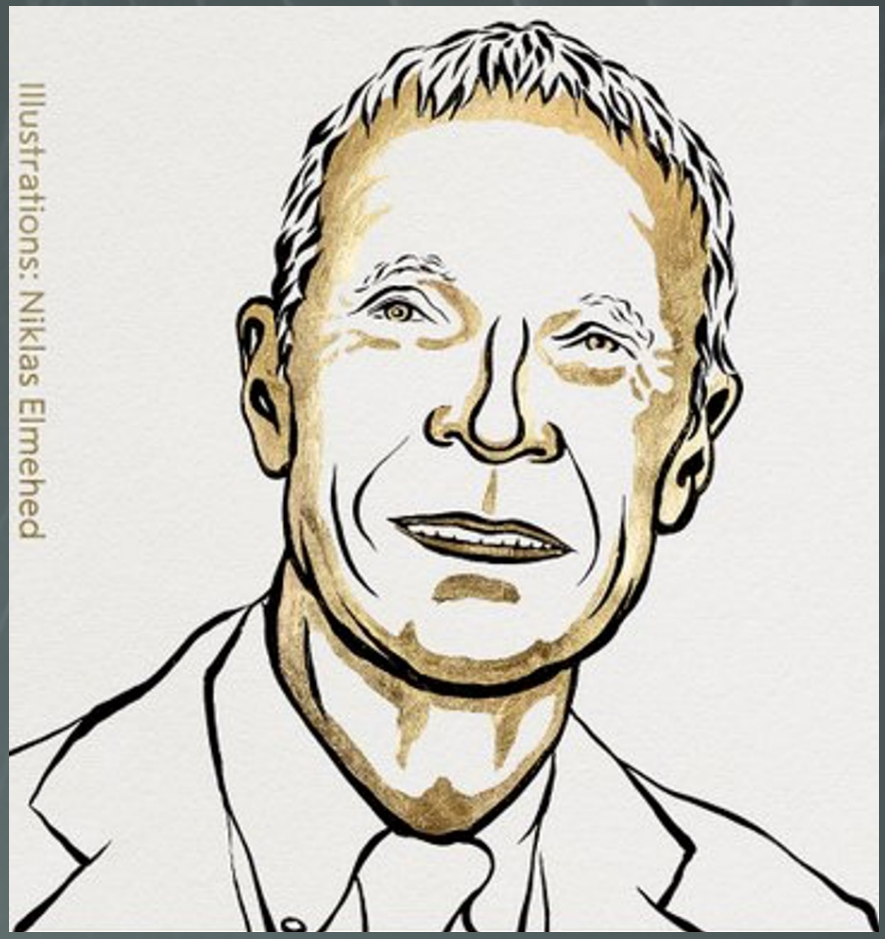


Phase transition:
Neural Networs are complex Systems

Hopfield Model

$$w_{i,j} = \frac{1}{N} \sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu$$

Prize motivation: “for foundational discoveries and inventions that enable machine learning with artificial neural networks”



J. J. Hopfield

Is Attention essentially a Hopfield Model in disguise ?

Attention Is All You Need: Transformer

- Paradigm shift in NLP: Transformer models revolutionized Natural Language Processing by replacing traditional sequential architectures with attention-based models, becoming the state of the art.
- Attention mechanism: The self-attention mechanism allows models to capture long-range dependencies and attend to all previous words in a sequence, providing an effective form of long-term memory.
- Scalability and efficiency: Transformers are highly parallelizable and well suited for training on large datasets using specialized hardware such as GPUs and TPUs.
- Real-world applications: They power many modern systems including machine translation, text generation, chatbots, search engines, and language understanding tasks.

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

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

*Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

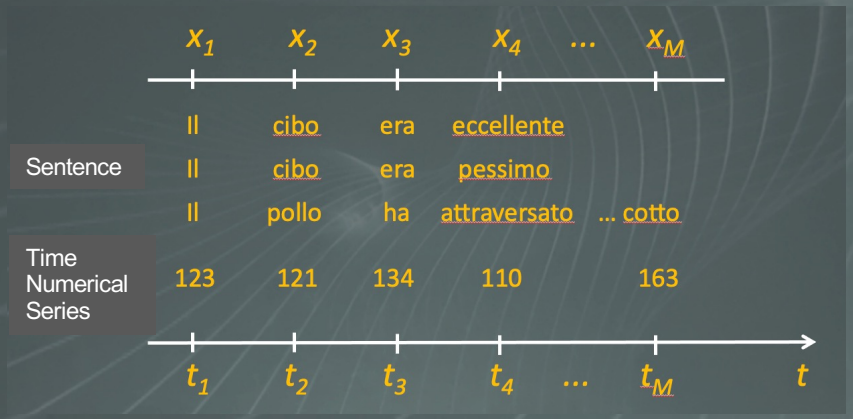
†Work performed while at Google Brain.

‡Work performed while at Google Research.

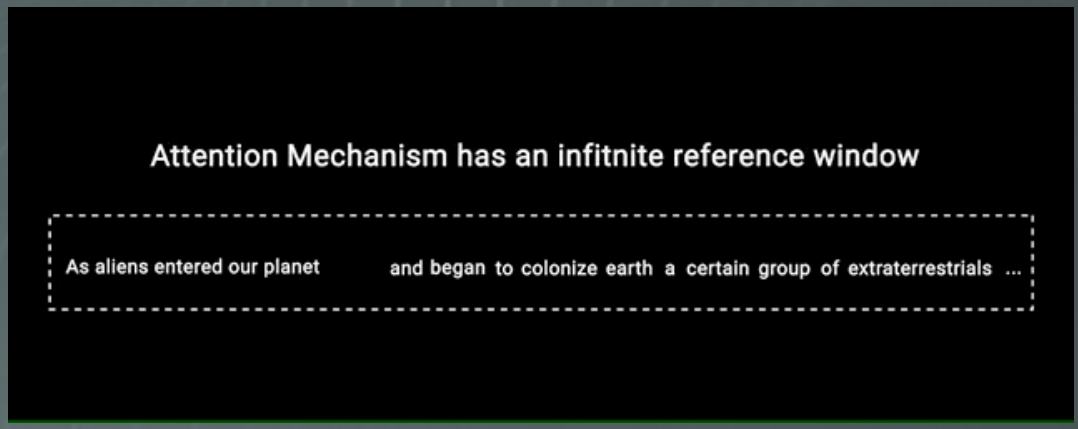
31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Attention Is All You Need: Transformer

The self-attention mechanism allows models to capture long-range dependencies and attend to all previous words in a sequence, providing an effective form of long-term memory



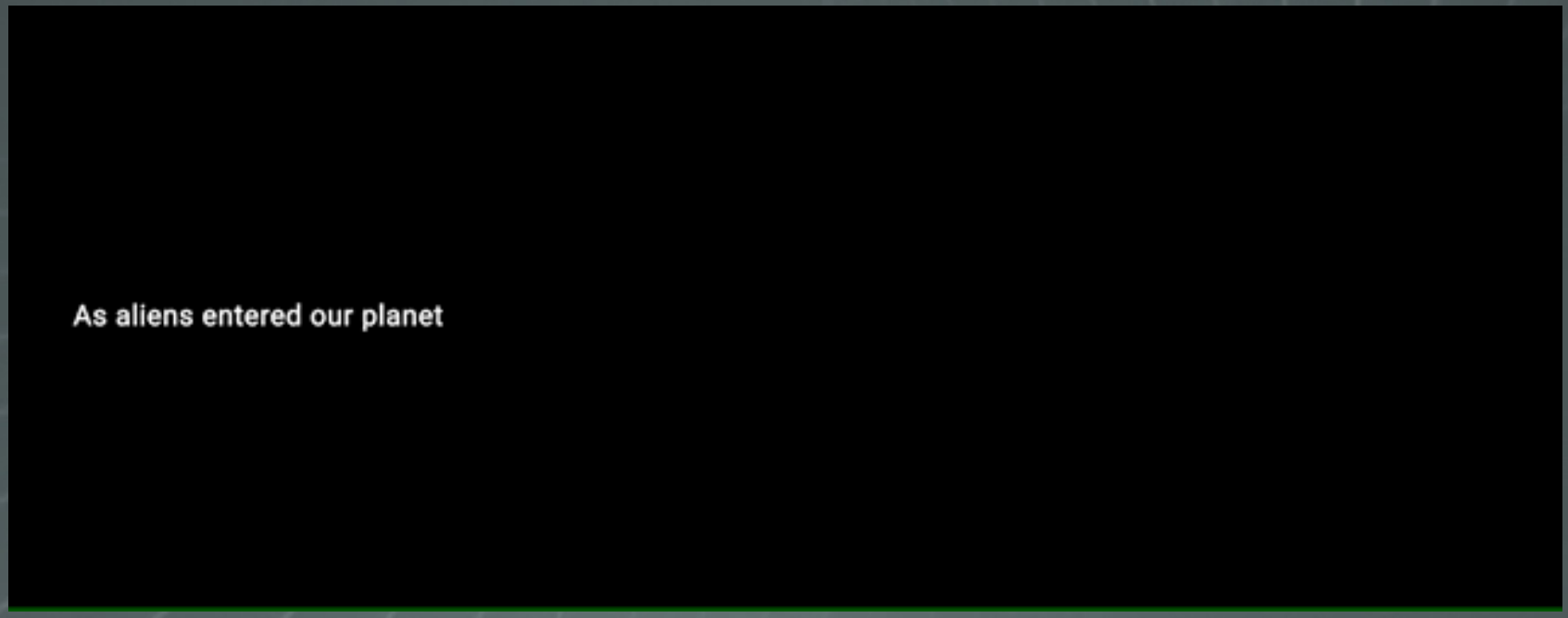
The input is a time series of varying values



Hypothetical reference window of Attention, RNN's, GRU's & LSTM's

Attention Is All You Need: Transformer

During training, the model learns which words to attend to

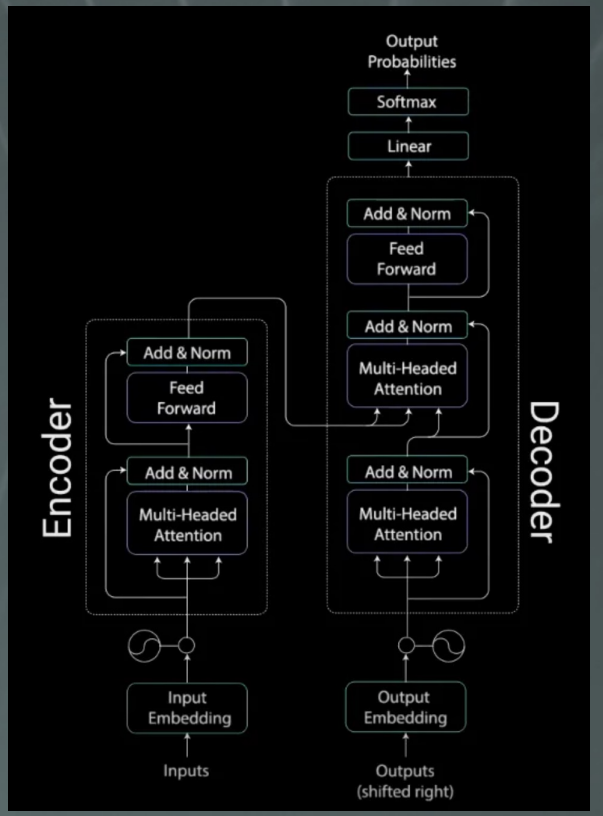


Attention mechanism focusing on different tokens while generating words 1 by 1

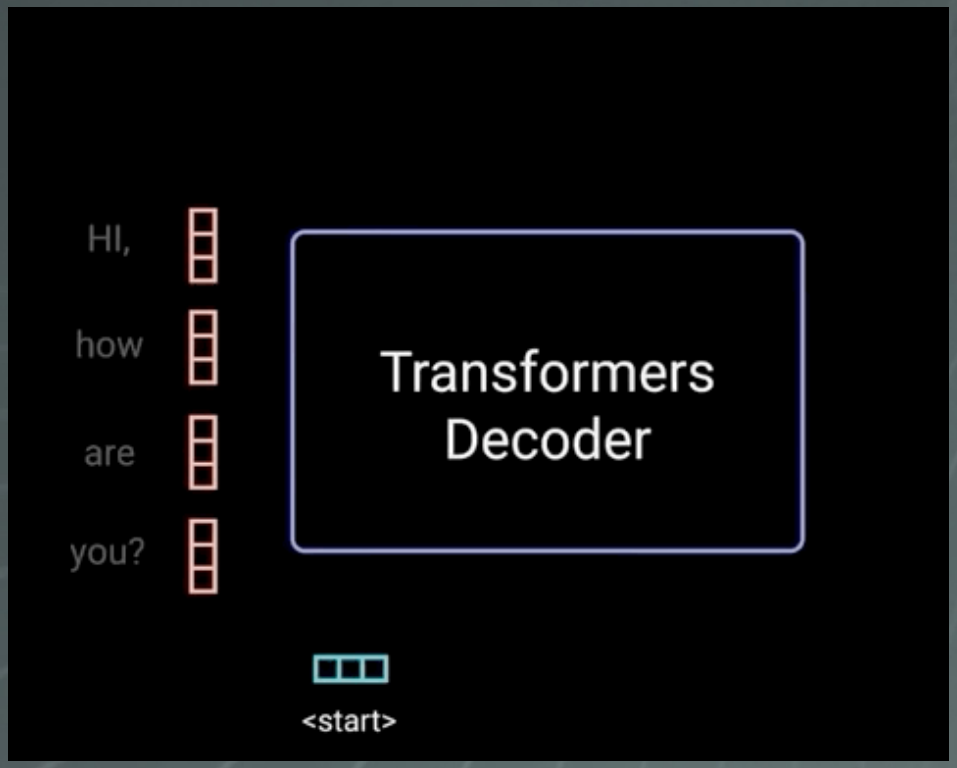
Attention Is All You Need: Transformer

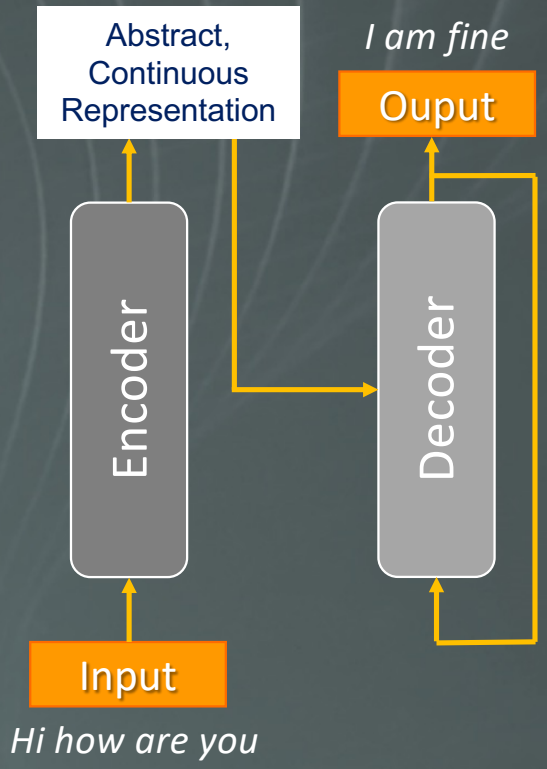
This is an attention-based Encoder-Decoder architecture.

- At a high level, the encoder transforms the input into an abstract, continuous representation.
- The decoder then iteratively generates the output, one step at a time, while simultaneously taking the previous output as input."



Attention Is All You Need: Transformer





Attention Is All You Need: Transformer

The self-attention mechanism begins by deriving query (Q), key (K), and value (V) vectors from the input.

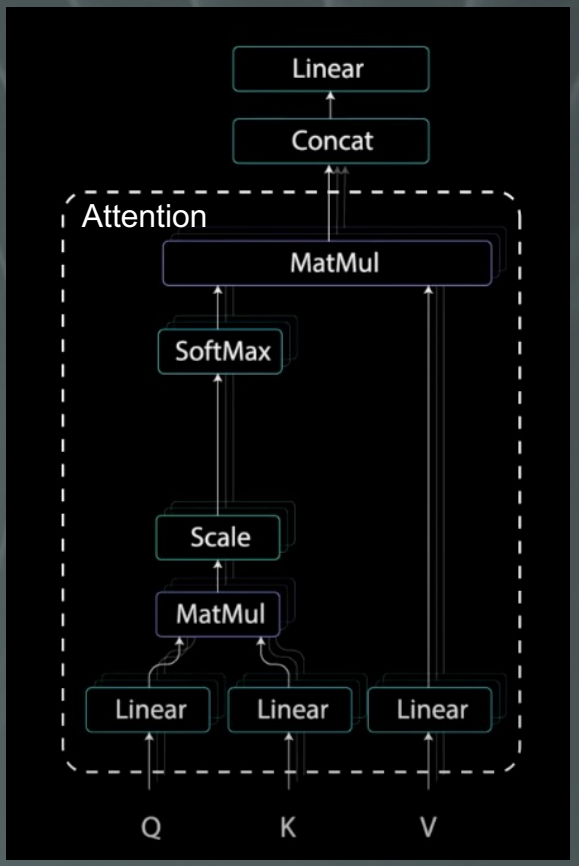
Input: Given a sequence of token embeddings $x_1, x_2, \ldots, x_n \in R^d$ three linear projections are learned (W_Q, W_K, W_V are learned weight matrices):

$q_i = W_Q x_i$ where q_i = query
$k_i = W_K x_i$ k_i = key
$v_i = W_V x_i$ v_i = value

Attention weights: The interaction between token i and token j is

$$a_{ij} = \frac{e^{\beta q_i \cdot k_j}}{\sum_m e^{\beta q_i \cdot k_m}}; \qquad \beta = \frac{1}{\sqrt{d}}$$

Output: The updated representation, attention, is $y_i = \frac{\sum_j v_i e^{\beta q_i \cdot k_j}}{\sum_m e^{\beta q_i \cdot k_m}}$



Interpretation: Each token becomes a weighted combination of value vectors, with weights that grow exponentially with query-key similarity.

Continuous Hopfield Network & Attention

Compare the two update rules

Attention $$y_i = \frac{\sum_j v_i e^{\beta q_i \cdot k_j}}{\sum_m e^{\beta q_i \cdot k_m}}$$

Identification

$x \leftrightarrow q$

$\xi^\mu \leftrightarrow k_j$

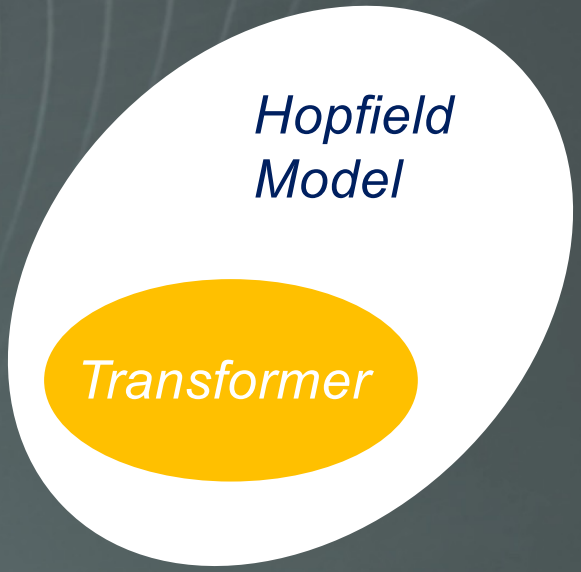
$v_i \leftrightarrow Stored\ Values$

Hopfield $$x_{new} = \frac{\sum_\mu \xi^\mu e^{\beta x \cdot \xi^\mu}}{\sum_\mu e^{\beta x \cdot \xi^\mu}}$$

Result: The attention mechanism is formally equivalent to the update rule of a continuous Hopfield network.

Interpretation: Attention performs associative retrieval from a set of stored vectors.

Attention as one-step associative retrieval



Attention Is All You Need

Hopfield Networks Is All You Need

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

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

[*]Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

[†]Work performed while at Google Brain.

[‡]Work performed while at Google Research.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Hopfield Networks is All You Need

Hubert Ramsauer[*] **Bernhard Schäfl**[*] **Johannes Lehner**[*] **Philipp Seidl**[*]
Michael Widrich[*] **Thomas Adler**[*] **Lukas Gruber**[*] **Markus Holzleitner**[*]
Milena Pavlović[‡,§] **Geir Kjetil Sandve**[§] **Victor Greiff**[‡] **David Kreil**[†]
Michael Kopp[†] **Günter Klambauer**[*] **Johannes Brandstetter**[*] **Sepp Hochreiter**[*,†]
[*]ELLIS Unit Linz, LIT AI Lab, Institute for Machine Learning, Johannes Kepler University Linz, Austria
[†]Institute of Advanced Research in Artificial Intelligence (IARAI)
[‡]Department of Immunology, University of Oslo, Norway
[§]Department of Informatics, University of Oslo, Norway

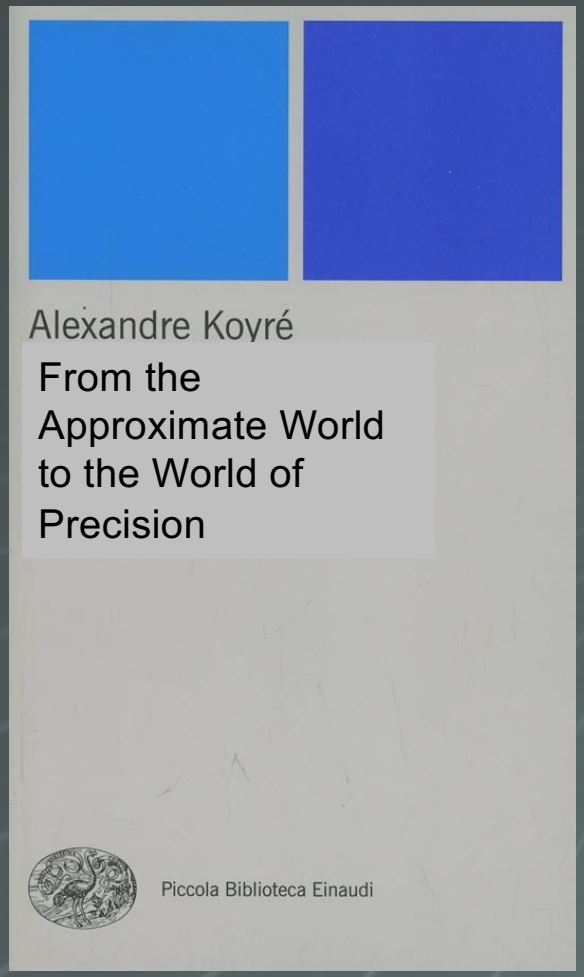
Abstract

We introduce a modern Hopfield network with continuous states and a corresponding update rule. The new Hopfield network can store exponentially (with the dimension of the associative space) many patterns, retrieves the pattern with one update, and has exponentially small retrieval errors. It has three types of energy minima (fixed points of the update): (1) global fixed point averaging over all patterns, (2) metastable states averaging over a subset of patterns, and (3) fixed points which store a single pattern. The new update rule is equivalent to the attention mechanism used in transformers. This equivalence enables a characterization of the heads of transformer models. These heads perform in the first layers preferably global averaging and in higher layers partial averaging via metastable states. The new modern Hopfield network can be integrated into deep learning architectures as layers to allow the storage of and access to raw input data, intermediate results, or learned prototypes. These Hopfield layers enable new ways of deep learning, beyond fully-connected, convolutional, or recurrent networks, and provide pooling, memory, association, and attention mechanisms. We demonstrate the broad applicability of the Hopfield layers across various domains. Hopfield layers improved state-of-the-art on three out of four considered multiple instance learning problems as well as on immune repertoire classification with several hundreds of thousands of instances. On the UCI benchmark collections of small classification tasks, where deep learning methods typically struggle, Hopfield layers yielded a new state-of-the-art when compared to different machine learning methods. Finally, Hopfield layers achieved state-of-the-art on two drug design datasets. The implementation is available at: https://github.com/ml-jku/hopfield-layers

1 Introduction

The deep learning community has been looking for alternatives to recurrent neural networks (RNNs) for storing information. For example, linear memory networks use a linear autoencoder for sequences as a memory (Carta et al., 2020). Additional memories for RNNs like holographic reduced representations (Danihelka et al., 2016), tensor product representations (Schlag & Schmidhuber, 2018; Schlag et al., 2019) and classical associative memories (extended to fast weight approaches) (Schmidhuber, 1992; Ba et al., 2016a;b; Zhang & Zhou, 2017; Schlag et al., 2021) have been suggested. Most approaches to new memories are based on attention. The neural Turing machine (NTM) is equipped with an external memory and an attention process (Graves et al., 2014). Memory networks (Weston et al., 2014) use an arg max attention by first mapping a query and patterns into a space and then retrieving the pattern with the largest dot product. End to end memory networks (EMN) make this attention scheme differentiable by replacing arg max through a softmax (Sukhbaatar et al., 2015a;b). EMN with dot products became very popular and implement a key-value attention (Daniluk et al., 2017) for self-attention. An enhancement of EMN is the transformer (Vaswani et al., 2017a;b) and its


Other conclusions, for
what they’re worth.

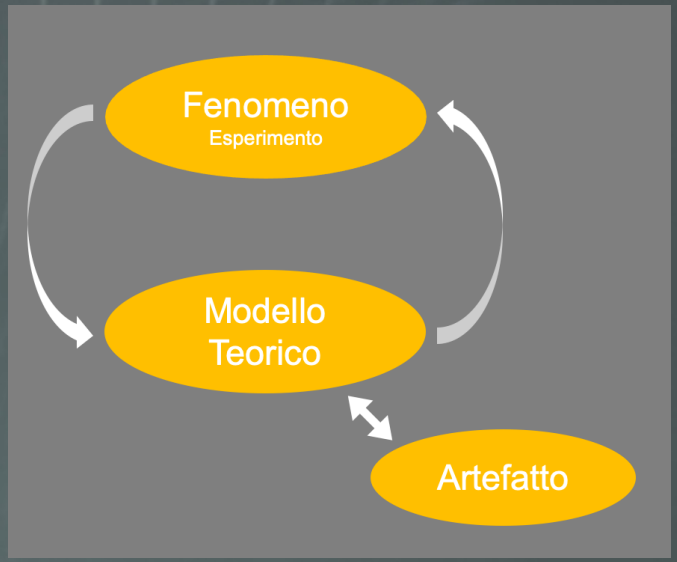


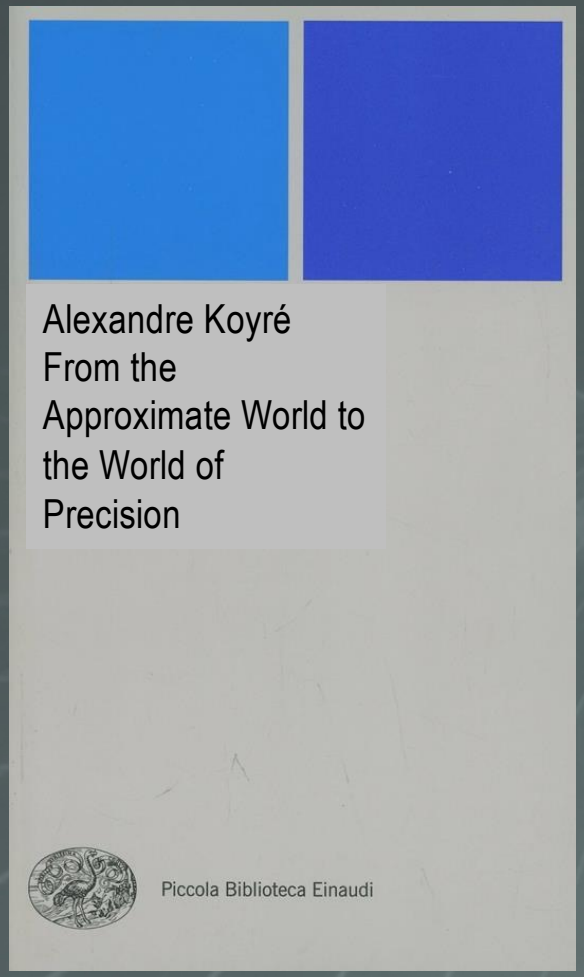
For Koyré, when he speaks of physics, he refers specifically to the modern introduction of mathematics into nature, transforming the natural world into a measurable and geometric entity.

The Galilean approach relies on precise measurement to distinguish between competing theoretical models.

$$\mathbf{E}(\mathbf{r}) = \frac{1}{4\pi\varepsilon_0}\frac{q_1}{r^2}\hat{\mathbf{r}}$$

Must be 2, not 2.0000000000 … 0000001




Alexandre Koyré
From the
Approximate World to
the World of
Precision
Piccola Biblioteca Einaudi

?


Chat Gpt
From the
World of Precision to
the Approximate
World
Piccola Biblioteca Einaudi

The Power of AI Hype: Speculative Risks, Real Politics

At the global AI summit in Bletchley Park, representatives from 28 countries, debated alleged existential risks posed by advanced artificial intelligence.

Vision of AGI (OpenAI)

Artificial General Intelligence is portrayed as a technology capable of elevating humanity — expanding economic abundance, accelerating scientific discovery, and providing universal cognitive tools that amplify human creativity and intelligence

Critical Perspective

When examined through a rigorous scientific lens, the catastrophic scenarios invoked in discussions of AGI remain closer to science fiction literature than to empirically grounded research

Closing Statement

The real danger does not lie in imaginary superintelligent machines, but in the political and economic ideologies underpinning these a-scientific approaches, and in their influence over the allocation of public research funding