

# Redes Neuronales

## U1: Introducción

---

Héctor Maravillo

## Section 1

### Introduction

---

# Artificial neural networks

Definition (Nunes et al., 2017)

**Artificial neural networks** are **computational models** inspired by the nervous system of living beings.

They have the ability to **acquire and maintain knowledge** (information based) and can be defined as a set of **processing units**, represented by artificial neurons, **interlinked** by a lot of interconnections (*artificial synapses*), implemented by vectors and matrices of synaptic weights.

# Key features

The most relevant features of artificial neural applications are:

- **Adapting from experience**
- **Learning capability**
- **Generalization capability**
- **Fault tolerance**
- **Distributed storage** neurons.

# Potential Application Areas

- Universal curve fitting (function approximation)
- Process control
- Pattern recognition (classification)
- Data clustering
- Prediction system
- Optimization
- Associative memory (image processing, signal transmission)

# Historical Overview

- In 1943, Warren McCulloch (neurophysiologist) and Walter Pitts (logician) introduced the **first mathematical model** of an artificial neuron inspired by biological neurons.  
The weights on a McCulloch-Pitts neuron are set so that the neuron performs a particular simple **logic function**, with different neurons performing different functions.

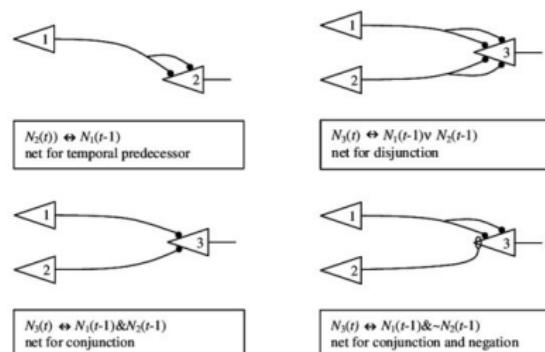
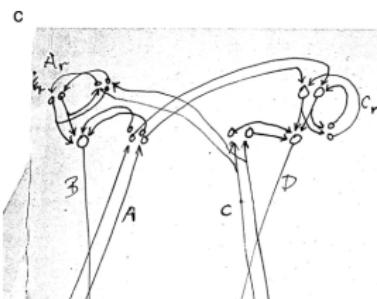
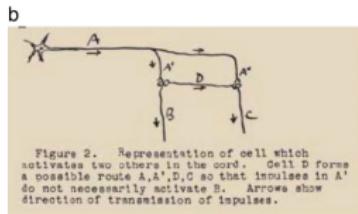
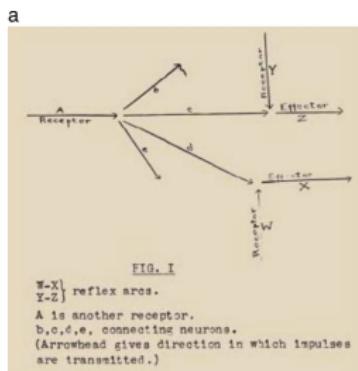


Figure: Diagrams of McCulloch and Pitts nets

- In 1949, Donald Hebb (psychologist) designed the first **learning rule** for artificial neural networks.  
He proposed that: *the strength (conductivity) of the synapse is proportional to the simultaneous activity of both connected neurons.*



Possible mechanism of reflex inhibition: on the assumption that afferents A fire B only when a supporting or reinforcing activity is going on in the reticular substance ( $A_r$  = a reinforcing) - and that the afferents C fire  $A_r$  completely. It is necessary to assume also that this sudden completion of  $A_r$  does not add to the normal level of excitation of A. In other words, the onset of inhibitory action is not heralded by a preliminary increased activation. It is also necessary to assume that the interneuronal connections of  $A$  to  $A_r$  are essential to the reinstatement of activity in  $A_r$ , and hence that  $A_r$  does become active again when  $A$  is re-activated, but not until then. Thus the cumulative loss of tonus postulated by my first paper of '34 is still essential as the matter stands at present.

Figure: Unpublished notes by Hebb

- In 1951, the students Marvin Le Minsky (mathematician) and Dean S. Edmonds (physicist) designed the **Stochastic Neural Analog Reinforcement Calculator (SNARC)** funded by the Office of Naval Research of the U.S. Department of Defense.
- The original SNARC system consisted of a randomly connected network of 40 Hebb Synapses.



Figure: The control panel for the C-1 gyroscopic autopilot used in the SNARC.

- Between 1957 and 1958, Frank Rosenblatt (psychologist) developed the first **neurocomputer** called the Mark I Perceptron, establishing the basic model of the **Perceptron**.  
The Perceptron model was capable of **recognizing simple patterns**, its learning rule uses an iterative weight adjustment that is more powerful than Hebb rule.
- In 1958, he published *The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain* where he formally presents its learning algorithm.

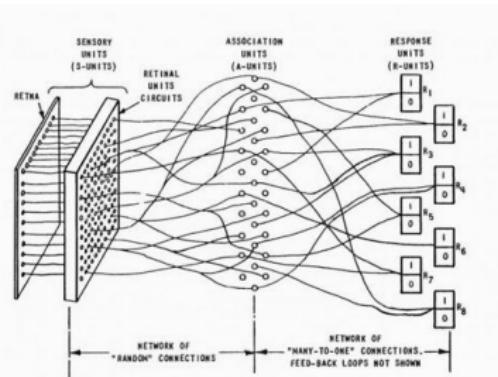
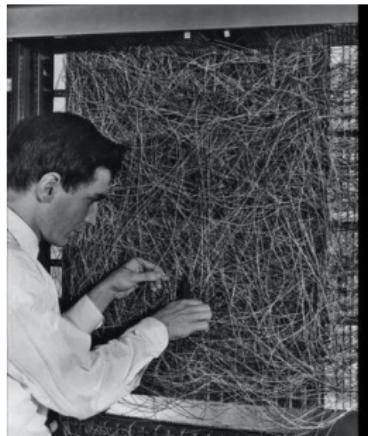


Figure 1 ORGANIZATION OF THE MARK I PERCEPTRON

- In 1960, Bernard Widrow and Ted Hoff (electrical engineers) developed a one-layer network called **ADALINE** (ADaptive LINEar Element). Later on, they proposed the MultipleADALINE (**MADALINE**).

These network used a learning rule (least mean square or **delta rule**) that adjust the weights to **reduce the difference** between the net input to the output unit and the desired output. While, the perceptron rule adjust the connection weights to a unit whenever the response of the unit is incorrect.

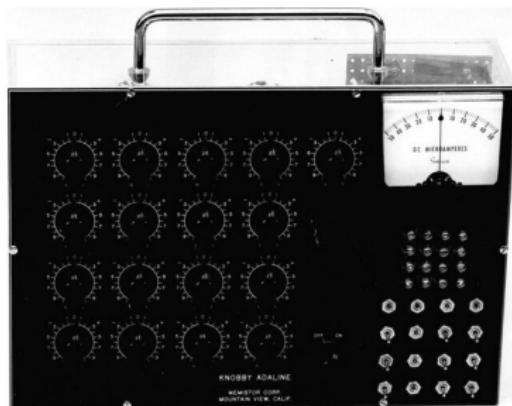
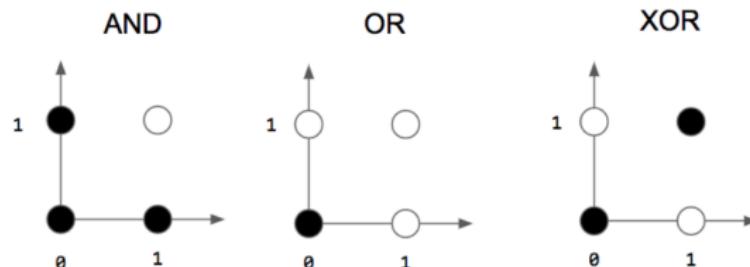


Figure: ADALINE machine, with hand-adjustable weights

- In 1962, Rosenblatt proved the **convergence** of a learning algorithm to one-layer **Perceptron**, a way to change the weights iteratively so that a desired computation was performed.
- In 1969, neurocomputing suffered a major setback with the publication of the classical book "*Perceptrons: An Introduction to Computation Geometry*" by Marvin Minsky (computer scientist) and Seymour Papert (mathematician). The authors discussed the **limitations** of the neural networks of one-layer (Perceptron and the ADALINE) and they demonstrated the impossibility of neural networks to classify patterns of **nonlinear separable classes**.
- Following the impact of that publication, researches (and **funding!**) on neural networks were greatly reduced.



- In 1975, Kunihiko Fukushima (computer scientist) developed a stepwise trained multilayered neural network for the interpretation of handwritten characters, called **Cognitron**.
- In 1979, Fukushima created the first **Convolutional Neural Network (CNN)** architecture, called neocognitron.

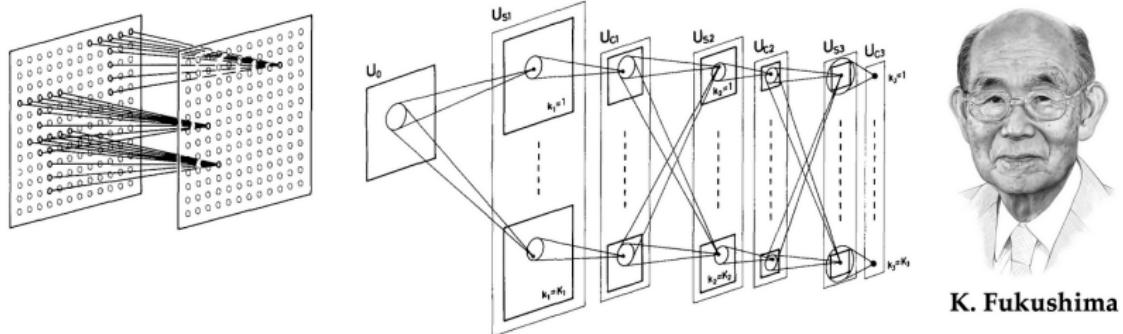


Figure: Convolutional Neural Network architecture

- 1980: The development of the **Adaptive Resonance Theory (ART)** network by Stephen Grossberg and Gail Carpenter (neuroscientists).
- 1982: The formulation of the **Self-Organized Maps (SOM)** by Teuvo Kohonen (computer scientist). These nets have applied to speech recognition and musical composition.

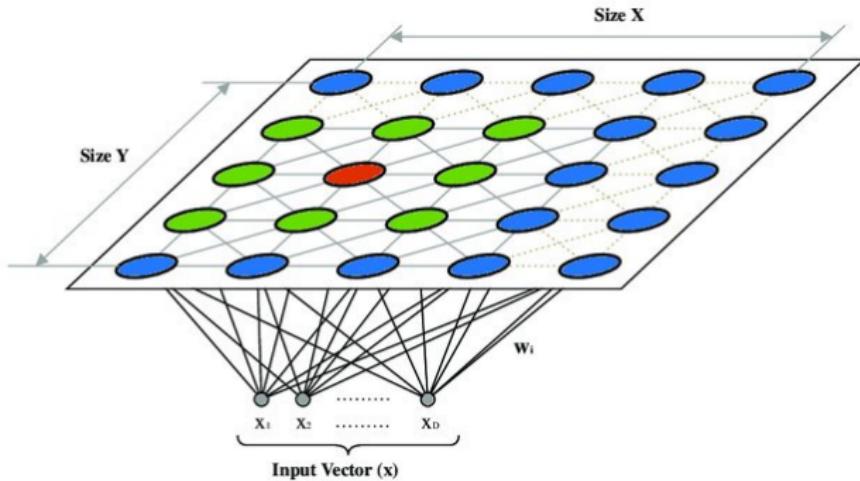


Figure: Self-organizing-map structure.

- 1982: A **recurrent network** based on energy functions proposed by John Hopfield (physicist). This work restored the prestige of artificial neural networks prior to 1969. The Hopfield network consists of only one layer whose neurons are fully interconnected with each other.

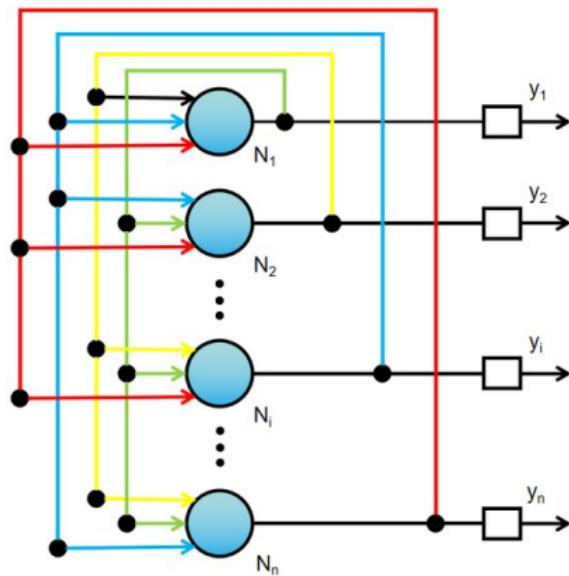


Figure: Hopfield neural network structure.

- In 1985, Geoffrey Hinton (psychologist and computer scientist) worked on a neural network approach for learning probability distribution, called **Boltzmann Machines**.  
Boltzmann Machines are networks just like neural nets and have units that are very similar to Perceptrons, but instead of computing an output based on inputs and weights, each unit in the network can compute a **probability** of it having a value of 1 or 0 given the values of connected units and weights.
- Between 1985 and 1990, **conferences and journals specializing** in neural networks were established, such as the IEEE International Conference on Neural Networks.

- In 1986, Rumelhart, Hinton, and Williams, published the book "Parallel Distributed Processing" which introduced the **Backpropagation Algorithm**. This algorithm enabled the training of **multi-layer networks**, solving the XOR problem and revitalizing interest in neural network research.
- This method was also discovered independently by David Werbos (1984), Parker (1985) and LeCun (1986)

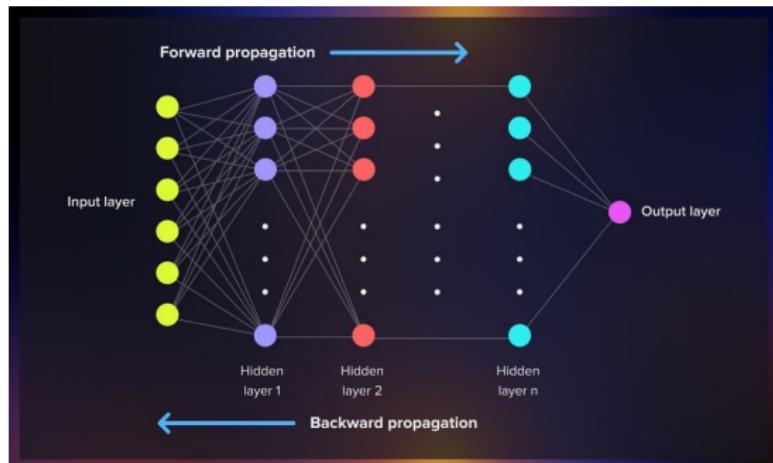


Figure: Backpropagation outline

- 1989: George Cybenko (mathematician) showed that **feedforward neural networks**, with a hidden layer and a continuous, non-polynomial sigmoidal activation function, are **universal approximators** of continuous functions on compact subsets on  $\mathbb{R}^n$ .
- 1989-1991: Hornik, Stinchcombe, and White extended Cybenko's theorem to other **non-polynomial activation functions** and more general domains.

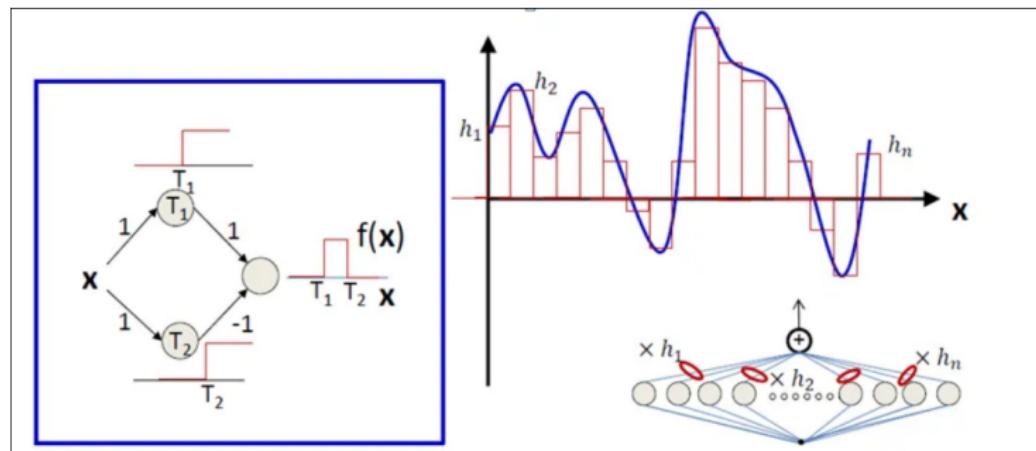


Figure: 3 unit Multi Layer Perceptron using Step function to approximate a continuous function

- In 1989, Yann LeCun (computer scientist) and his team at AT&T Bell Labs developed the first **convolutional neural network** trained with backpropagation, the **LeNet**, to characterize recognize of U.S. Postal Service.

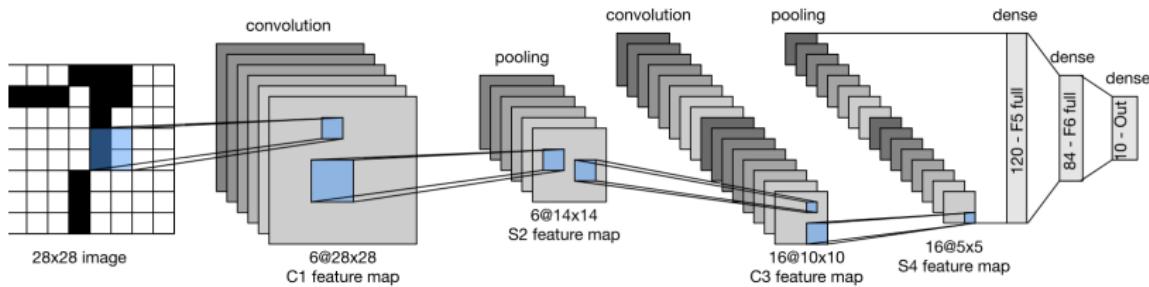


Figure: Data flow in LeNet

- In 1989, Alexander Waibel (computer scientist) et al. introduced **time-delay neural networks (TDNN)** for phoneme recognition, where each neuron processed only a subset of the input with multiple sets of weights for different delays. This approach used a '**moving window**' of audio, allowing the same audio segments to be processed by neurons with different weights based on their position in the window.

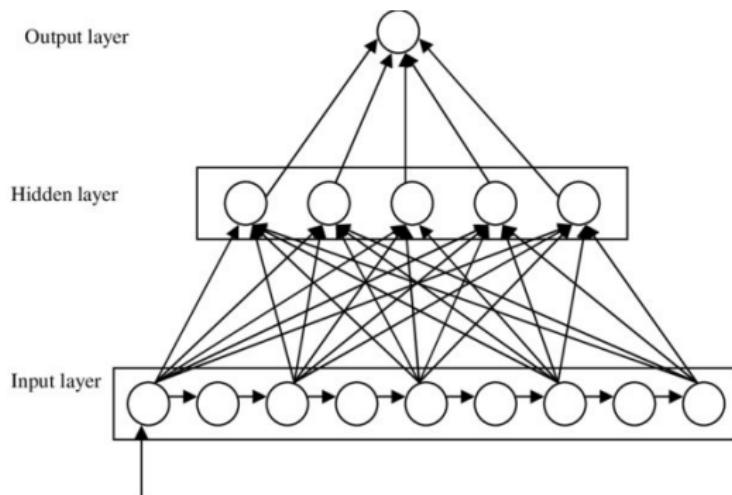


Figure: Diagram of a TDNN

- 1991: Hochreiter's Thesis identified that deep neural networks suffer from the vanishing or **exploding gradient problem**, where error signals either diminish rapidly or grow uncontrollably during backpropagation, particularly as the network depth increases.
- 1994: The learning algorithm based on **Levenberg-Marquardt method (LMA)** to solve non-linear least squares problem, which improves the efficiency of artificial neural networks (Hagan and Menhaj). It combines the advantages of **gradient descent method** and **Gauss-Newton method**.

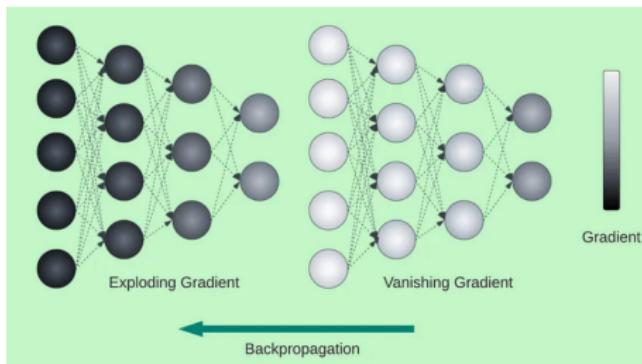


Figure: Exploding and vanishing gradient problem

- 1995: LeCun publishes **LeNet-5**, a convolutional network successfully applied to **handwritten digit recognition (MNIST)**, marking a milestone in computer vision.

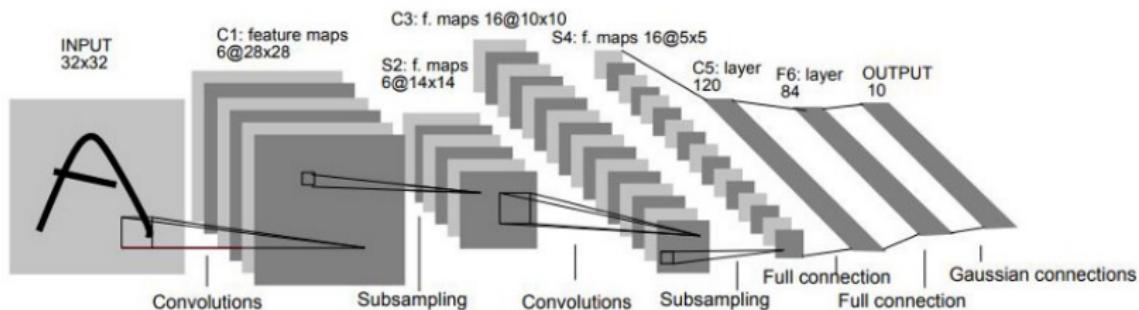


Figure: Lenet-5

- 1997: Jürgen Schmidhuber and Sepp Hochreiter (computer scientists) introduced the **Long Short-Term Memory (LSTM)** networks, which provided a solution for effectively training recurrent neural networks by managing long-term dependencies and mitigating the gradient problems.

The basic LSTM idea is used units with the identity function as activation, and has a connection to itself with fixed weight of 1.0. Due to constant derivative of 1.0, errors backpropagated through a LSTM cannot vanish or explode but stay as they are.

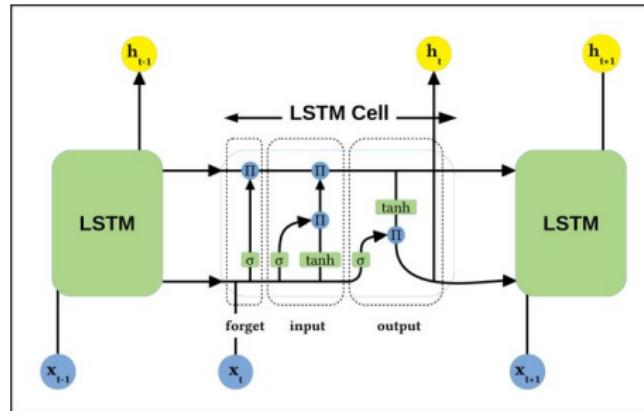


Figure: Basic Long-Short Term Memory (LSTM) architecture.

- 2003: Hinton published the seminal paper “*A Neural Probabilistic Language Model.*” This work introduced the use of neural networks for **language modeling**.
- Hinton, Osindero and Teh introduced **Deep Belief Networks**, trained layer by layer using **Restricted Boltzmann Machines (RBM)**. They partially addressed the vanishing gradient problem in deep networks and rekindled interest in **training deep architectures**.
- 2009: Hinton, Deng, Yu and colleagues applied **Deep Neural Networks** to **automatic speech recognition**, replacing traditional Gaussian Mixture Models - Hidden Markov Models systems. DNNs achieved significant reductions in word error rate on benchmark tasks such as **TIMIT** and **Switchboard**.

- For several years, interest and funding in neural networks declined, while attention shifted to more efficient and effective learning methods, such as SVMs and Random Forests.
- 2011: Some authors begin to analyze **Quantum neural network (QNN)**.

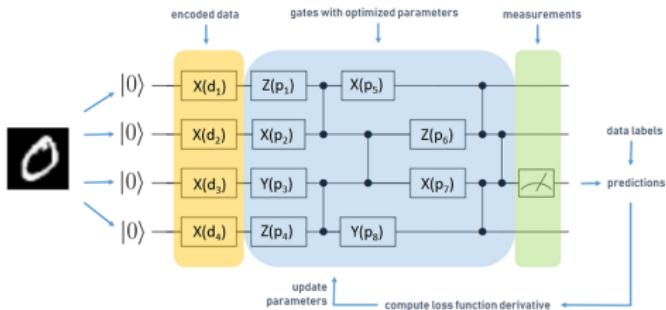


Figure: Simplified QNN for a classification task with 4 qubits.

- Krizhevsky, Sutskever ahnd Hinton (team SuperVision) won the **ImageNet competition** using a convolutional neural network, called **AlexNet** with 8 learned layers ( 60 million parameters), achieving a top-5 error of 15% and surpassing all previous AI-based image classifiers.

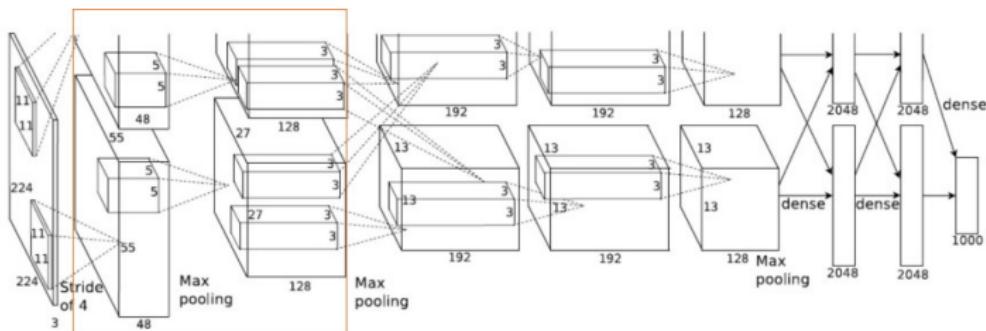


Figure: AlexNet architecture.

- 2013: Mikolov, Chen, Corrado, Sutskever and Dean at Google developed **Word2Vec** a method for learning dense **vector representations of words** using shallow, two-layer neural networks, enabling semantic similarity to be captured in the vector space.
- 2014: Sutskever, Vinyals, and Le at Google introduced **Seq2Seq**, an architecture based on **recurrent neural networks (LSTMs)** with an **encoder–decoder structure** for mapping input sequences to output sequences, enabling major advances in tasks such as machine translation, image captioning, and speech recognition.

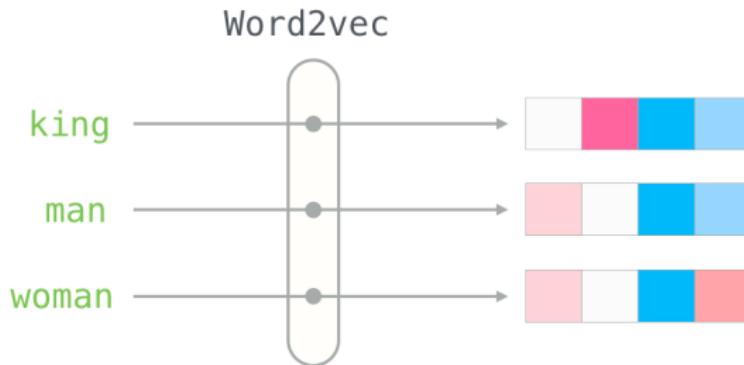


Figure: word2vec example.

- In 2014, Ian Goodfellow introduced **Generative Adversarial Networks (GANs)**, an unsupervised machine learning algorithm featuring two **competing** neural networks: a generative model that creates content and a discriminative model that assesses whether the content is authentic.



Figure: Example of images produced by GAN

- In 2017, Google introduced the **Transformer**, a **deep learning architecture**. The text is converted into numerical representations called tokens, and each token is converted into a vector by looking up a table of word representations. At each layer, each token is contextualized within the scope of the context window with other tokens: the signal for key tokens to be amplified and less important tokens to be de-emphasized.



Figure: Eight Google scientists who revolutionized AI with the 'Transformer' model

# Overview

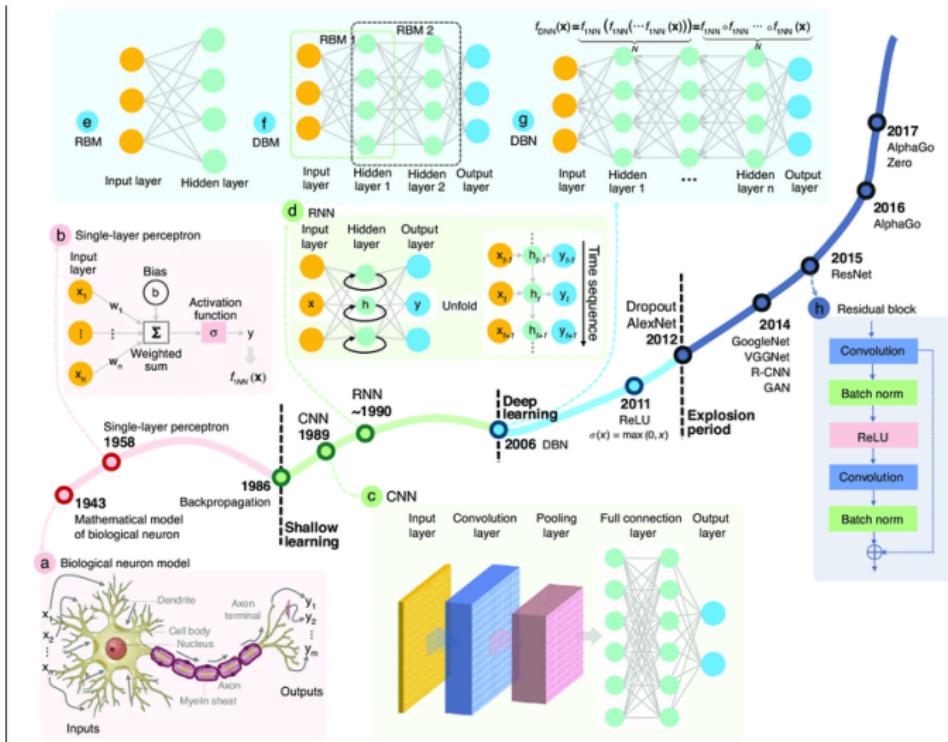


Figure: Evolution of neural networks

# Network Architecture

- **Feed-Forward** artificial neural network scheme
  - Data moves strictly from **input to output** units in a feed-forward manner.
  - Data processing may involve **multiple layers**, but there are no feedback connections.
  - Examples include **Perceptron** (Rosenblatt) and **Adaline** (Adaptive Linear Neuron).
- **Recurrent** artificial neural network scheme
  - These networks incorporate **feedback connections**.
  - Examples include Kohonen **Self-Organizing Maps (SOM)** and **Hopfield networks**.

# Biological Neuron

The fundamental cell of the central nervous system is the **neuron**, and its role come down to **conduct impulses** (electrical stimuli originated from physical-chemical reactions) under certain operation conditions.

The three main parts of a cell are: **dendrites**, **cell body (soma)** and **axon**.

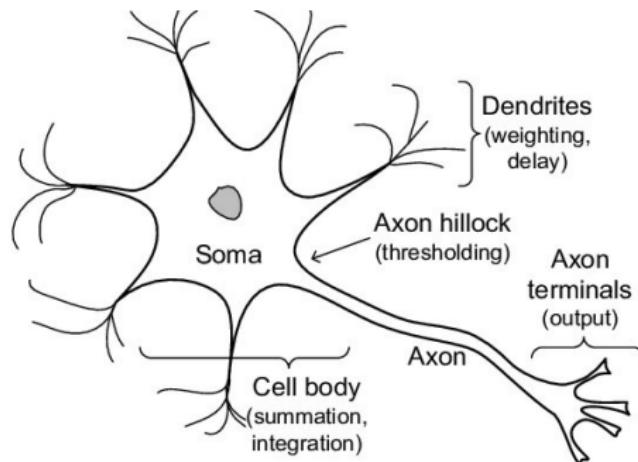


Figure: Diagram of a biological neuron

## Parts of a neuron

- **Dendrites** are composed of several thin extensions that form the dendritic tree. The fundamental purpose of dendrites is to **acquire, continuously, stimuli** from several other neurons (connectors) or from the external environment.
- The **soma** is responsible for **processing** all the **information** that comes from the dendrites, to produce an **activation potential** that indicates if the neuron can trigger an electric impulse along its axon.
- The **axon** is composed of a single extension whose mission is to guide the **electrical impulses** to other connecting neuron. The axon termination is composed of branches called **synaptic terminals**.

# Synapses

- The **synapses** are the connections which enable the transfer of electric axon impulses from a particular neuron to dendrites of other neurons.
- There is no physical contact between the neurons, so the **neurotransmitter elements** released on the junction are in charge of weighting the transmission from one neuron to another

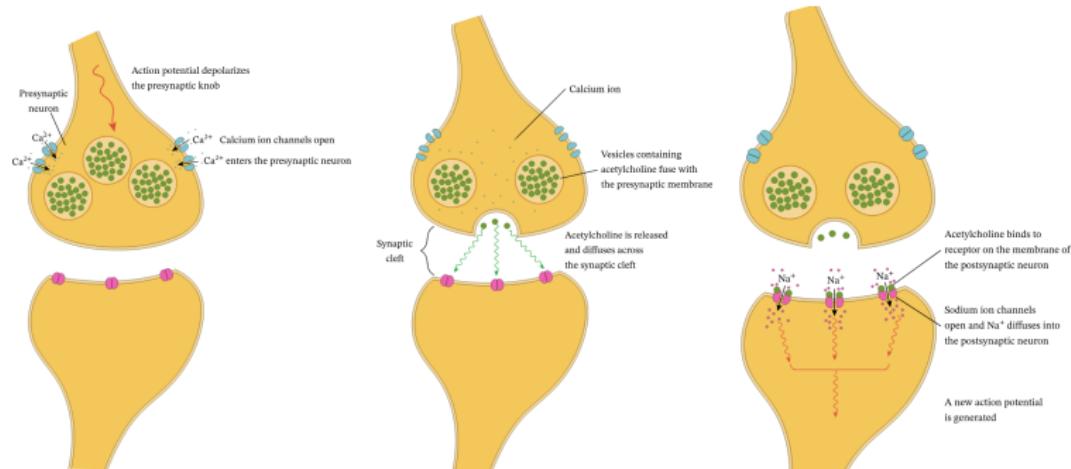


Figure: Synapsis process outline

# Mechanism of Neural Cell Communication

- The **neural membrane** action potential has negative values when in rest (**polarized**), meaning there is a larger concentration of negative ions inside the membrane than at its exterior.
- When the nervous cell is **stimulated** (depolarized) with an impulse higher than its **activation threshold** (-55mV), caused by the variation of internal concentrations of sodium and potassium ions, it initiates an electrical impulse which will propagate throughout its axon with a maximum amplitude of 35mV (millivolts).
- The maximum value of the action voltage (35mV) is fixed for all neurons, however, the **signal duration** in time is variable.
- As soon as the excitation process ends, the membrane will be consequently **repolarized** meaning the action voltage will return to its rest voltage (-70mV).

# Mechanism of Neural Cell Communication

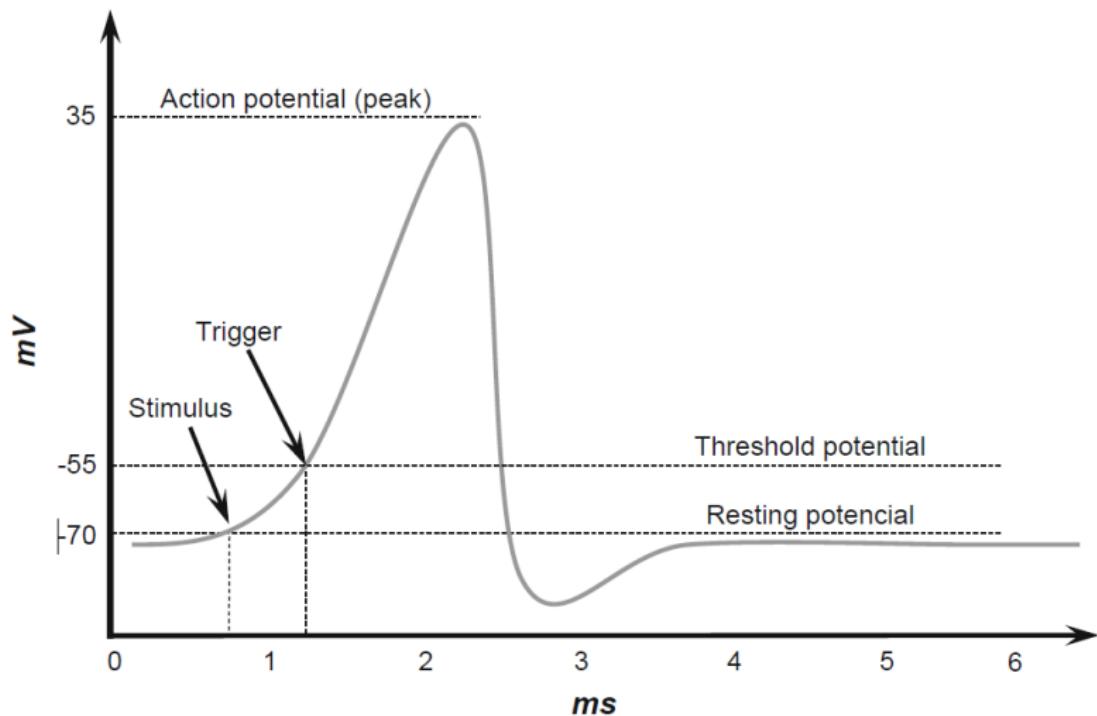


Figure: Stages of the action potential

# Plasticity

The **plasticity** permits the developing nervous system to adapt to its surrounding environment.

In an adult brain, plasticity may be accounted for by three mechanisms:

- The **creation** of new synaptic **connections** between neurons.
- The **modification** of the strength or efficiency of existing synaptic connection.
- The combinations of different amino acids enable practically an unlimited number of different proteins, allowing for an incredible **diversity** in structure and function, which is essential for the complexity and adaptability of neuron system.

# References

- Cowan, J. & Sharp, D. Neural Nets and Artificial Intelligence. *Daedalus*, 117(1): 85–121.
- Fausset, L. *Fundamentals of Neural Networks. Architectures, Algorithms and Applications*, Pearson Education, 1994.
- Nunes da Silva, I., Hernane D., Andrade, R., Bartocci, L., and dos Reis, S. *Artificial Neural Networks. A Practical Course*, Springer, 2017.