SC4001/4042 Overview

Neural Networks and Deep Learning

Assistant Professor
Ying Wei

Biological neural networks

Artificial neural networks are inspired by the biological neural networks in the brain. The three pounds of jelly-like material found within our brain is the most complex machine on earth and perhaps in the universe.

It consists of a **densely interconnected** set of nerve cells, or basic information-processing units, called **neurons**. Human brain incorporates nearly 10 billion neurons, each connected to about 10,000 other neurons with 60-100 trillion **connections**, synapses, between them.

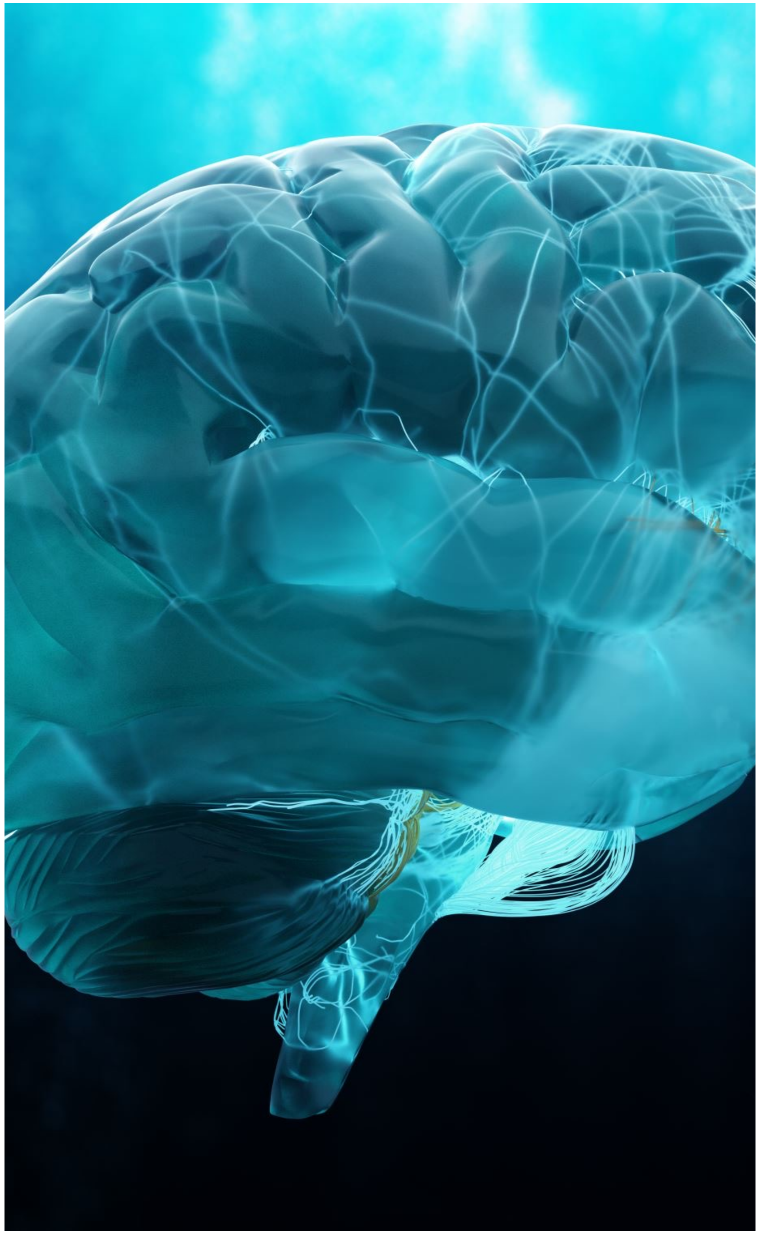
By using multiple neurons simultaneously, the brain performs its functions much faster than the fastest computers in existence today. An **artificial neural network** is defined as a model of reasoning based on the principles of the human brain.

Brain vs computer

- Typical **operating speeds** of biological neurons is in milliseconds (10^{-3} s), while silicon chip operate in nanoseconds (10^{-9} s). But Brain makes up (for slower rate of operation of a neuron) by having significant number of neurons with **massive interconnections** between them.

- Human brain is extremely **energy efficient**, using approximately 10^{-16} joules per operation per second, whereas the computers use around 10^{-6} joules per operation per second.

- Brains have an **evolution** history of tens of millions of years, computers have been evolving for tens of decades.



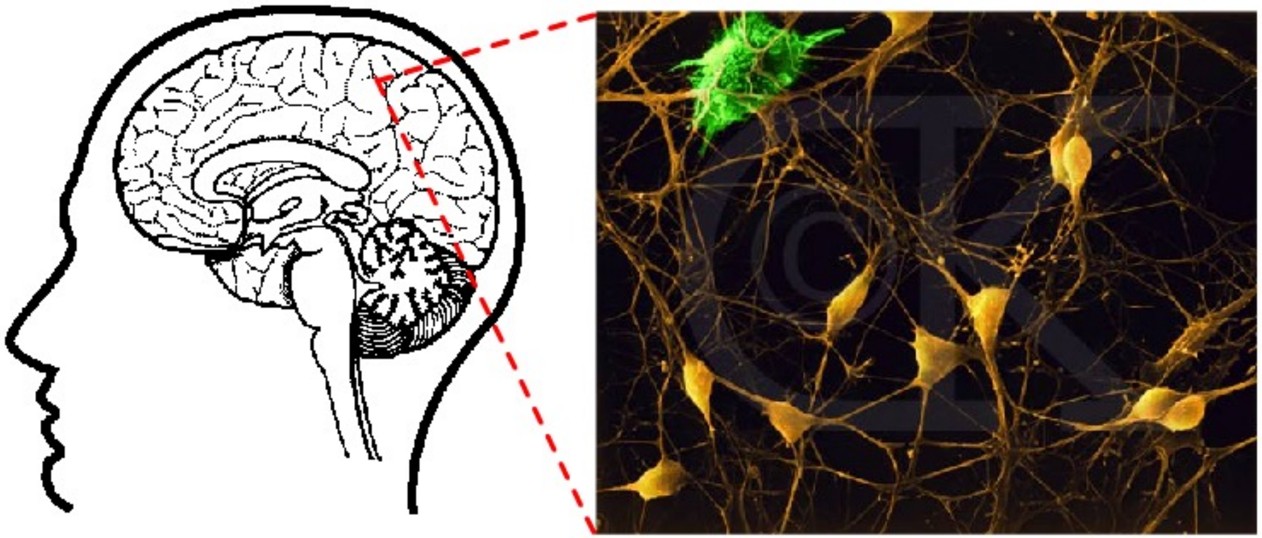
Information processing in the brain

Our brain is highly complex, **non-linear parallel information-processing** system.

Information is distributed throughout the whole network, rather than at specific locations and stored and processed in a neural network simultaneously. Today's computers have one or several processors but each neuron in the brain can be considered as a simple processor.

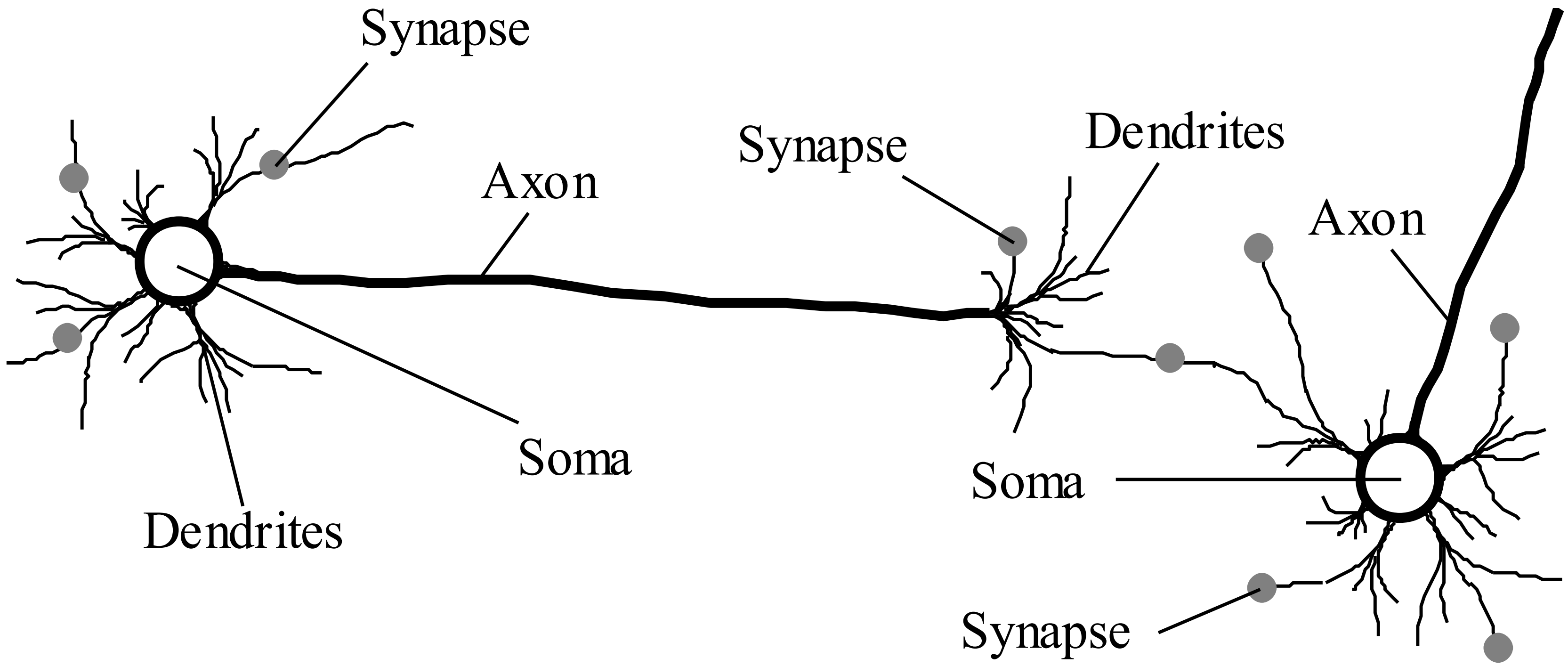
Learning is a fundamental and essential characteristic of biological neural networks. The ease with which they can learn, led to attempts to emulate biological neural networks in a computer.

Biological neural networks



Each of the yellow blobs in the picture above are neuronal cell bodies (soma), and the lines are the input and output channels (dendrites and axons) which connect them.

Schematic of biological neuron



Components of biological neurons

A **biological neuron** consists of the following components:

- ***Soma***: Cell body which processes incoming activation signals and converts input into output activations. The nucleus of soma contains the genetic material in the form of DNA.
- ***Axon***: Transmission lines that send activation signals to other neurons
- ***Dendrites***: Receptive zones that receive activation signals from other neurons
- ***Synapses***: Allow weighted signal transmission between the dendrites and axons. Process of transmission is by diffusion of chemicals.

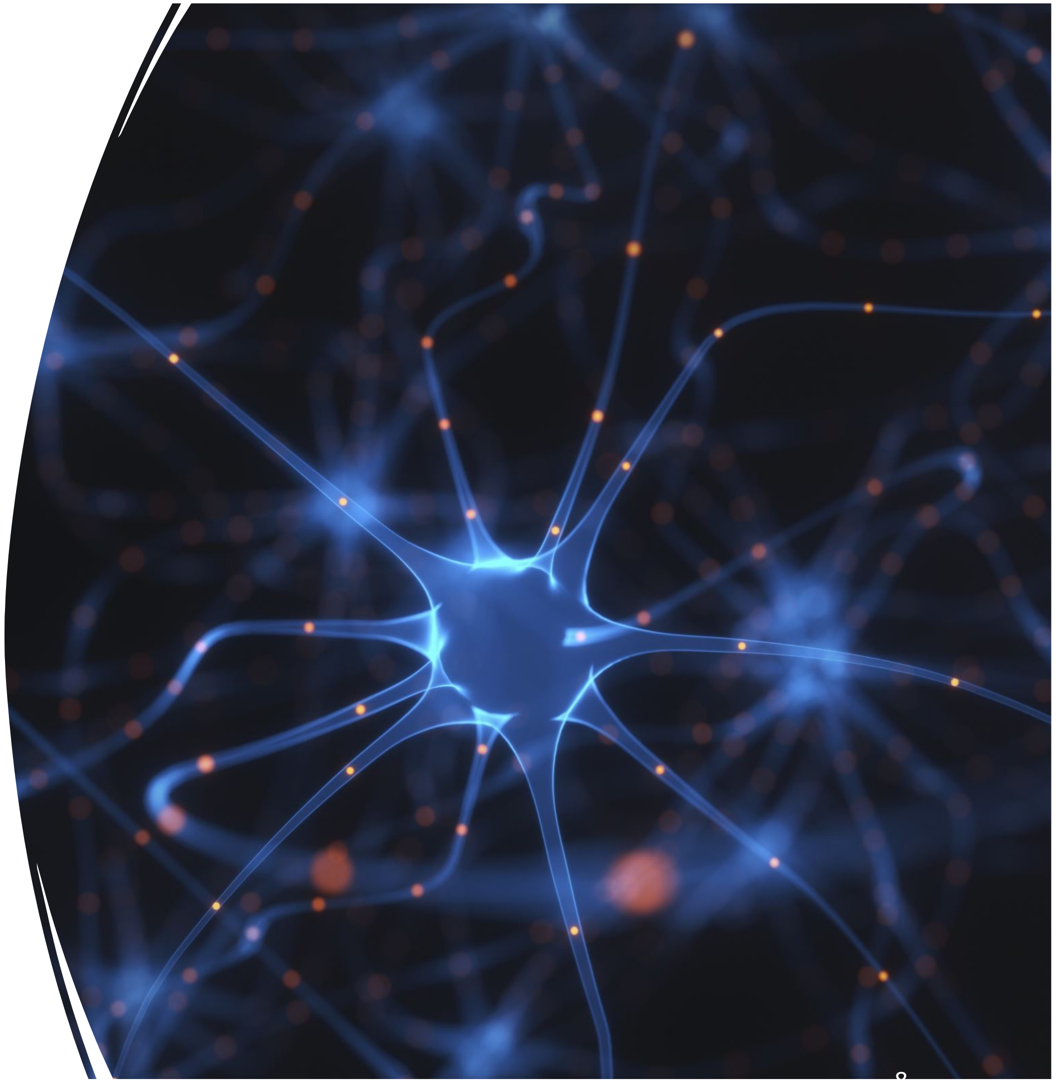
Although neuronal cell body performs majority of cells function, most of the cells total volume is taken up by axons (about 90%).

Signal flow of biological neurons

Each neuron receives electrochemical inputs **(neurotransmitters)** from other neurons at the dendrites.

Neuroreceptors receive such signals and **soma** sums the incoming signals. If the sum of these electrical inputs is sufficiently powerful to activate the neuron, it **transmits an electrochemical signal** along the **axon**, and passes this signal to the other neurons whose **dendrites** are attached at any of the **axon** terminals.

These attached neurons when its internal voltage exceeds a certain **threshold** (-70mv) may then fire. Signals (**action potentials**) flow along the axon as a form of electric pulses and make synapses on other neurons. Note that synapses could be either excitatory or inhibitory.



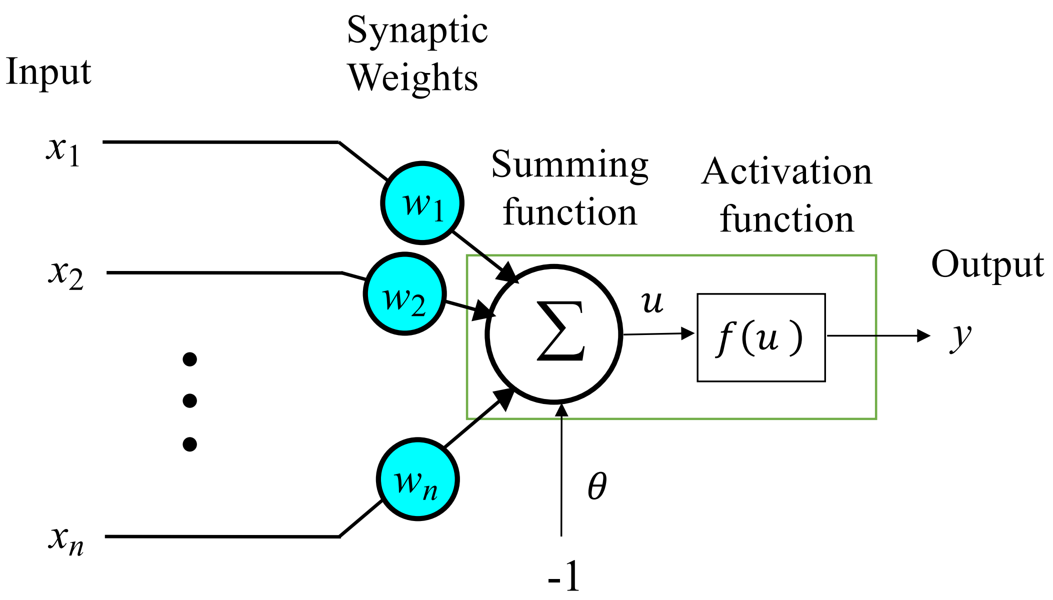
Artificial neural networks

Artificial neural networks attempt to mimic biological neural networks in the brain. The neuronal signals flow along the axon in the form of electric pulses or **action potentials**.

There are two types of artificial neural networks. One that emulates the action potentials are referred to as **spiking neural networks** and the others that emulate the aggregate of action potentials are rate-based or **activation-based neural networks**.

Neural networks discussed in this class are activation-based. However, spiking neural networks are more amenable for hardware implementations.

Artificial neuron model



Input $\boldsymbol{x} = (x_1 \quad x_2 \quad \cdots \quad x_n)^T$

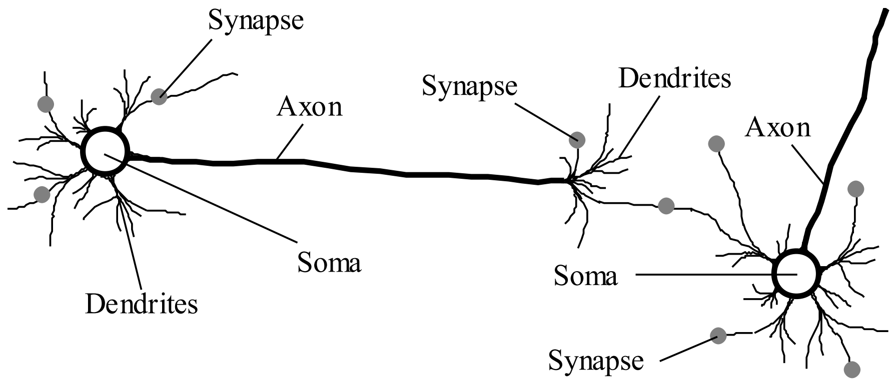
u – total synaptic input

f – activation function

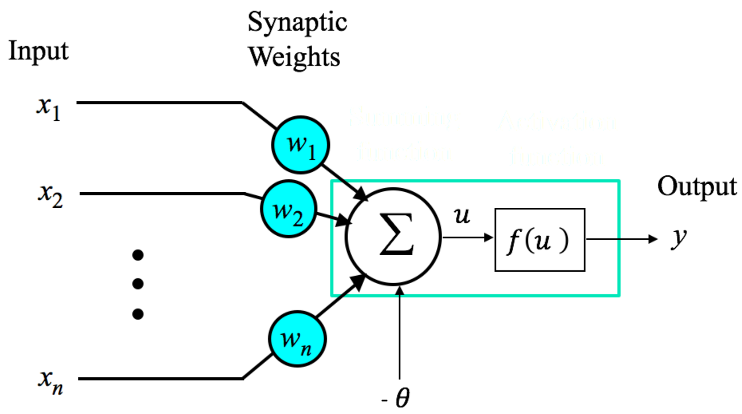
y - output

Analogy between biological and artificial neurons

Biological neuron

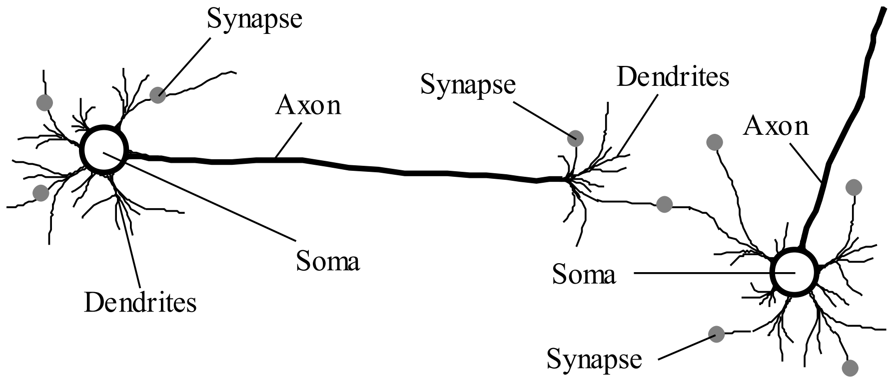


Artificial neuron

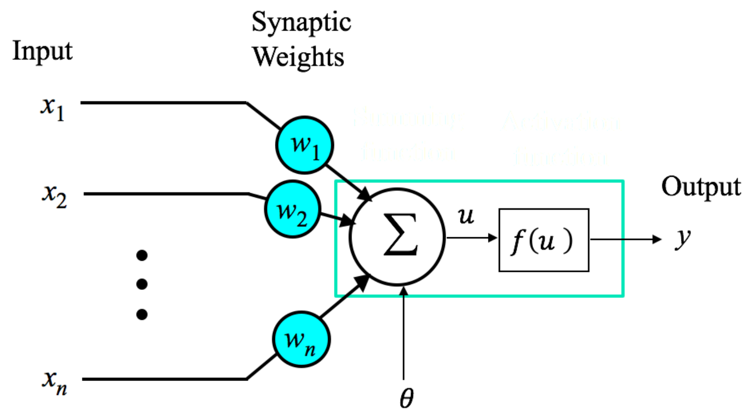


Analogy between biological and artificial neurons

Biological Neuron	Artificial Neuron
Soma	Sum + Activation function
Dendrite	Input
Axon	Output
Synapse	Weight



- *McCulloch-Pitts neuron* (~ 1940) is an artificial neuron with binary inputs and outputs
- *Perceptron* (~ 1950) is another name for an artificial neuron with analog inputs



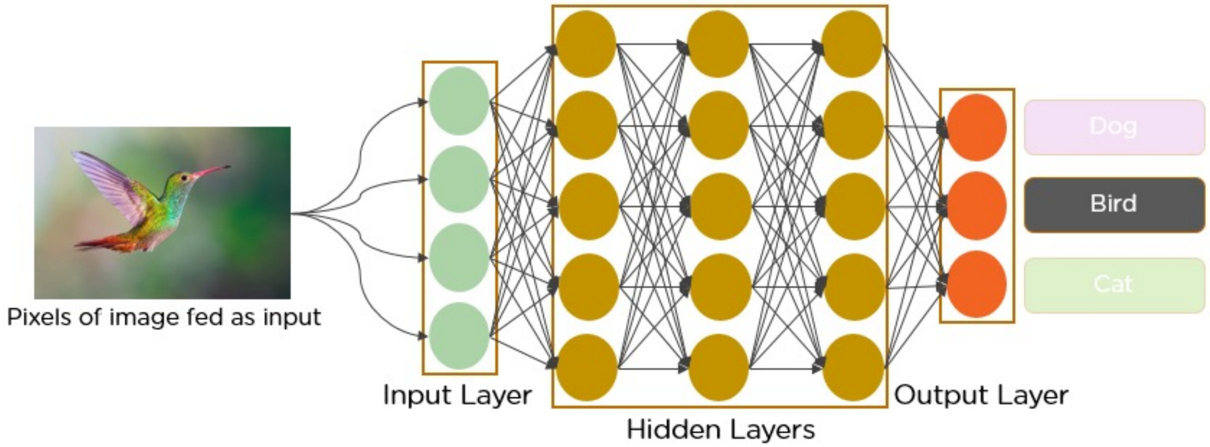
feedforward neural networks

- A **three-layer network** (two hidden layers and output layer).

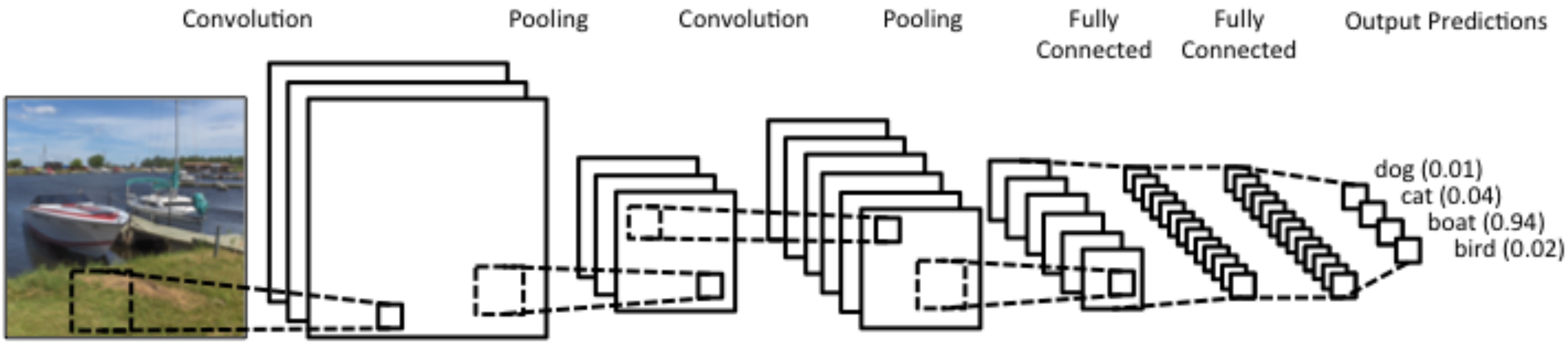
- The **input layer** consists of input nodes that receive input signals.

- The layers between input layer and **output layer** are referred to as **hidden layers**.

- If the **depth** (the number of layers) is large, feed forward neural networks are referred to as **deep neural networks.**



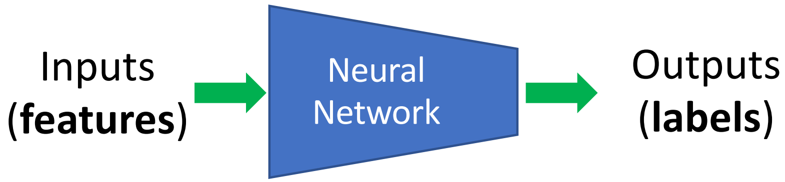
Deep convolutional neural networks



Alternate layers of convolutional and pooling, followed by fully connected layers.

Predictive analytics with neural networks

Neural network is a computational paradigm for machine learning and data analytics.



Neural networks are to be **trained** first with **training data.** Once trained, neural networks are able to **predict** labels for new data.

Predictive analytics:
Regression*:* outputs are continuous variables: age, height, income, etc.
Classification: outputs are discrete variables: sex, type of flowers, digits, etc.

History

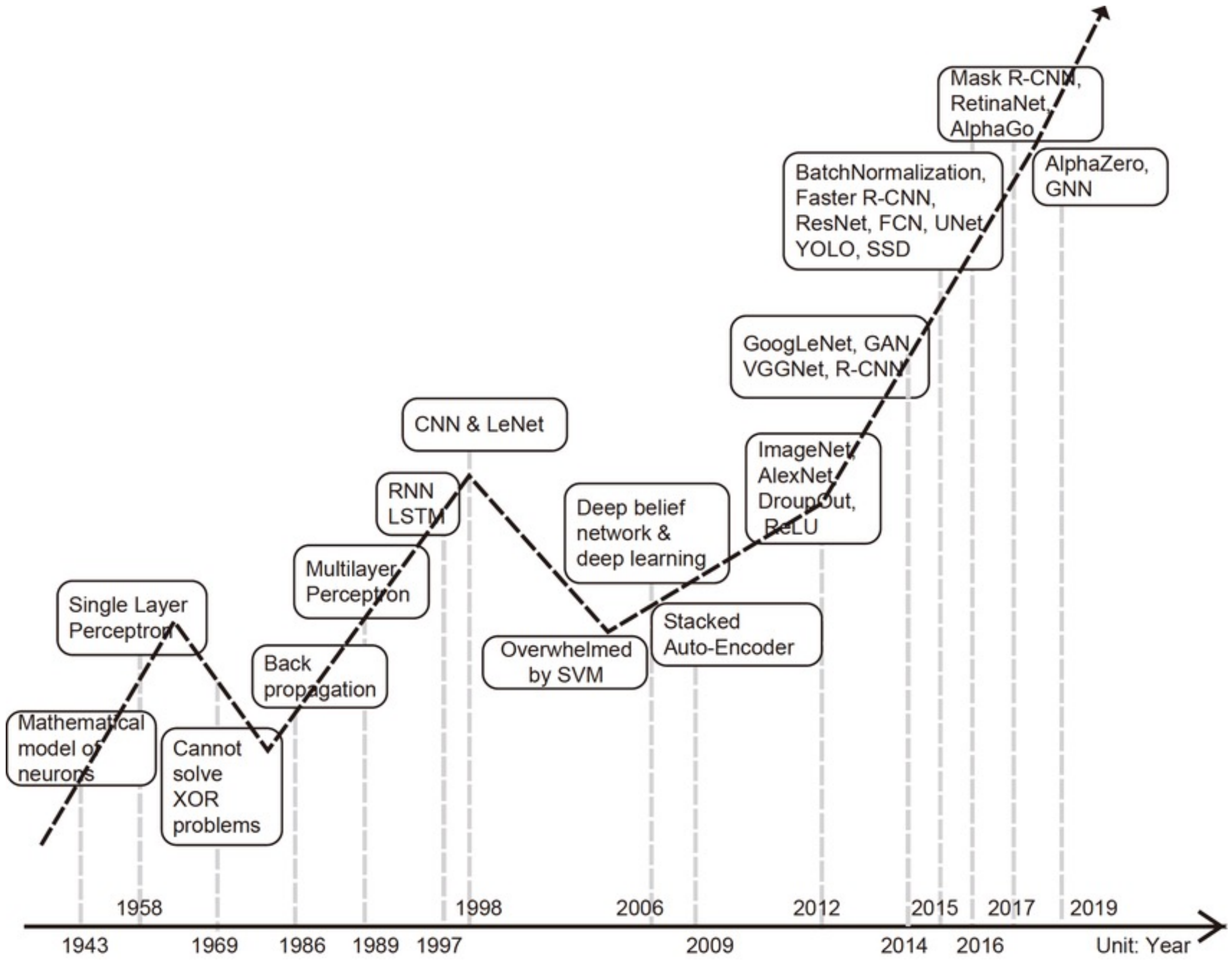
Modern view of Neural Networks (NN) began in the **1940s**. – **Warren McCulloch, Walter Pitts, Donald Hebb**.

By **late Sixties**, most of the basic ideas and concepts necessary for neural computing had already been formulated: **perceptron**, **gradient descent learning**, etc.

Practical solution emerges for neural networks only in the mid-eighties; for example, **backpropagation algorithm** (**1985**).

Major reason for the delay in using large neural networks was technological: no powerful workstations to model and experiment with ANN; algorithms for learning large neural networks were unknown.

Emergence of **deep learning** since **2012**: human like performance was achieved in object recognition, using deep convolutional neural networks such as AlexNet.



Mathematical model of neurons
Single Layer Perceptron
Cannot solve XOR problems
Back propagation
Multilayer Perceptron
RNN LSTM
CNN & LeNet
Overwhelmed by SVM
Deep belief network & deep learning
Stacked Auto-Encoder
ImageNet, AlexNet, DroupOut, ReLU
GoogLeNet, GAN, VGGNet, R-CNN
BatchNormalization, Faster R-CNN, ResNet, FCN, UNet, YOLO, SSD
Mask R-CNN, RetinaNet, AlphaGo
AlphaZero, GNN
1943
1958
1969
1986
1989
1997
1998
2006
2009
2012
2014
2015
2016
2017
2019
Unit: Year

SC4001 Learning Objectives

1. Interpret artificial neuron as an abstraction of biological neuron and explain how it can be used to build deep neural networks that are trained to perform various tasks such as regression and classification

2. Identify the underlying principles, architectures, and learning algorithms of various types of neural networks;

3. Select and design a suitable neural network for a given application;

4. Implement deep neural networks that can efficiently run on computing machines.

Pre-requisites: MH1810, SC1004, SC1003, SC1007

Comfortable with some Mathematics. Linear Algebra. Basic Calculus.

Need programming skills

Course Hours

- **Lectures**: Thursday 3:30pm – 5:30pm (LT19A)
- **Tutorial**: Wednesday 4:30pm – 5:30pm (LT1),

- *Tutorials start from 3rd week*

Course Topics

First Half:

1. NN fundamentals
2. Regression and Classification
3. Neuron layers
4. Deep neural networks (DNN)
5. Model selection and overfitting
6. Convolution neural networks (CNN)

Second Half:

7. Convolution neural networks (CNN) architectures
8. Recurrent neural networks (RNN) and Gated RNN
9. Attention
10. Autoencoders
11. Generative adversarial networks (GAN)

Python and PyTorch

- **Python >=3.11** is the programming language
- **Pytorch >=2.0** Libraries:
  - PyTorch: https://pytorch.org/
  - Codes of lecture examples and tutorials will be provided.
- Codes are provided as **Jupyter Notebook** (.ipynb) files.

Assessment

- **Programming Assignment (25%):**

  Individual: handout Feb 16, deadline Mar 15

- **Project (35%):**

  Group of up to 3 students: handout Mar 15, deadline Apr 12

- **Final exam (40%):**

  Open book

For the Assignment and the project, codes and a report are to be submitted to NTULearn by the deadline. Late submissions will be penalized!

Group project

Project (35%) – Group (up to three)

- Project ideas handout: Mar 15
- Deadline: Apr 12

The students are to propose the project and form project groups. The topic could also be selected from given project ideas.

The project includes potential research issue related to neural networks theory/application, literature survey, and design and implementation of a potential solution. Comparisons with existing solutions are to be performed.

A report, codes, and a video presentation are to be submitted to NTU Learn by the deadline by one of the group members. The project report should contain the names of all the project members. No need to inform prior to report submission.

Assignments and projects

- Python and Pytorch are recommended for assignments and projects
- **PC with at least 1 GPU** is recommended
- Access to **SCSE GPU Cluster** (GPU-TC) server for those who needs computational power. Students will have accounts after add-and-drop period is over. Email: scsegpu-tc@ntu.edu.sg
- Reports are to be submitted in pdf format and codes are to be submitted in a .zip file to NTU Learn before the deadline
- Late submissions are penalized (each day at 5% up to 3 days)
- Assessment criteria are indicated in the handout.

Text and References

Text (for additional reading)
Deep Learning , I. Goodfellow, Y. Bengio, and A. Courville, MIT Press, 2016
http://www.deeplearningbook.org/

References
http://deeplearning.net/tutorial/
http://deeplearning.stanford.edu/wiki/index.php/UFLDL_Tutorial

PyTorch:
https://pytorch.org/

TensorFlow, Keras
https://www.tensorflow.org/
https://keras.io/

Lecturers

- Assistant Professor Wei Ying (First Half)
- Assistant Professor Pan Xingang (Second Half)

Instructors (First Half)

Professor Wei Ying
Office: Block N4, Room 2c-117
Telephone number: (65) 67904288
E-mail: ying.wei@ntu.edu.sg
Web: https://wei-ying.net/
Office Hours: Wednesdays 5:30 – 6:30pm

TAs: Mr Liao Chang
Email: TBD

Thank you.