# CS 103 -15
# Knowledge and Deep Learning

Jimmy Liu 刘江

2020-12-25

# All The Best to CS 103 Students

# Group Project Update

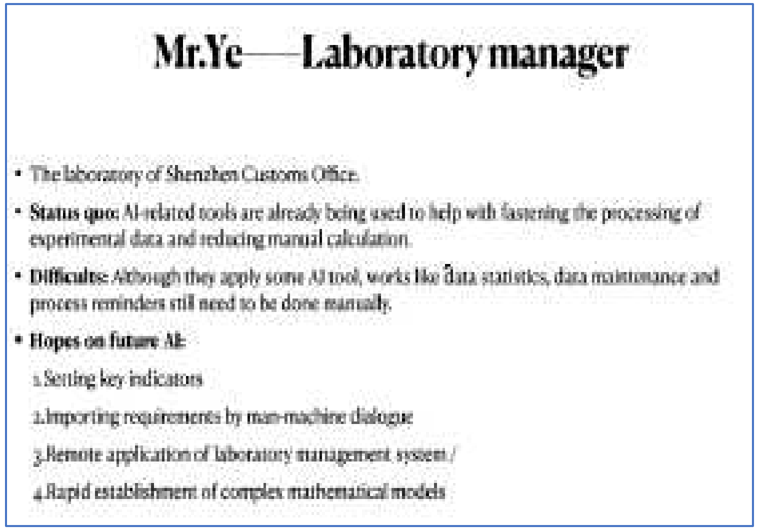
# 课程项目汇报时间安排

| 小组编号 | 时间 | 主题 | 成员 |
|---|---|---|---|
| 3 | 十三周（上午） | High Score Gamer | 易辰朗、许天淇、黄北辰（组长）、赵思源、朱佳伟、宛清源 |
| 4 | 十三周（上午） | AI application on diabetes | 周钰奇、李仪轩、董叔文、湛掌、胡钧淇（组长）、裴鸿婧 |
| 10 | 十三周（上午） | 人工智能对白内障分级的算法综述 | 赵宇航、徐格蕾、陈星宇、祖博瀛、黄弋骞（组长） |
| 22 | 十三周（上午） | 深度学习在自动驾驶中的应用 | 王晓轩 |
| 13 | 十四周（上午） | AI in Lab | 孙含曦、于松琦、**罗西（组长）** |
| 14 | 十四周（上午） | 人脸识别算法的发展与应用 | **易翔（组长）**、陈俊滔、罗景南、胡泰玮、文颖潼、吴杰翰 |
| 17 | 十四周（上午） | AI虚拟主播制作计划 | 王标、**张倚凡（组长）**、李康欣、何泽安、曾宇祺、Zhang Kenneth |
| 19 | 十四周（上午） | 校园巴士路线优化 | 王祥辰、何鸿杰、吴子彧、**樊青远（组长）**、方琪涵、袁通 |

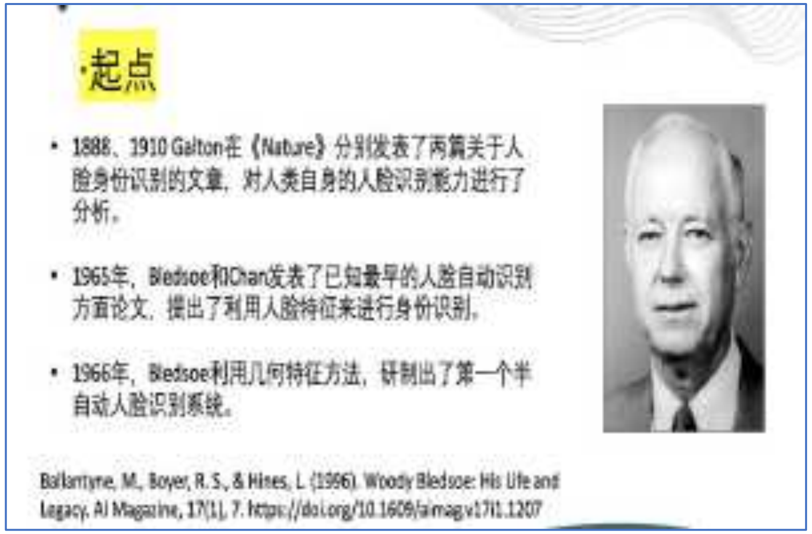
# 课程项目汇报时间安排

| 小组编号 | 时间 | 主题 | 成员 |
|---|---|---|---|
| 1 | 十三周（下午） | AI+斗地主 | 孙永康、李怀武、胡鸿飞、吴一凡、杨光、张习之、金肇轩（组长）、于佳宁 |
| 5 | 十三周（下午） | AI in lung cancer | 夏瑞浩、李悦明、龚颖璇、吴云潇潇（组长）、姜欣瑜、王英豪 |
| 6 | 十三周（下午） | 基于MRI图像的阿尔茨海默症分类 | 董廷臻、郑英炜（组长）、李博翱、朱嘉楠、李杨燊 |
| 7 | 十三周（下午） | AI Applications in Breast Cancer Imaging | 林文心、翟靖蕾（组长）、孙瀛、林宝月、陈帅名、冀鹏宇 |
| 8 | 十三周（下午） | Applications of artificial intelligence in covid-19 patients | 罗岁岁（组长）、周雅雯、肖雨馨、程旸、尹子宜 |
| 9 | 十三周（下午） | 基于OCT图像的眼部多种疾病诊断和分析的调研 | 何忧、郭煜煊、朱寒旭、赵子璇（组长）、王子杰、张晓新 |
| 2 | 十三周（下午） | AI+五子棋 | 周贤玮、韩梓辰（组长）、赵云龙、张坤龙、夏星晨 |
| 11 | 十四周（下午） | 句子图片的文本情感分析 | 唐云龙、刘叶充、**刘旭坤（组长）**、马卓远、陈子蔚、江欣乐、陈浩然 |
| 12 | 十四周（上午） | gesture recognition | 车文心、张静远、张骥霄（组长）、杜鹏辉 |
| 15 | 十四周（下午） | 人工智能在无障碍设施领域中的使用调查 | **马子晗（组长）**、陈沐尧、林小璐、任艺伟、王增义 |
| 16 | 十四周（下午） | identification of handwriting elements | 刘通、**谈思序（组长）**、赵伯航、张皓淇 |
| 18 | 十四周（下午） | 人工智能技术在个性化推荐系统上的应用与研究 | 谭雅静、刘思岑、Ooi Yee Jing、孟宇阳、**杨锦涛（组长）** |
| 20 | 十四周（下午） | 给线稿上色的强大AI的算法研究 | **韩晗（组长）**、刘思语、赵晓蕾、陈松斌 |
| 21 | 十四周（下午） | 人工智能在皮肤癌诊断领域的可能性探索 | 刘宇欣、**李修治（组长）**、沈睿琦 |

# Group 13: AI in Lab

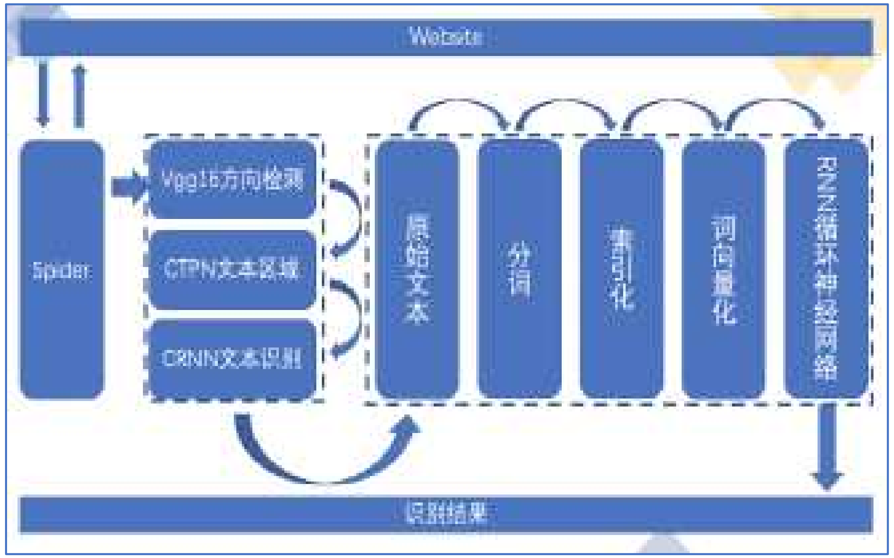

# Group 14: 人脸识别算法的发展与应用

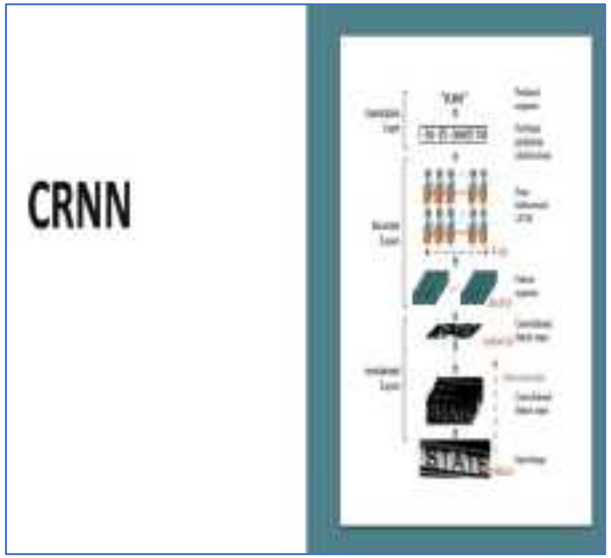




# Group 17: AI虚拟主播制作计划

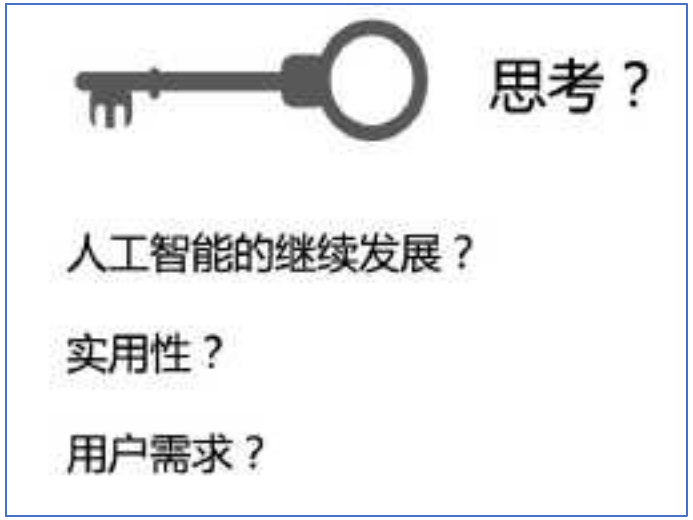

# Group 19:校园巴士路线优化


# Group 11:句子图片的文本情感分析



# Group 15:有关人工智能在视障人群辅助器械中的应用的研究

# Group 16: Hand-written numerals recognition


# Group 18:个性化推荐系统


# Group 20: AI for coloring line drafts

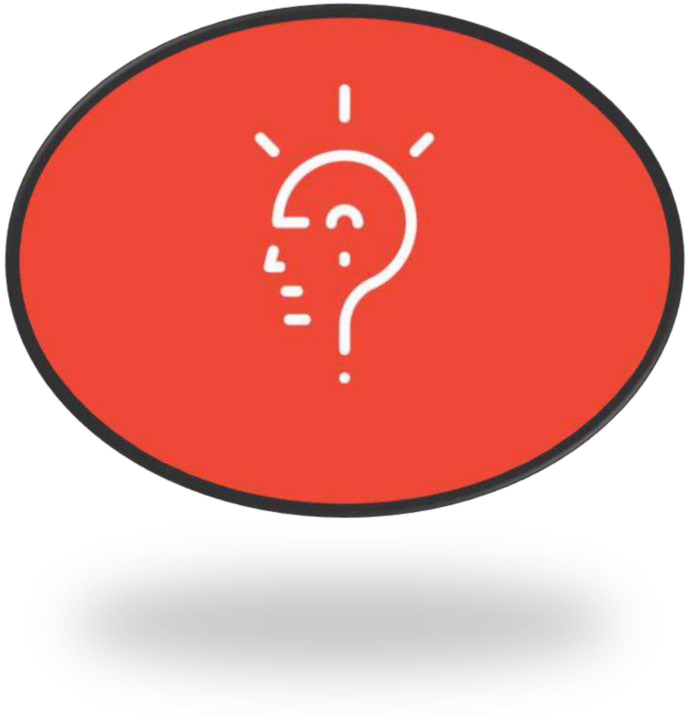

# Group 12: Gesture Recognition


# Group 21: The possibility of clinical application of artificial intelligence in skin cancer diagnosis


# 打分统计结果

| 分数等级 | 15 | 14 | 13 | 12 | 平均分 | 最后分数 |
|---|---|---|---|---|---|---|
| 第1组 | 30 | 19 | 12 | 1 | 14.258 | 15 |
| 第2组 | 31 | 18 | 0 | 1 | 14.58 | 15 |
| 第3组 | 24 | 3 | 2 | 1 | 14.667 | 15 |
| 第4组 | 16 | 11 | 3 | 0 | 14.433 | 15 |
| 第5组 | 31 | 29 | 5 | 3 | 14.294 | 15 |
| 第6组 | 23 | 16 | 10 | 3 | 14.135 | 15 |
| 第7组 | 28 | 21 | 4 | 2 | 14.364 | 15 |
| 第8组 | 20 | 22 | 5 | 3 | 14.18 | 15 |
| 第9组 | 36 | 17 | 1 | 3 | 14.509 | 15 |
| 第10组 | 14 | 11 | 6 | 0 | 14.258 | 15 |
| 第11组 | 28 | 19 | 6 | 2 | 14.327 | 15 |
| 第12组 | 15 | 7 | 0 | 0 | 14.682 | 15 |
| 第13组 | 18 | 9 | 0 | 0 | 14.667 | 15 |
| 第14组 | 11 | 11 | 1 | 0 | 14.435 | 15 |
| 第15组 | 20 | 23 | 6 | 1 | 14.24 | 15 |
| 第16组 | 22 | 12 | 4 | 1 | 14.41 | 15 |
| 第17组 | 23 | 6 | 0 | 0 | 14.793 | 15 |
| 第18组 | 19 | 16 | 9 | 1 | 14.178 | 15 |
| 第19组 | 23 | 7 | 0 | 0 | 14.767 | 15 |
| 第20组 | 23 | 20 | 2 | 4 | 14.265 | 15 |
| 第21组 | 28 | 10 | 2 | 0 | 14.65 | 15 |
| 第22组 | 18 | 9 | 2 | 0 | 14.552 | 15 |

# Any Question?



# Knowledge

3 5

1. Back Propagation
2. Supporter Vector Machine
3. Machine Learning
4. Knowledge

# TOP 10 Machine Learning Algorithms

## 1 K-MEANS CLUSTERING

Aims to find groups in given data set. The number of groups is represented by a variable called K.

## 2 NAIVE BAYES CLASSIFIER

A family of algorithms which assume that values of the features used in the classification are independent.

## 3 K-NEAREST NEIGHBOR (KNN)

A simple algorithm that stores all existing data objects and classifies the new data objects based on a similarity measure.

# TOP 10 Machine Learning Algorithms

4

## SUPPORT VECTOR MACHINES (SVM)

Used to sort two data sets by similar classification. Draw lines (hyperplanes) that separate the groups according to some patterns.

5

## DECISION TREE

A machine learning technique for data mining that creates classification or regression models in the shape of a tree structure.

6

## GENERALIZED LINEAR MODELS (GLM)

Combines a number of models including linear regression models, logistic regression, Poisson regression, ANOVA, log-linear models and etc.

7

## NEURAL NETWORKS

Nonlinear models which represent a metaphor for the functioning of the human brain.

# TOP 10 Machine Learning Algorithms

8

## ASSOCIATION RULES

If/then statements that aim to uncover some relationships between unrelated data in a given database.

9

## GENETIC ALGORITHMS

A family of stochastic search algorithms witch mechanism is inspired by the process of neo-Darwinian evolution.

10

## LATENT DIRICHLET ALLOCATION (LDA)

A generative probabilistic model designed for collections of discrete data.

# Knowledge

3 5

1 Back Propagation

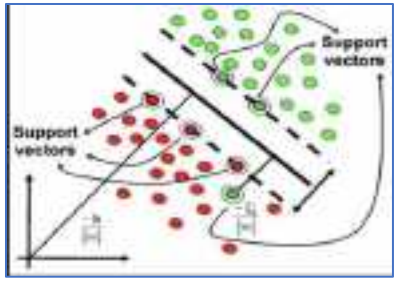
2 Supporter Vector Machine

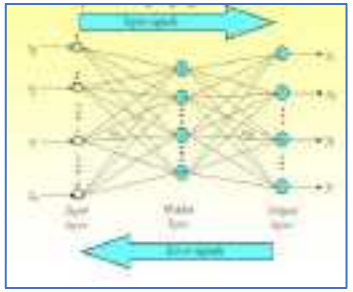
3 Machine Learning

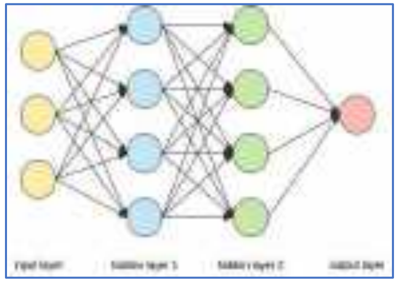
4 Knowledge

# AI Algorithm Development –Machine Learning

- Back Propagation
- Support Vector Machine
- Machine Learning
- Knowledge
    - Knowledge Representation
    - Knowledge Graph
    - Knowledge Tree
    - Knowledge Search
    - Deep Blue






# Knowledge Representation - Data

## Knowledge Representation and Reasoning

Knowledge representation and reasoning (KR², KR&R) is the field of artificial intelligence (AI) dedicated to representing information about the world in a form that a computer system can utilize to solve complex tasks such as diagnosing a medical condition or having a dialog in a natural language. Knowledge representation incorporates findings from psychology about how humans solve problems and represent knowledge in order to design formalisms that will make complex systems easier to design and build. Knowledge representation and reasoning also incorporates findings from logic to automate various kinds of reasoning, such as the application of rules or the relations of sets and subsets.

W Wikipedia

Lists - linked lists are used to represent hierarchical knowledge. LISP, the main programming language of AI, was developed to process lists

Trees - graphs which represent hierarchical knowledge.

Semantic networks - nodes and links - stored as propositions.

Schemas - used to represent commonsense or stereotyped knowledge.

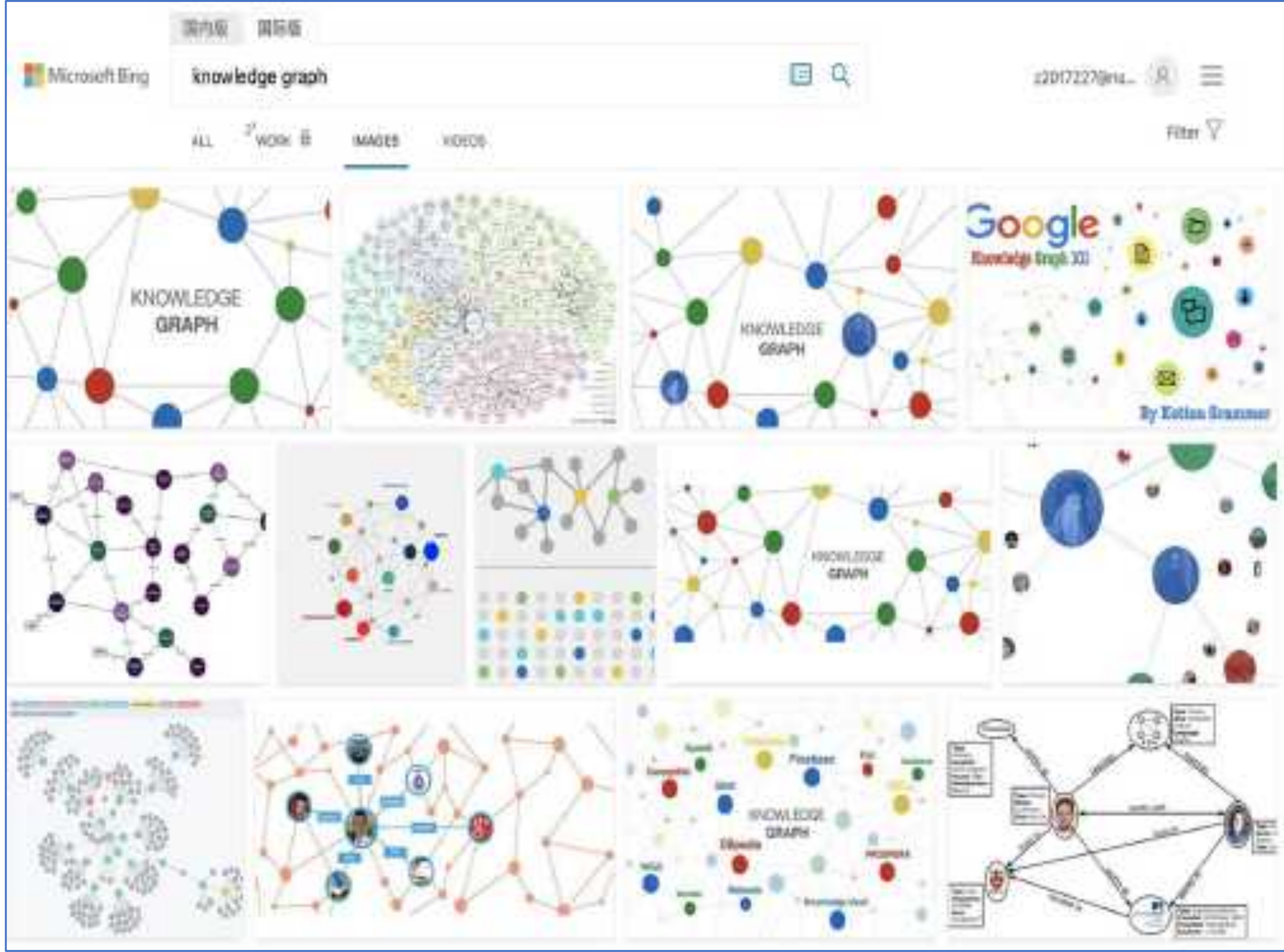
- Frames - Describe objects. Consist of a cluster of nodes and links manipulated as a whole. Knowledge is organised in slots. Frames are hierarchically organised.
- Scripts - Describe event rather than objects. Consist of stereotypically ordered causal or temporal chain of events.

# Knowledge Graph

## Knowledge Graph

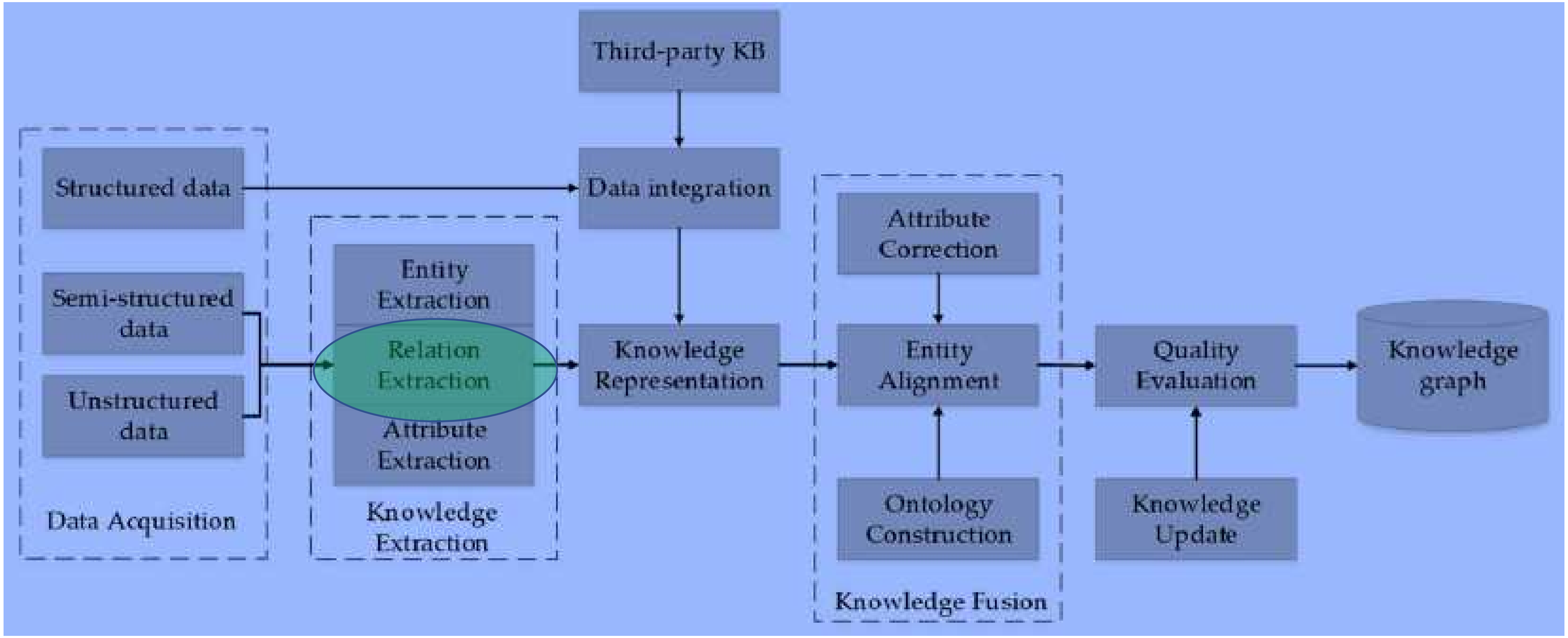

The Google Knowledge Graph is a knowledge base used by Google and its services to enhance its search engine's results with information gathered from a variety of sources. The information is presented to users in an infobox next to the search results. These infoboxes were added to Google's search engine in May 2012, starting in the United States, with international expansion by the end of the year. Google has referred to these infoboxes, which appear to the right (top on mobile) of search results, as "knowledge panels".

Wikipedia Official site




1. Singhal A., Official Google Blog: Introducing the Knowledge Graph: things, not strings. Official Google Blog (2012) 1-8.

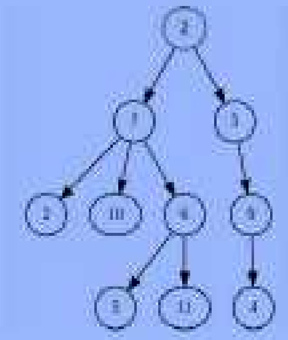
# Technical Architecture of Knowledge Graph



# Knowledge Tree

## Tree

Data Structure

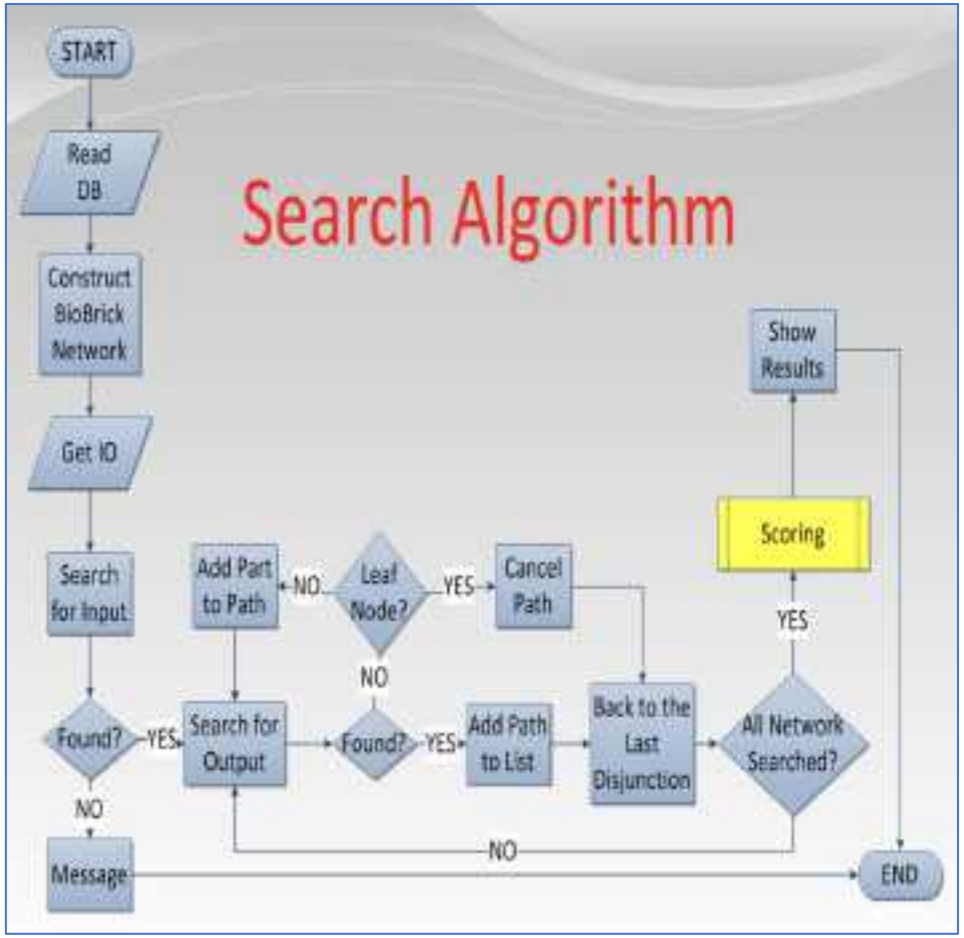


In computer science, a tree is a widely used abstract data type that simulates a hierarchical tree structure, with a root value and subtrees of children with a parent node, represented as a set of linked nodes.

- A tree is non-linear data structure in which items are arranged in a sorted sequence. It is used to impose a hierarchical structure on a collection of data items.
- A tree is recursively defined as a set of one or more nodes where one node is designated as the root of the tree and all the remaining nodes can be partitioned into non-empty sets each of which is a sub-tree of the root.
- A node is a structure which may contain a value, a condition, or represent a separate data structure (which could be a tree of its own).
- Each node in a tree has zero or more child nodes, which are below it in the tree (by convention, trees grow down, not up as they do in nature).

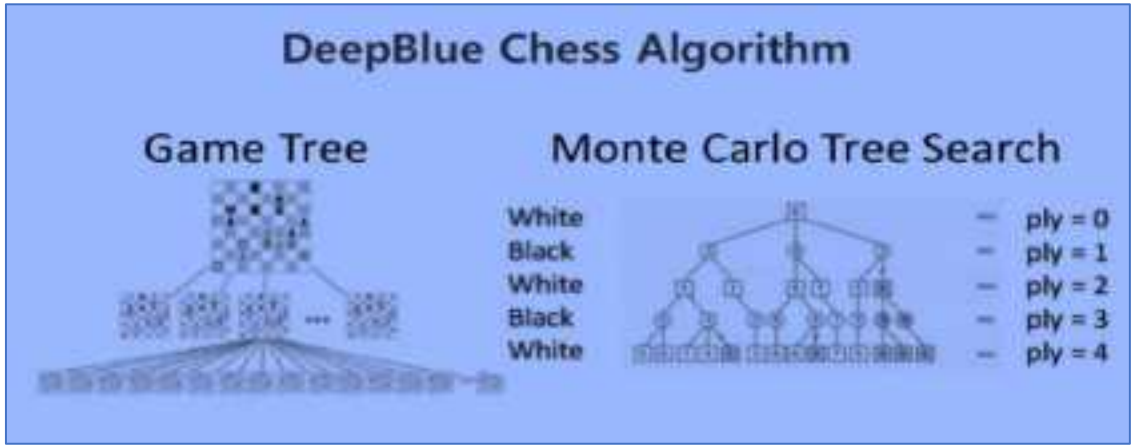
# Knowledge Search

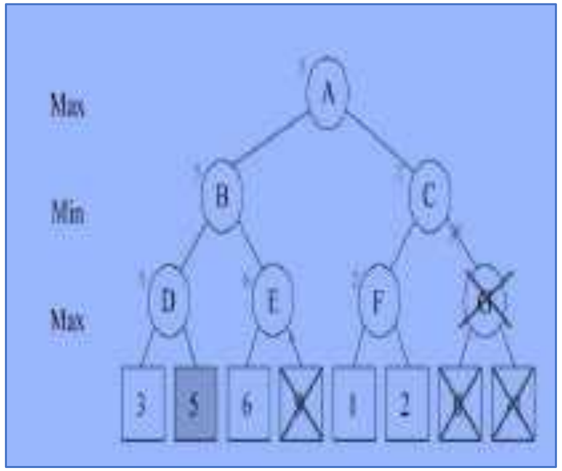
- State space
  - the set of all states reachable from the initial state
- Knowledge Representation of the state space
  - A powerful tool for representing the structure and complexity of problems
- Problem solving
  - A process of searching the state space for a path to a solution
- Search strategy (algorithms)
  - The choice of which state to expand



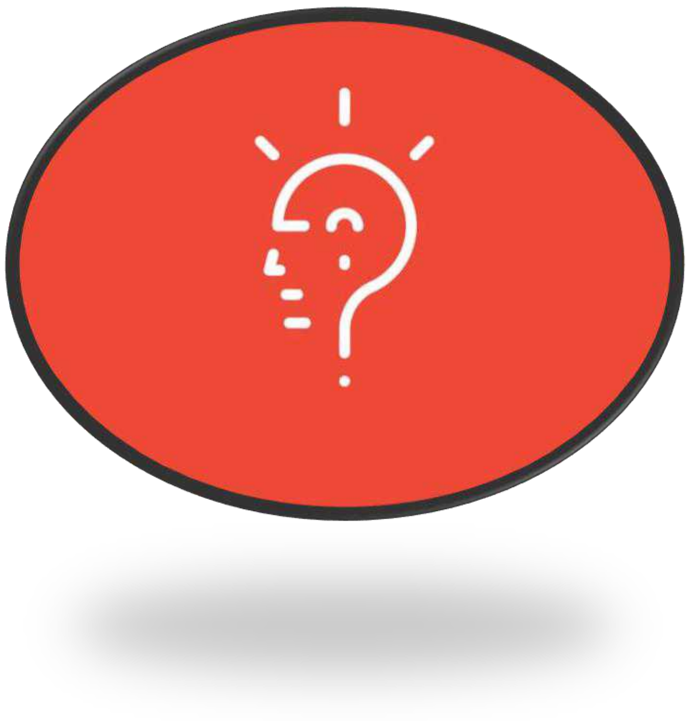
# Deep Blue

On May 11, 1997, an IBM computer called IBM ® Deep Blue ® beat the world chess champion after a six-game match: two wins for IBM, one for the champion and three draws. The match lasted several days and received massive media coverage around the world. It was the classic plot line of man vs. machine. Behind the contest, however, was important computer science, pushing forward the ability of computers to handle the kinds of complex calculations needed to help discover new medical drugs; do the broad financial modeling needed to identify trends and do risk analysis; handle large database searches; and perform massive calculations needed in many fields of science.




# Any Question?



# Deep Learning

3 6

1 Deep Learning

2 Convolution

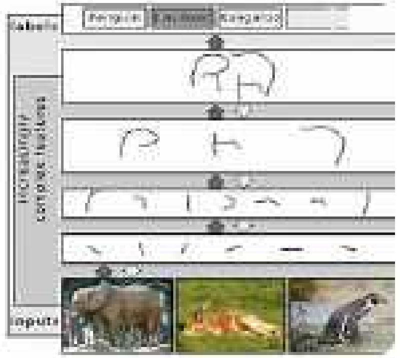
3 CNN Deep Learning

4 Deep Learning and Machine Learning

# Deep Learning (Wiki 2019 and 2020)

"Deep learning is a branch of machine learning based on a set of algorithms that attempt to model high-level abstractions in data by using multiple processing layers, with complex structures or otherwise, composed of multiple non-linear transformations."


## Deep Learning



Deep learning (also known as deep structured learning) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.

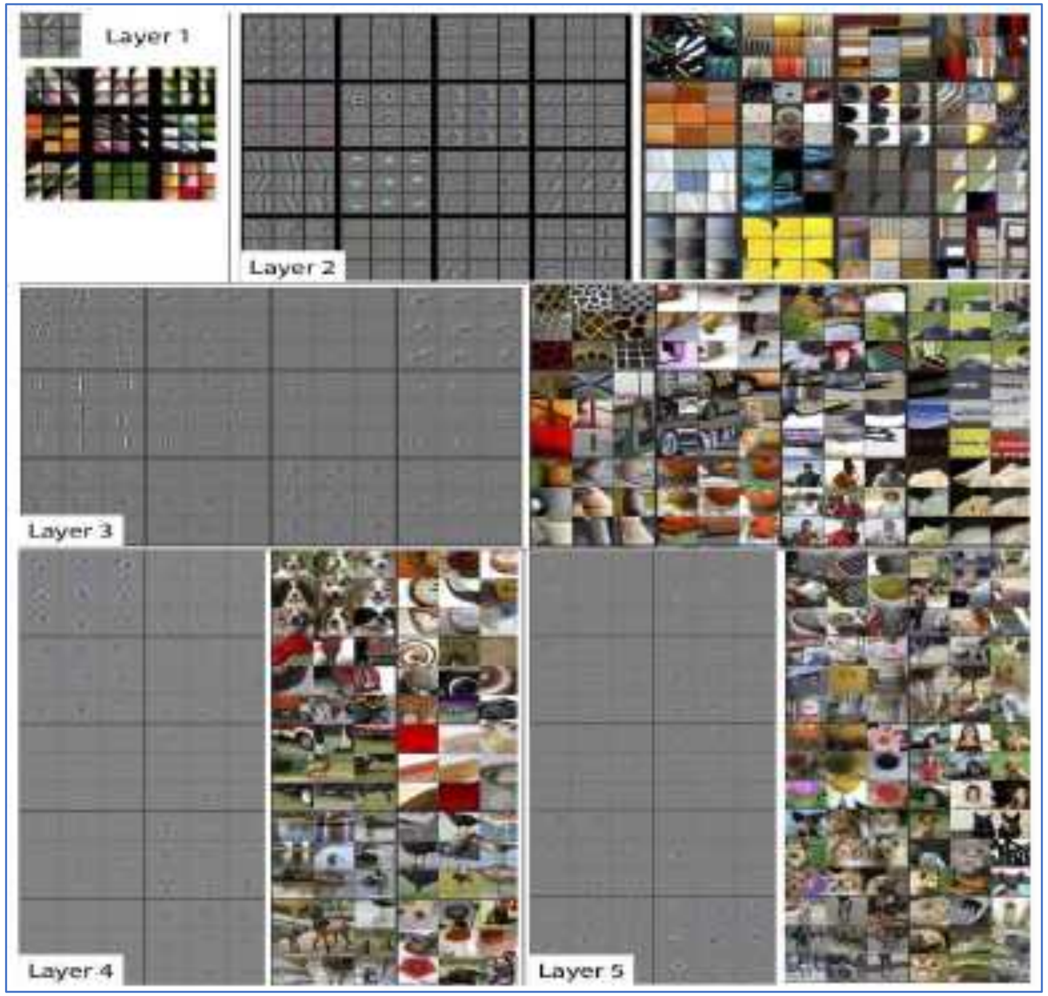
Wikipedia

# Representation Learning

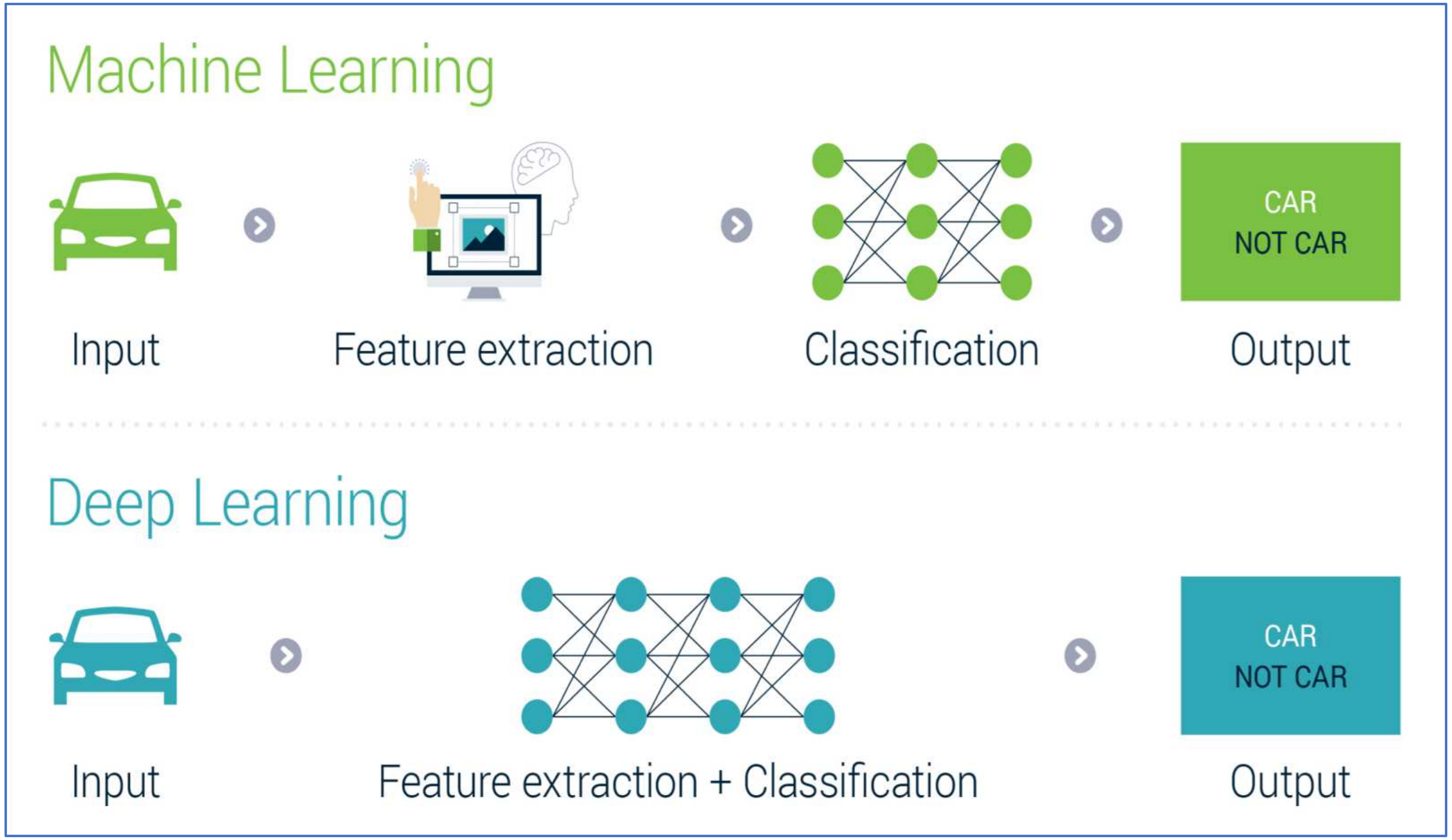
## Feature Learning

In machine learning, feature learning or representation learning is a set of techniques that allows a system to automatically discover the representations needed for feature detection or classification from raw data. This replaces manual feature engineering and allows a machine to both learn the features and use them to perform a specific task.

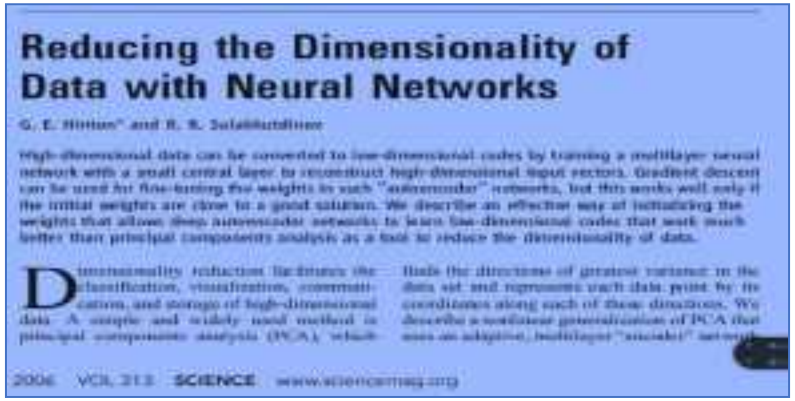
W Wikipedia



# Machine Learning and Deep Learning



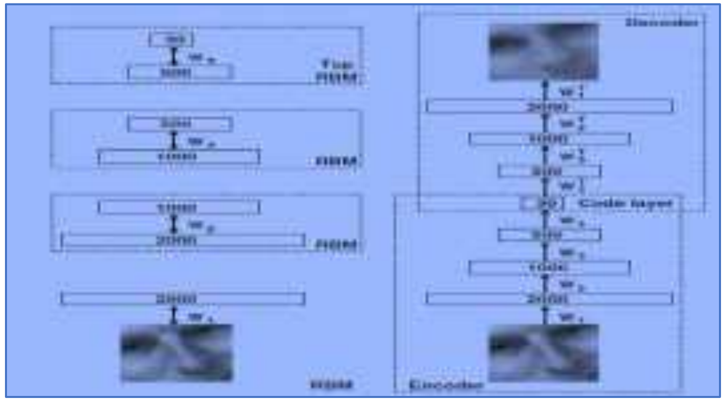
# Deep Learning Era



## Reducing the Dimensionality of Data with Neural Networks

G. E. Hinton* and R. R. Salakhutdinov

2006 VOL 313 SCIENCE www.sciencemag.org

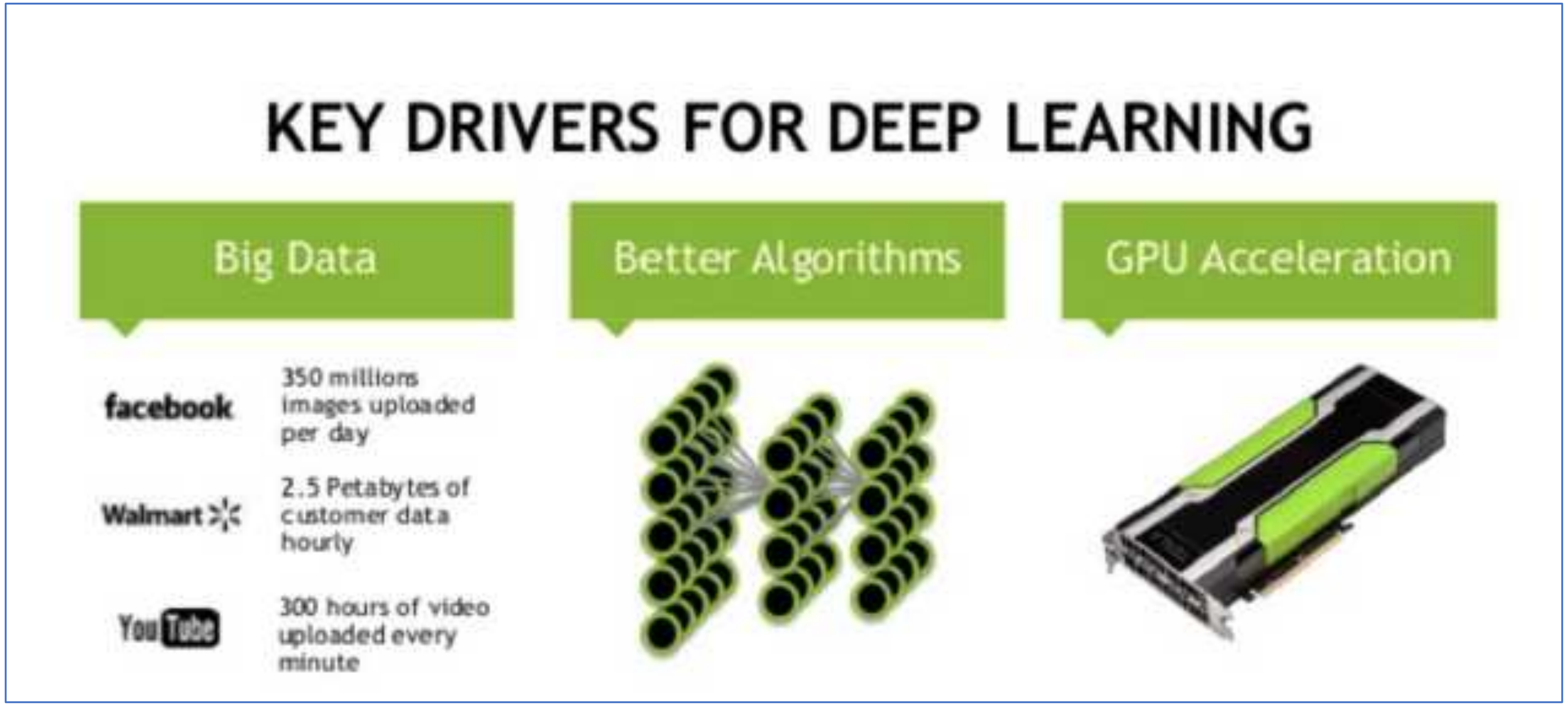



"A fast learning algorithm for deep belief nets"
-- Hinton et al., 2006

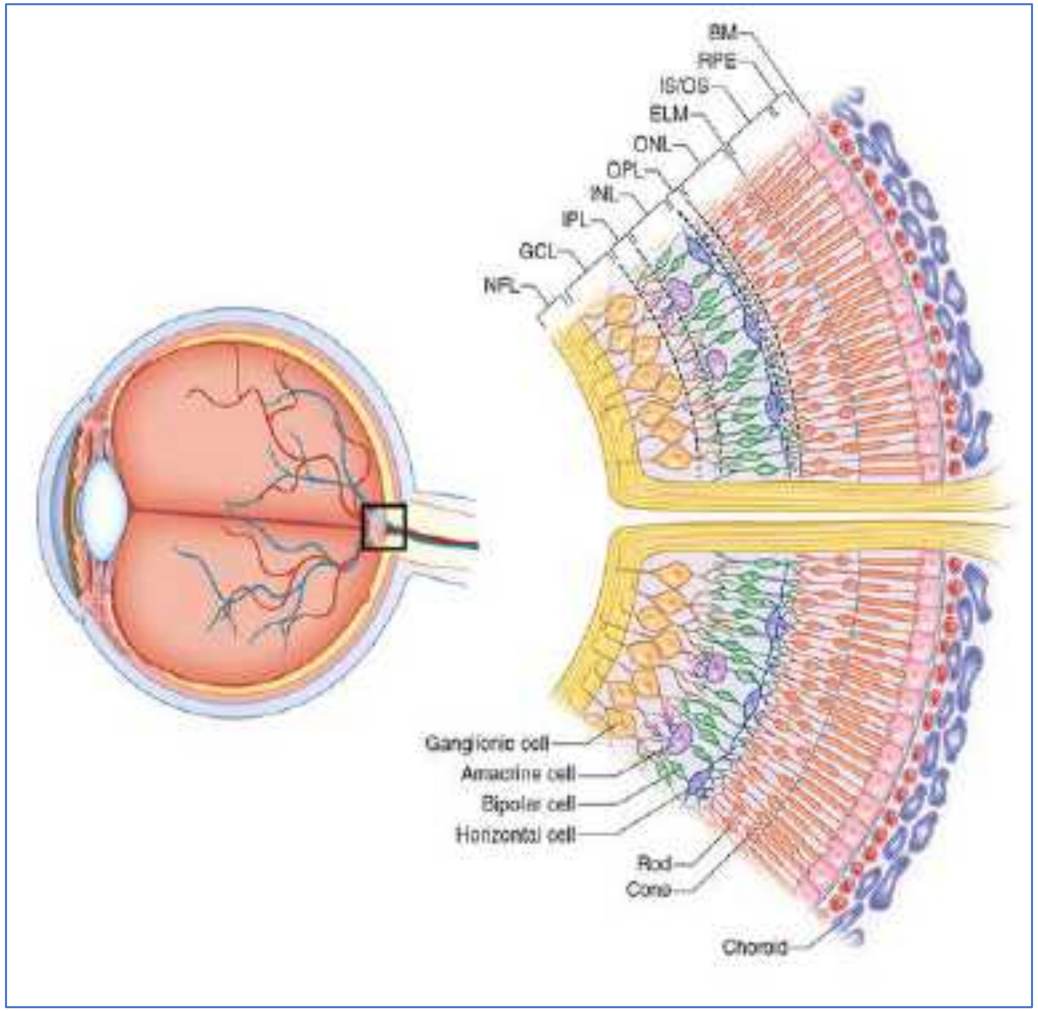
"Reducing the dimensionality of data with neural networks"
-- Hinton & Salakhutdinov

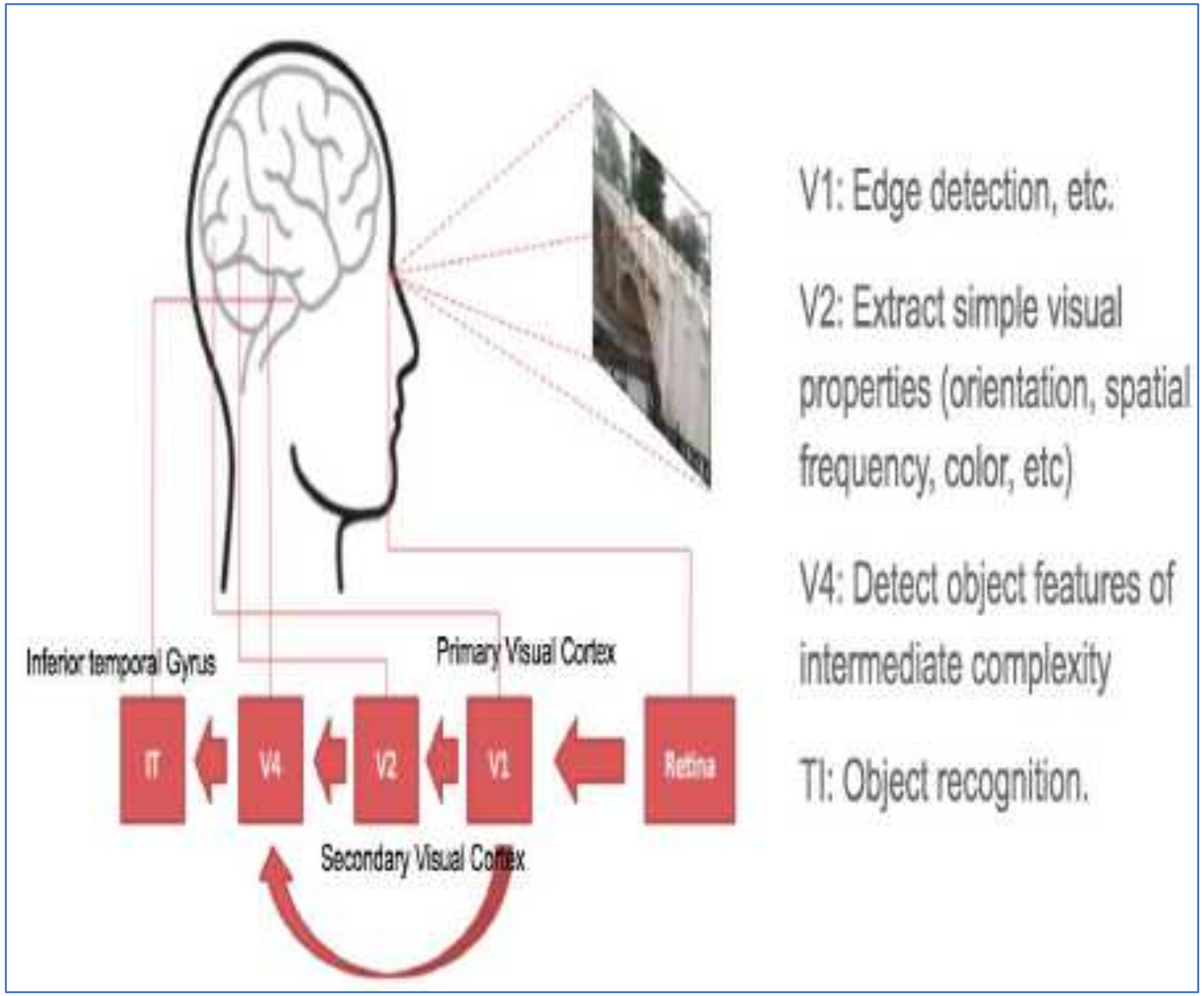
Geoffrey Hinton
University of Toronto

# 3 Pillars for Deep Learning



# Recall: Deep Layered Visual Input to Brain



# Deep Learning

3 6

1 Deep Learning

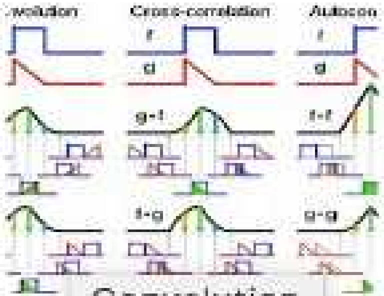
2 Convolution

3 CNN Deep Learning Models

4 Deep Learning and Machine Learning

# Convolution and CNN

## Convolution



Convolution

In mathematics (in particular, functional analysis), convolution is a mathematical operation on two functions (f and g) that produces a third function (f*g) that expresses how the shape of one is modified by the other. The term convolution refers to both the result function and to the process of computing it. It is defined as the integral of the product of the two functions after one is reversed and shifted.

Wikipedia

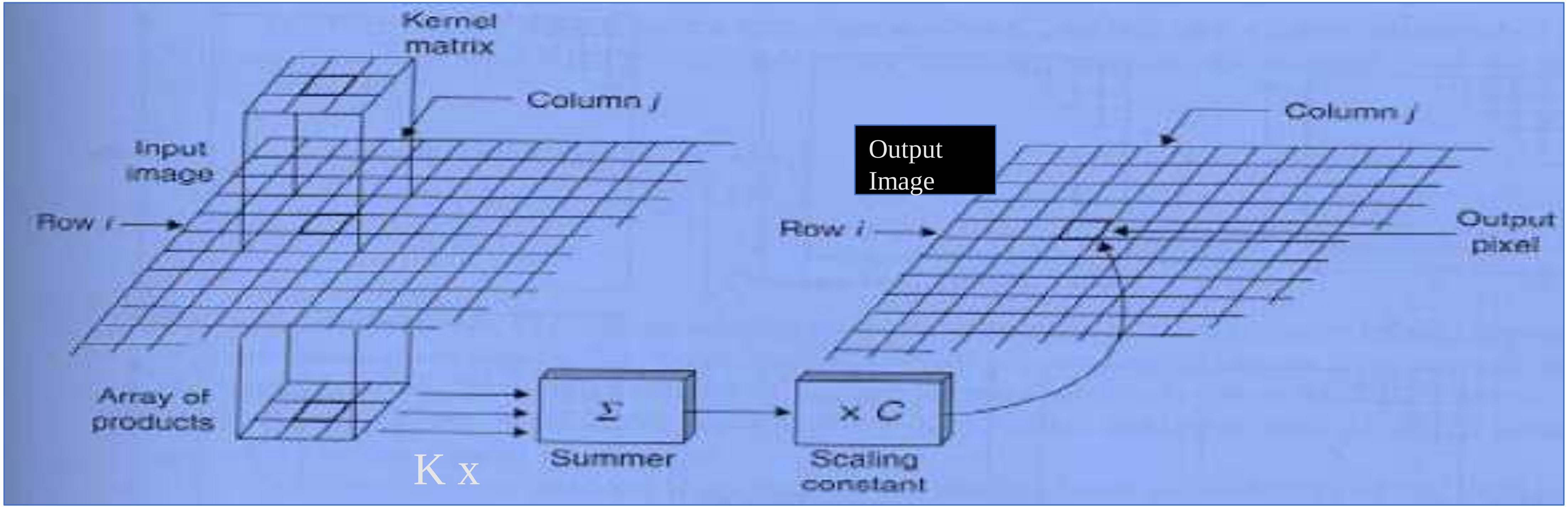
## Convolutional Neural Network

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics. They have applications in image and video recognition, recommender systems, image classification, medical image analysis, natural language processing, brain-computer interfaces, and financial time series.

Wikipedia

# Two Concepts: Correlation

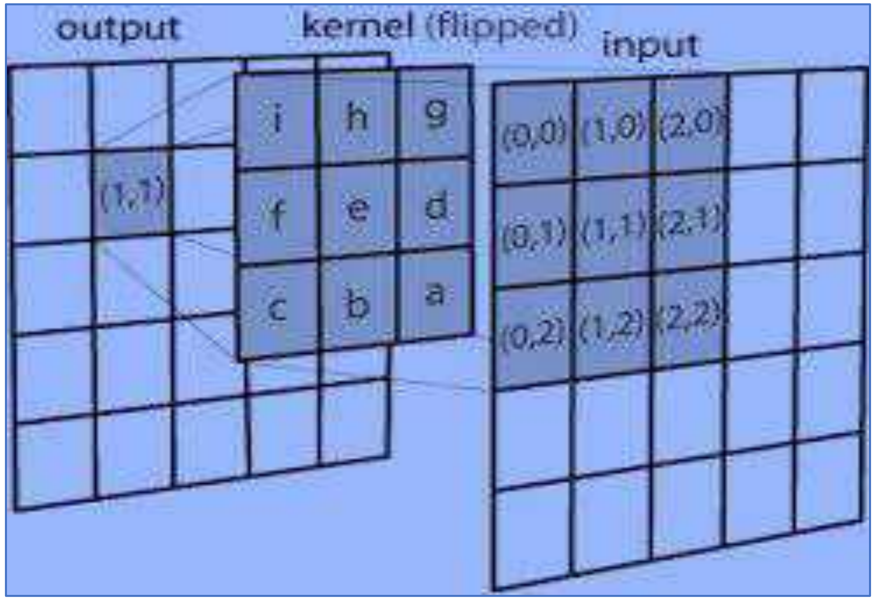
$$g(x,y) = w(x,y) \cdot f(x,y)$$




$$g(x,y) = w(x,y) \cdot f(x,y) = \sum_{s=-K/2}^{K/2} \sum_{t=-K/2}^{K/2} w(s,t) f(x+s, y+t)$$

# Two Concepts: Convolution

$$g(x,y)=w(x,y)*f(x,y)$$




$$g(x,y)=w(x,y)*f(x,y)=\sum_{s=-K/2}^{K/2}\sum_{t=-K/2}^{K/2} w(s,t)f(x-s,y-t)$$

# Correlation and Convolution

- Convolution is the same as correlation except that the mask is <u>flipped</u> both horizontally and vertically.

$$g(x,y)=w(x,y)\cdot f(x,y)=\sum_{s=-K/2}^{K/2}\sum_{t=-K/2}^{K/2} w(s,t)f(x+s,y+t)$$

$$g(x,y)=w(x,y)*f(x,y)=\sum_{s=-K/2}^{K/2}\sum_{t=-K/2}^{K/2} w(s,t)f(x-s,y-t)$$

- Note that if w(x,y) is symmetric, that is w(x,y)=w(-x,-y), then convolution is equivalent to correlation!

# Discrete 2D Convolution

- Suppose f(x,y) and g(x,y) are images of size

  A x B and C x D

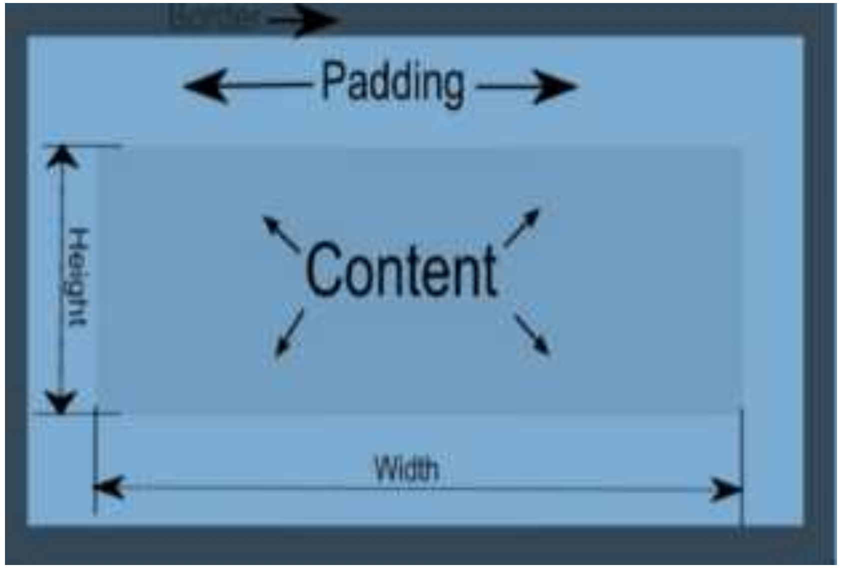
- The size of f(x,y) * g(x,y) would be N x M where

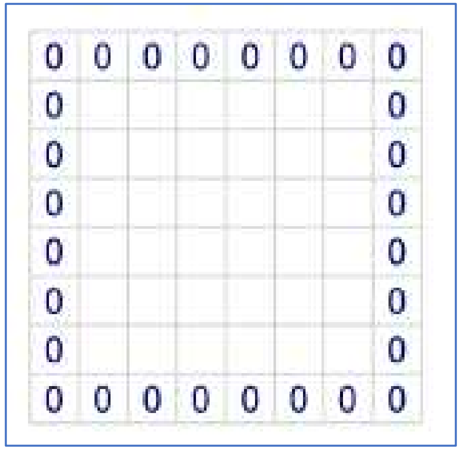
  N=A+C-1 and M=B+D-1

- Extended images (i.e., pad with zeroes)

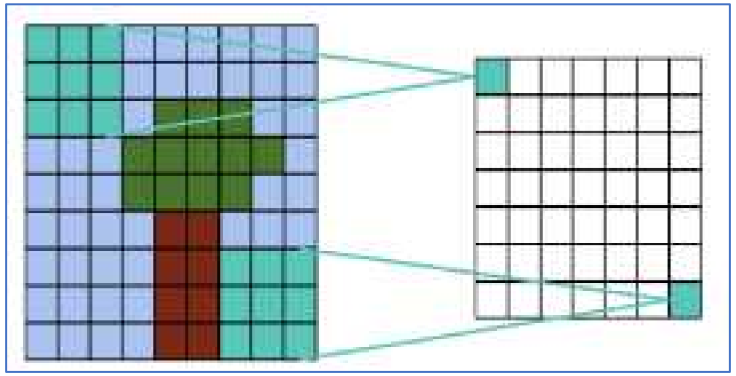
- Association Rule holds for Convolution

  Y = (X1*X2)* F = X1 * (X2*F)

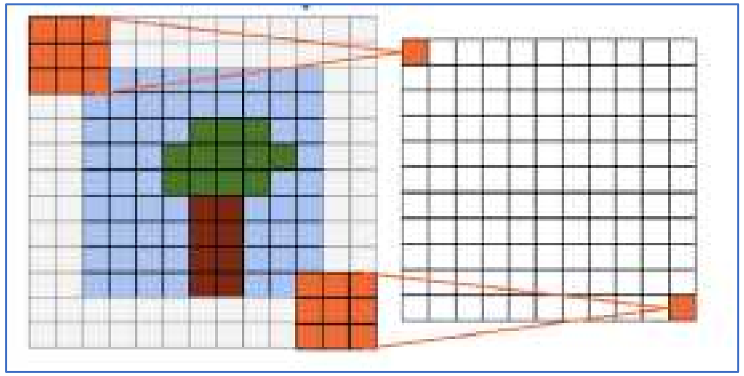


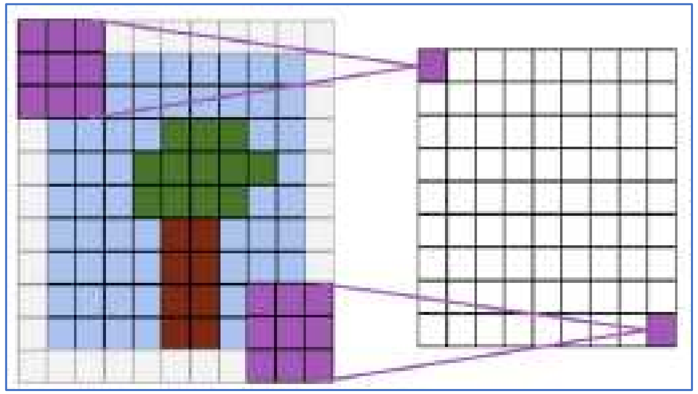
# Padding and Output Size



output size < input size

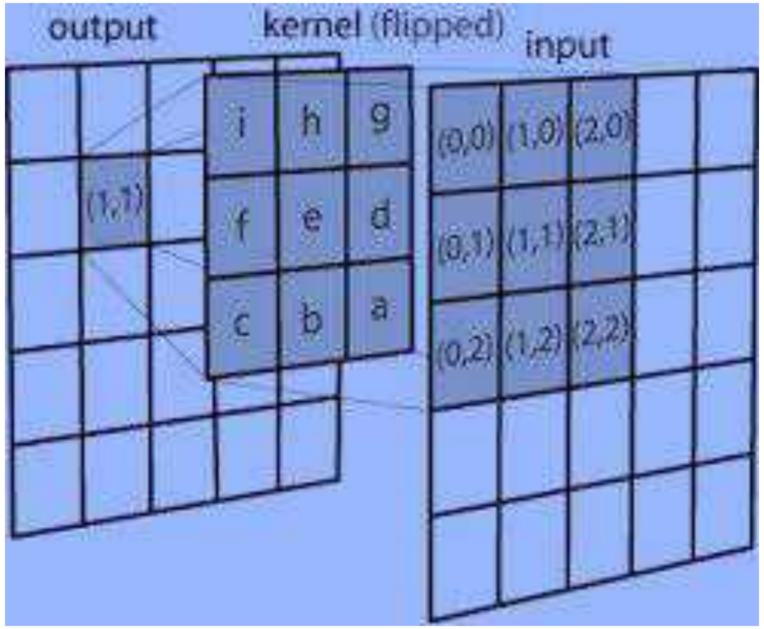


output size > input size



output size = input size

# Calculating a Discrete Convolution




During convolution, we take each kernel coefficient in turn and multiply it by a value from a neighborhood of the image lying under the kernel. We apply the kernel to the image in such a way that the value of the top-left corner of the kernel (*w*) is multiplied by the value at the bottom-right corner of the neighborhood.

# Convolution Exercise and Homework Y(0,0), Y(2,1)

- Given the Image X and the Convolution operator H.

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

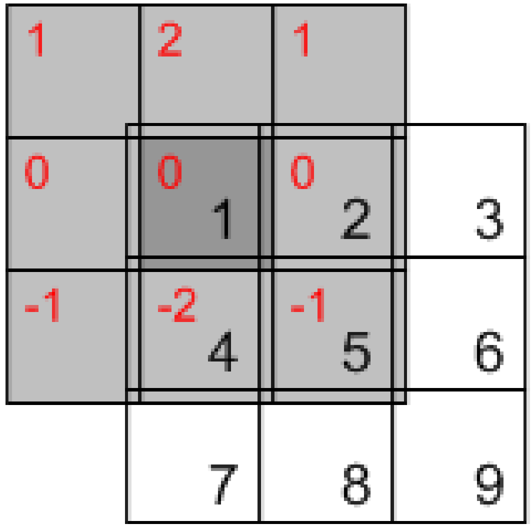
| n \ m | -1 | 0 | 1 |
|---|---|---|---|
| -1 | -1 | -2 | -1 |
| 0 | 0 | 0 | 0 |
| 1 | 1 | 2 | 1 |

- The 2D discrete convolution of Y = X*H is here is defined as:

$$y(m,n)=\sum_{k=-1}^{1}\sum_{j=-1}^{1} x(j,k)h(m-j,n-k)$$

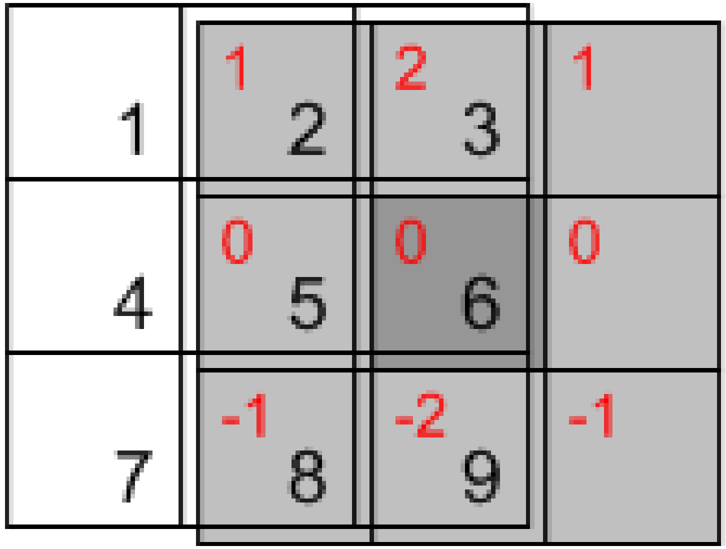
- Please use the zero-padding to extend the image during calculation
- Please calculate Y

# Convolution Exercise: Answer for Y(0,0)




$$
\begin{aligned}
y[0,0] = {} & x[-1,-1]\cdot h[1,1] + x[0,-1]\cdot h[0,1] + x[1,-1]\cdot h[-1,1] \\
& + x[-1,0]\cdot h[1,0] + x[0,0]\cdot h[0,0] + x[1,0]\cdot h[-1,0] \\
& + x[-1,1]\cdot h[1,-1] + x[0,1]\cdot h[0,-1] + x[1,1]\cdot h[-1,-1] \\
= {} & 0\cdot 1 + 0\cdot 2 + 0\cdot 1 + 0\cdot 0 + 1\cdot 0 + 2\cdot 0 + 0\cdot(-1) + 4\cdot(-2) + 5\cdot(-1) = -13
\end{aligned}
$$

# Convolution Exercise: Answer for Y(2,1)

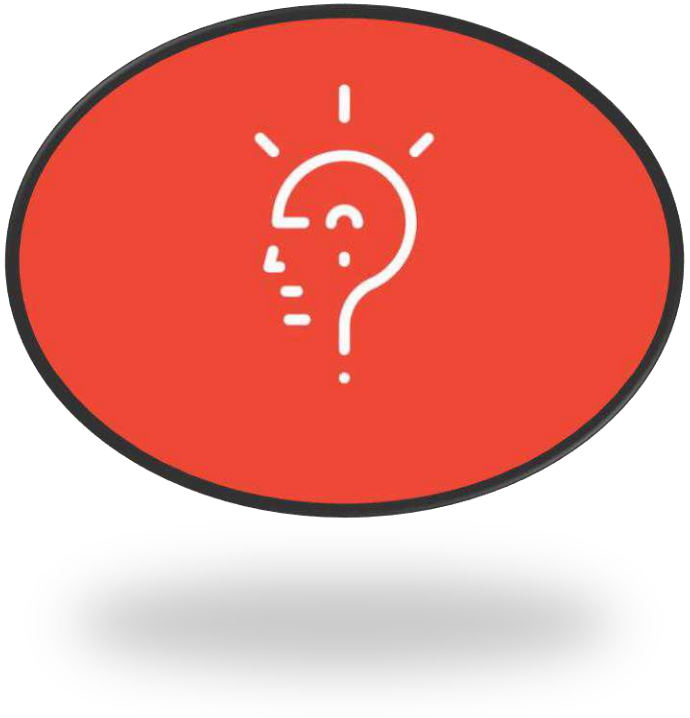



$$
\begin{aligned}
y[2,1] = {} & x[1,0]\cdot h[1,1] + x[2,0]\cdot h[0,1] + x[3,0]\cdot h[-1,1] \\
& + x[1,1]\cdot h[1,0] + x[2,1]\cdot h[0,0] + x[3,1]\cdot h[-1,0] \\
& + x[1,2]\cdot h[1,-1] + x[2,2]\cdot h[0,-1] + x[3,2]\cdot h[-1,-1] \\
= {} & 2\cdot 1 + 3\cdot 2 + 0\cdot 1 + 5\cdot 0 + 6\cdot 0 + 0\cdot 0 + 8\cdot(-1) + 9\cdot(-2) + 0\cdot(-1) = -18
\end{aligned}
$$

# Convolution Exercise Home Work and Answer

| | | |
|---|---|---|
| -13 | -20 | -17 |
| -18 | -24 | -18 |
| 13 | 20 | 17 |

# Any Question?



# Deep Learning

3 6

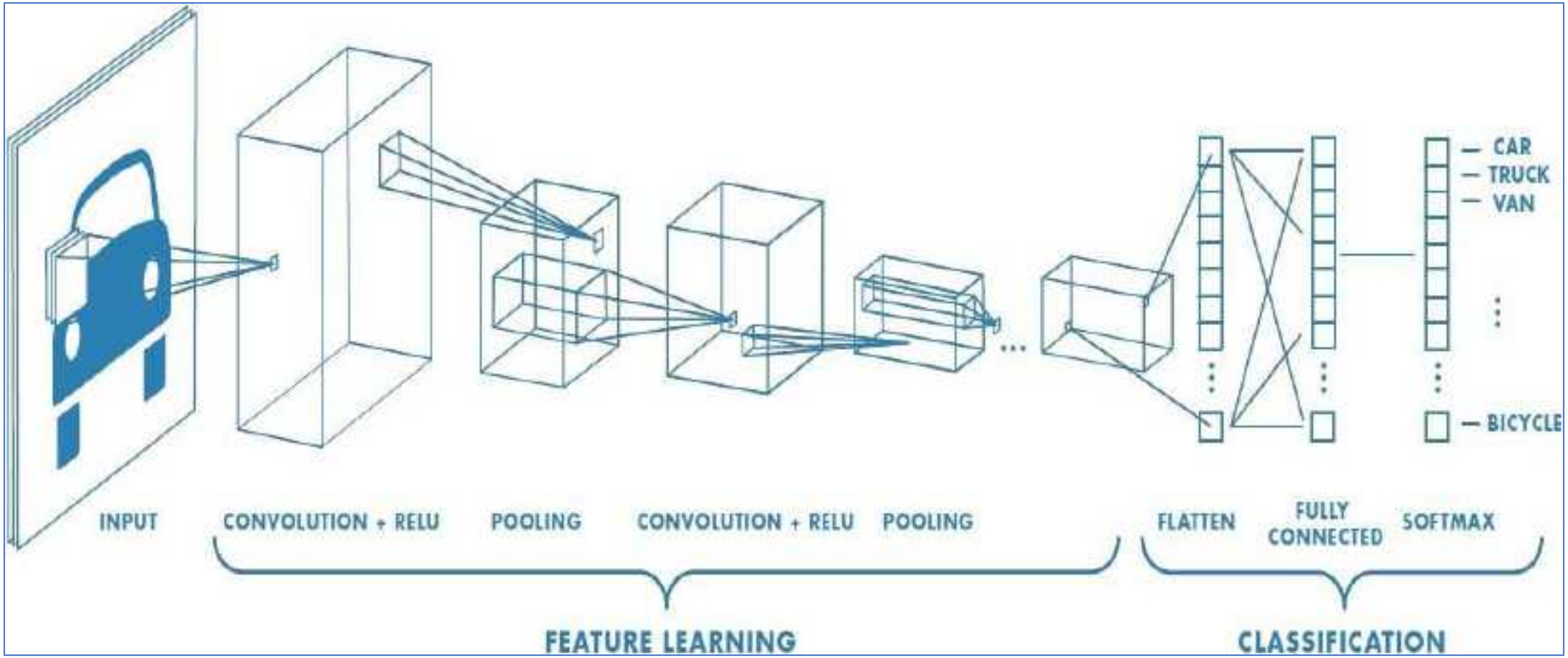
1 Deep Learning

2 Convolution

3 CNN Deep Learning

4 Deep Learning and Machine Learning

# Convolutional Neural Network



# Convolutional Layer

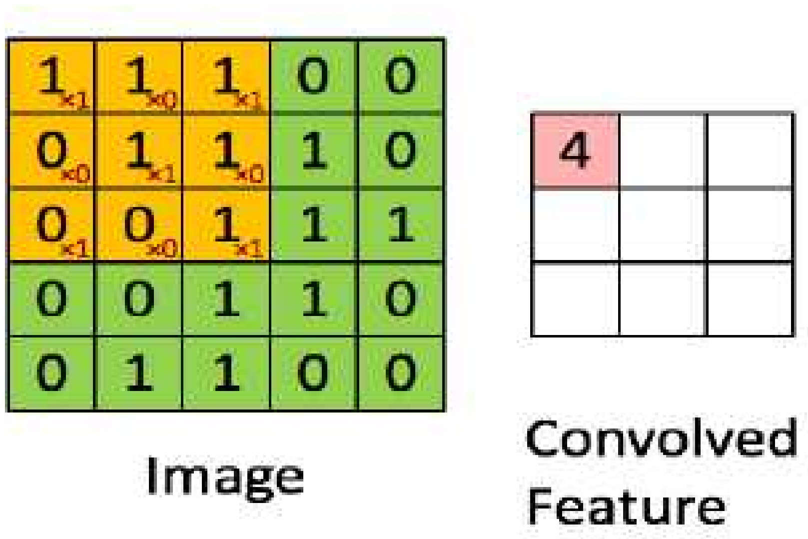
**A convolutional layer within a neural network should have the following attributes:**

1. Convolutional kernels defined by a width and height (hyper-parameters).
2. The number of input channels and output channels (hyper-parameter).
3. The depth of the Convolution filter (the input channels) must be equal to the number channels (depth) of the input feature map.

Convolutional layers convolve the input and pass its result to the next layer.

**Characteristics:**

- *Local connectivity*
- *Spatial arrangement*
- *Parameter sharing*



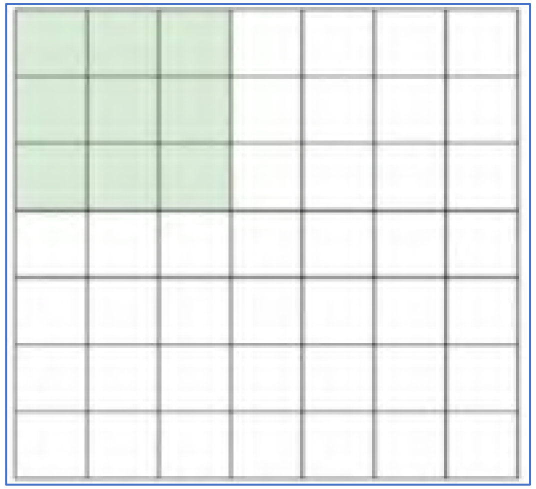

Convoluting a 5x5x1 image with a 3x3x1 kernel to get a 3x3x1 convolved feature

# Convolution for a 3X3 kernel

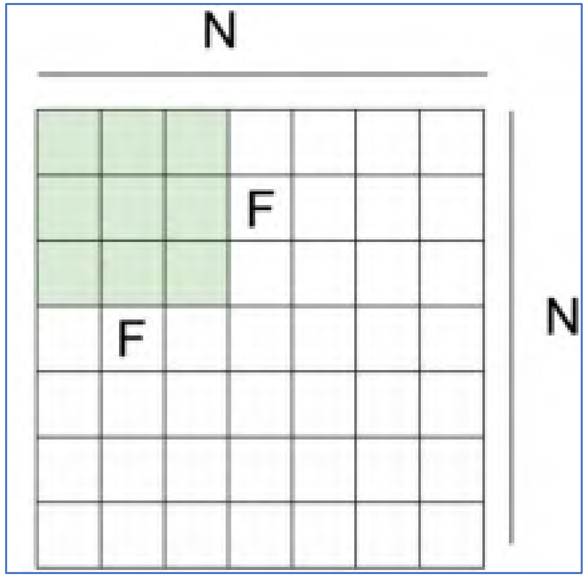
$$g(x,y)=\sum_{k=-1}^{1}\sum_{j=-1}^{1} w(j,k)\,f(x-j,y-k)$$

$$\begin{aligned}g(x,y)=\;& w(-1,-1)f(x+1,y+1) + w(0,-1)f(x,y+1) + w(1,-1)f(x-1,y+1) +\\ & w(-1,0)f(x+1,y) + w(0,0)f(x,y) + w(1,0)f(x-1,y) +\\ & w(-1,1)f(x+1,y-1) + w(0,1)f(x,y-1) + w(1,1)f(x-1,y-1)\end{aligned}$$

# Output Size Computing without padding



3x3 filter & stride=1
7x7 input (spatially) +assume 3x3 filter + applied with stride 1 => 5x5 output!
3x3 filter & stride=2
7x7 input (spatially) +assume 3x3 filter + applied with stride 1 => 3x3 output

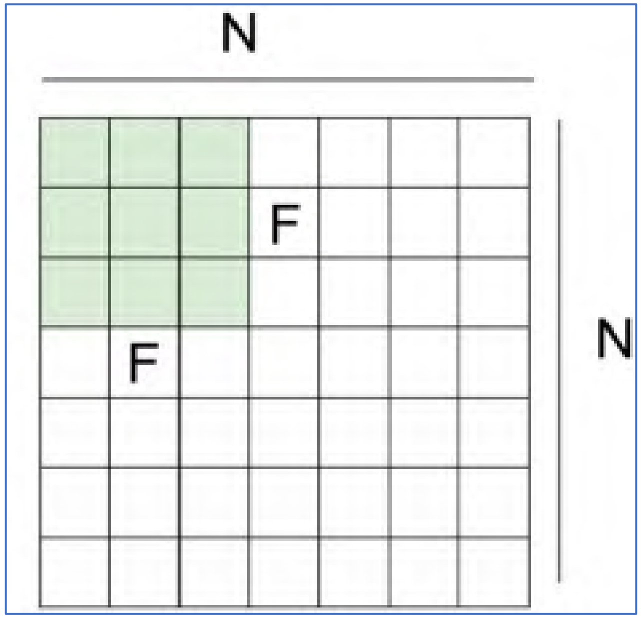



N:输入width/height, F: 卷积核大小/filter_size, stride:步长

$$outputsize = \frac{N - F}{stride} + 1$$

e.g. N = 7, F = 3:
stride 1 => (7 - 3)/1 + 1 = 5
stride 2 => (7 - 3)/2 + 1 = 3
stride 3 => (7 - 3)/3 + 1 = 2.33

# Output Size Computing with padding




e.g. input 7x7

3x3 filter, applied with stride 1

pad with 1 pixel border => 7x7 output!

in general, common to see CONV layers with stride 1, filters of size FxF, and zero-padding with (F-1)/2. (will preserve size spatially)

e.g. F = 3 => zero pad with 1

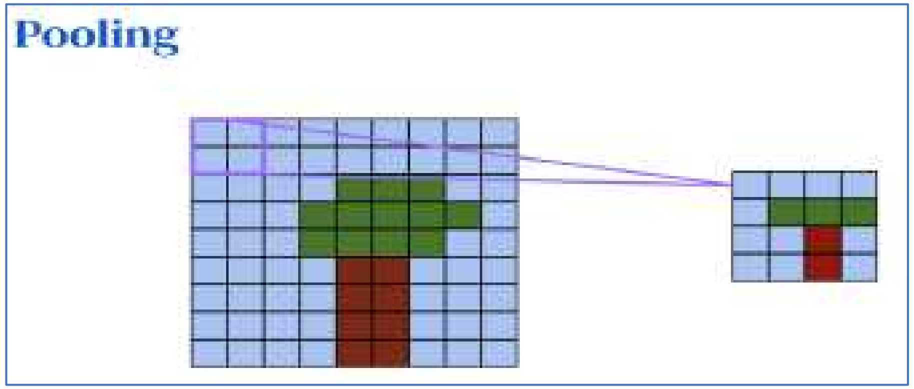
F = 5 => zero pad with 2 F = 7 => zero pad with 3

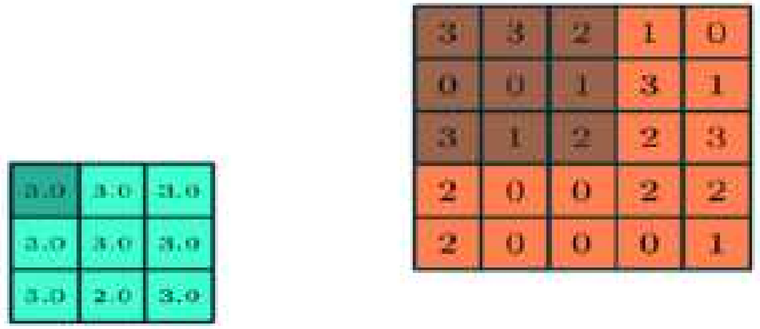
N:输入width/height, F: 卷积核大小/filter_size, stride:步长, Pad: paddings大小

$$outputsize = \frac{N - F + 2 * Pad}{stride} + 1$$

# Pooling Layer

Convolutional networks may include local or global pooling layers to streamline the underlying computation. Pooling layers reduce the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. Local pooling combines small clusters, typically 2 x 2. Global pooling acts on all the neurons of the convolutional layer. In addition, pooling may compute a max or an average. *Max pooling* uses the maximum value from each of a cluster of neurons at the prior layer. *Average pooling* uses the average value from each of a cluster of neurons at the prior layer.

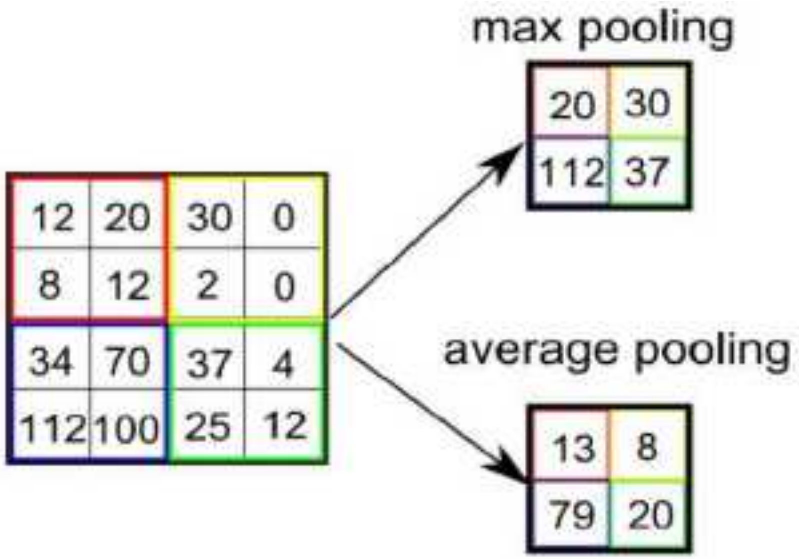


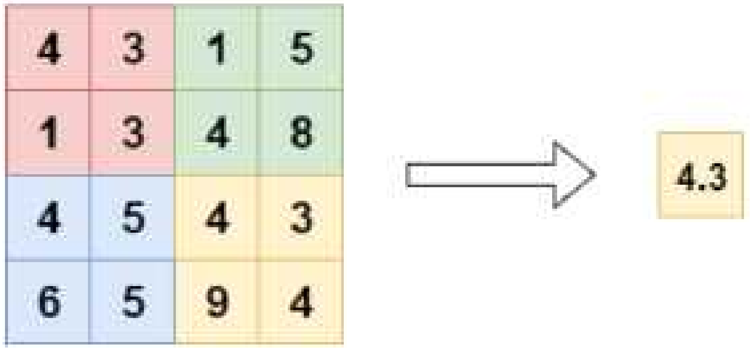




3x3 pooling over 5x5 convolved feature

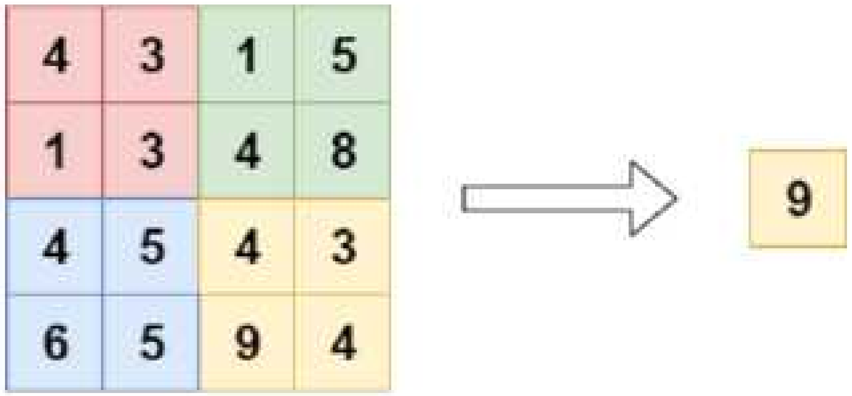
# Pooling Layer

- Accepts a volume of size $W_1 \times H_1 \times D_1$
- Requires two hyperparameters:
  - their spatial extent $F$,
  - the stride $S$,
- Produces a volume of size $W_2 \times H_2 \times D_2$ where:
  - $W_2 = (W_1 - F)/S + 1$
  - $H_2 = (H_1 - F)/S + 1$
  - $D_2 = D_1$
- Introduces zero parameters since it computes a fixed function of the input
- For Pooling layers, it is not common to pad the input using zero-padding.



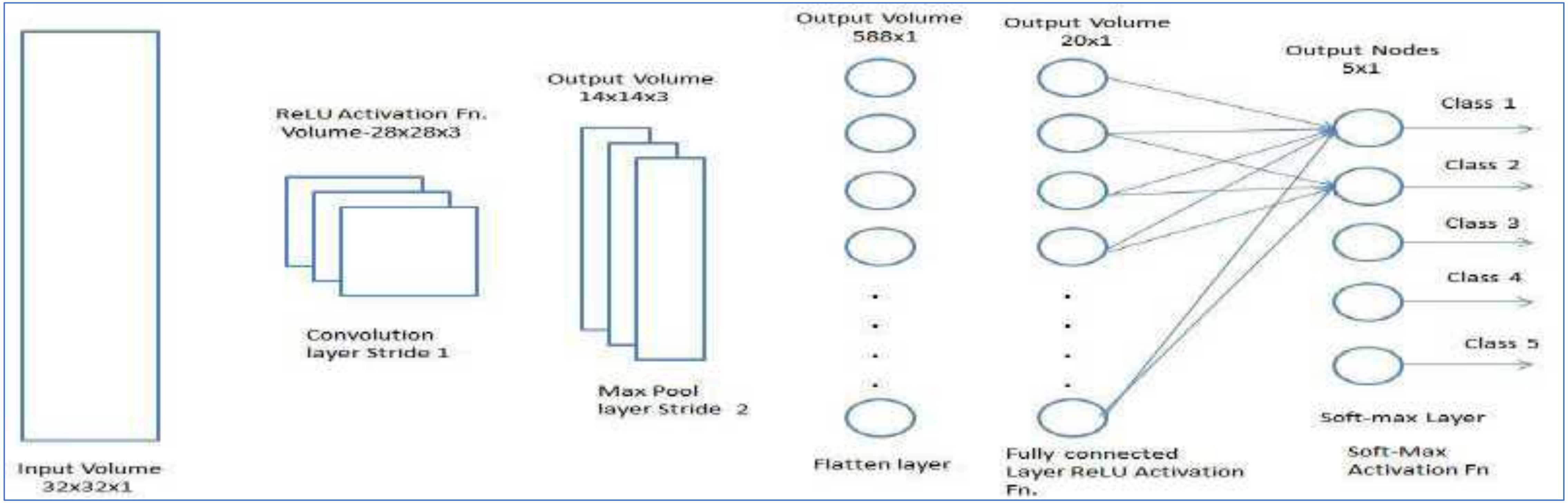

Global Average Pooling



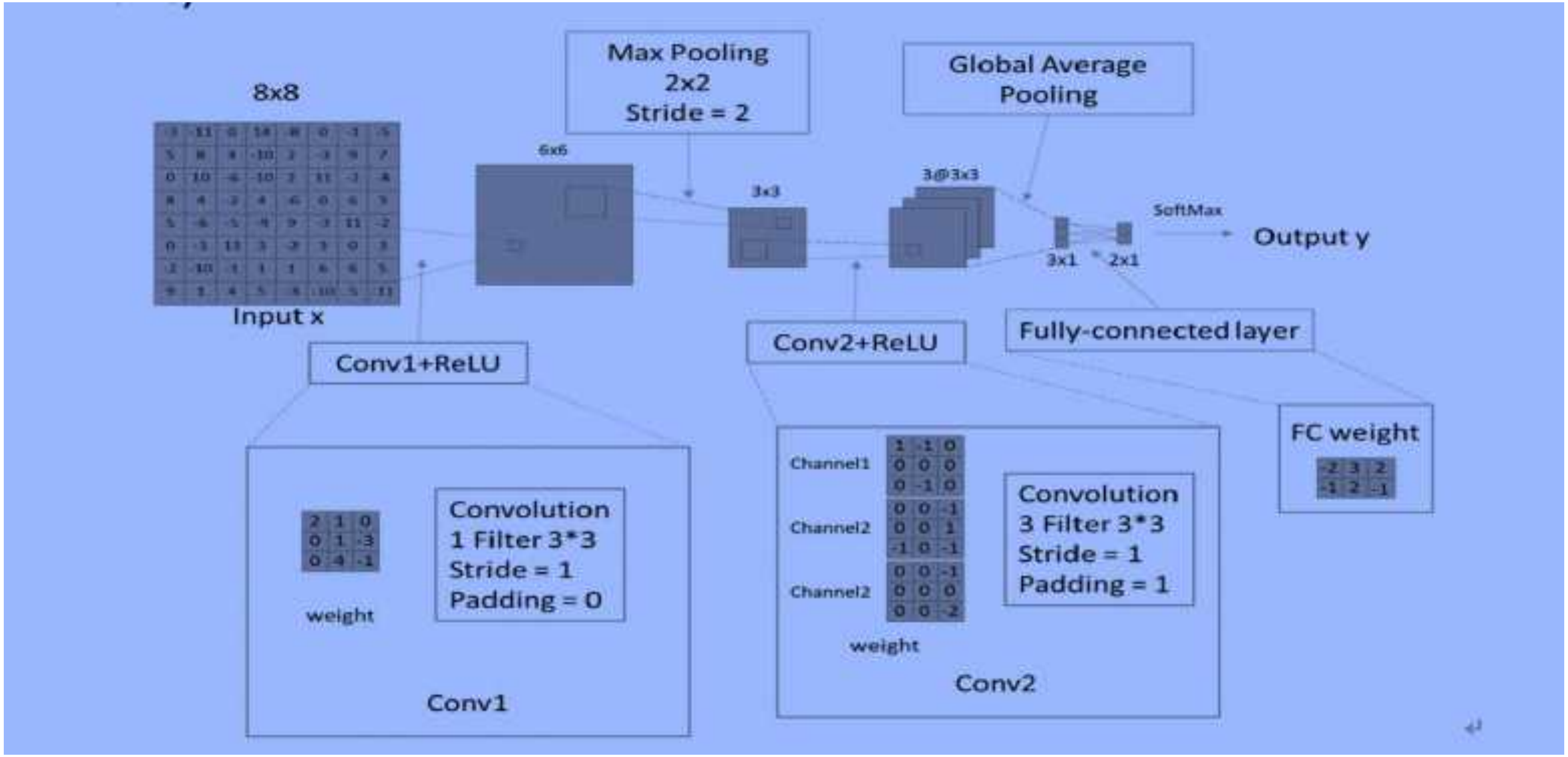

Global Max Pooling

# Fully Connected Layer (FC Layer)

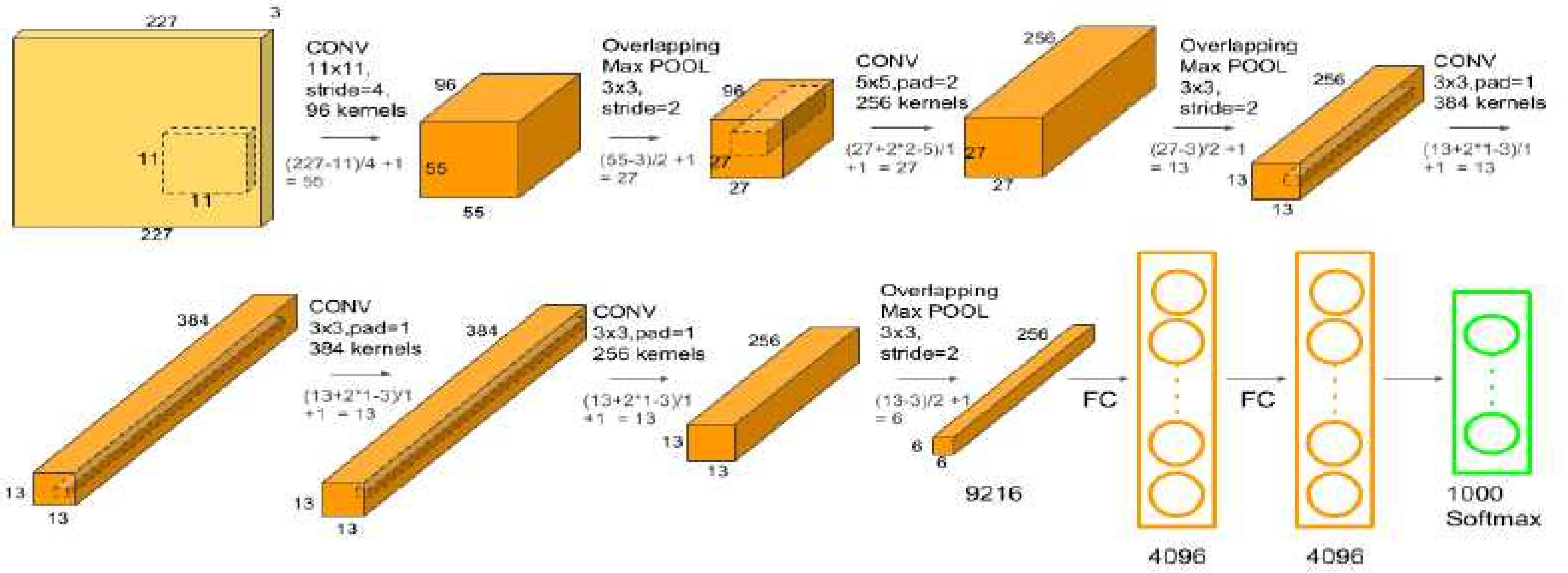
Fully connected layers connect every neuron in one layer to every neuron in another layer. It is in principle the same as the traditional multi-layer perceptron neural network (MLP). The flattened matrix goes through a fully connected layer to classify the images.



# A Typical CNN Example



# Output Size AlexNet Computing



# Number of Parameters of a Conv Layer

$W_c$ = Number of weights of the Conv Layer.

$B_c$ = Number of biases of the Conv Layer.

$P_c$ = Number of parameters of the Conv Layer.

$K$ = Size (width) of kernels used in the Conv Layer.

$N$ = Number of kernels.

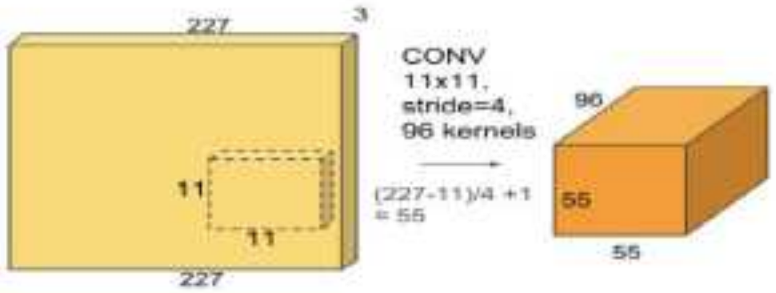
$C$ = Number of channels of the input image.

$$W_c = K^2 \times C \times N$$
$$B_c = N$$
$$P_c = W_c + B_c$$

In a Conv Layer, the depth of every kernel is always equal to the number of channels in the input image. So every kernel has $K^2 \times C$ parameters, and there are $N$ such kernels. That's how we come up with the above formula.

**Example**: In AlexNet, at the first Conv Layer, the number of channels ($C$) of the input image is 3, the kernel size ($K$) is 11, the number of kernels ($N$) is 96. So the number of parameters is given by

$$W_c = 11^2 \times 3 \times 96 = 34,848$$
$$B_c = 96$$
$$P_c = 34,848 + 96 = 34,944$$



# Number of Parameters of a Fully Connected (FC) Layer connected to a Conv Layer

$W_{ff}$ = Number of weights of a FC Layer which is connected to an FC Layer.

$B_{ff}$ = Number of biases of a FC Layer which is connected to an FC Layer.

$P_{ff}$ = Number of parameters of a FC Layer which is connected to an FC Layer.

$F$ = Number of neurons in the FC Layer.

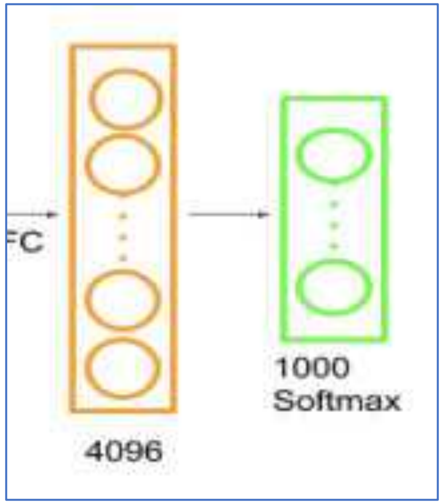
$F_{-1}$ = Number of neurons in the previous FC Layer.

$$W_{ff} = F_{-1} \times F$$
$$B_{ff} = F$$
$$P_{ff} = W_{ff} + B_{ff}$$

In the above equation, $F_{-1} \times F$ is the total number of connection weights from neurons of the previous FC Layer the neurons of the current FC Layer. The total number of biases is the same as the number of neurons ($F$).

**Example**: The last fully connected layer of AlexNet is connected to an FC Layer. For this layer, $F_{-1} = 4096$ and $F = 1000$. Therefore,

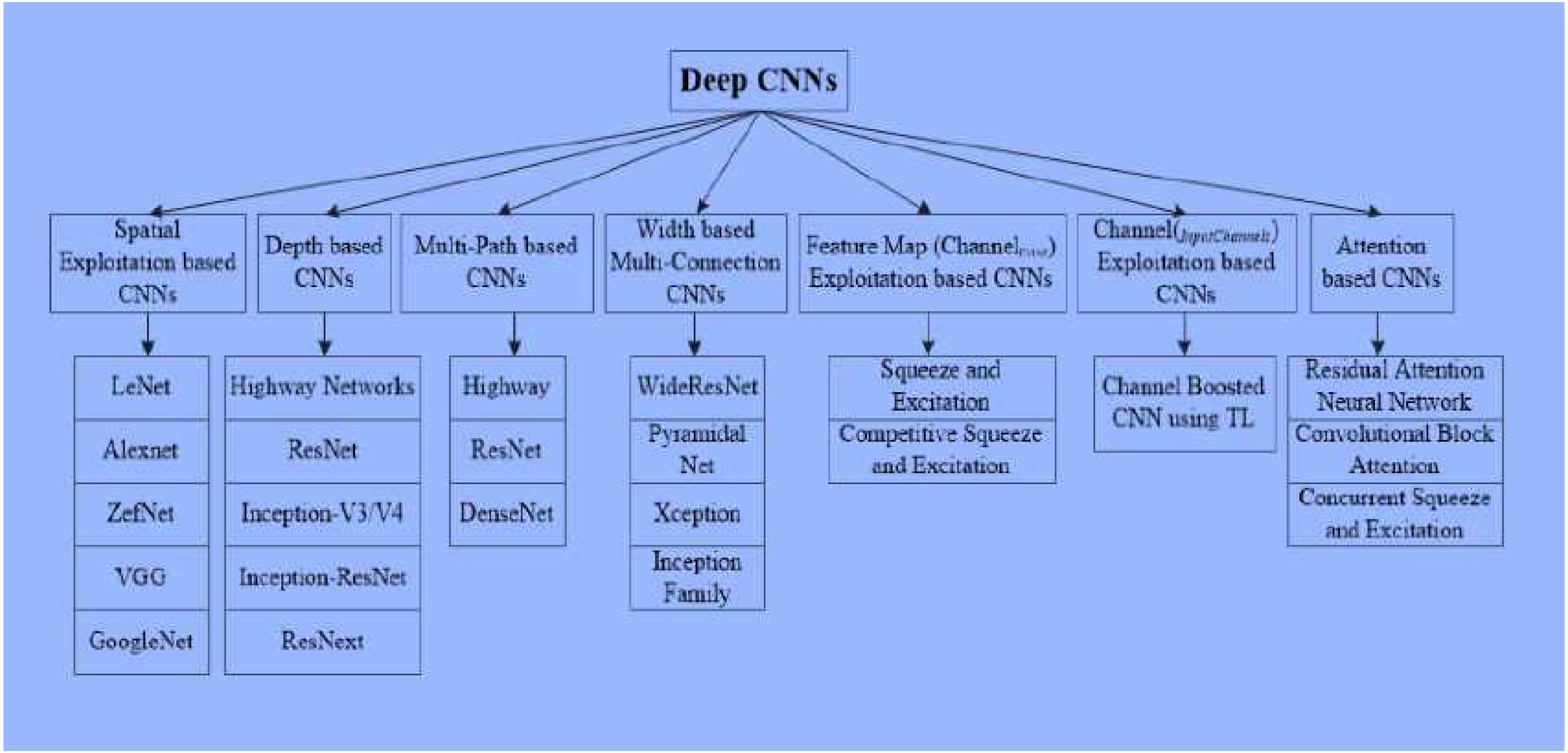
$$W_{ff} = 4096 \times 1000 = 4,096,000$$
$$B_{ff} = 1,000$$
$$P_{ff} = W_{ff} + B_{ff} = 4,097,000$$



# No. of Parameters and Tensor Sizes in AlexNet

| Layer Name | Tensor Size | Weights | Biases | Parameters |
|---|---|---|---|---|
| Input Image | 227x227x3 | 0 | 0 | 0 |
| Conv-1 | 55x55x96 | 34,848 | 96 | 34,944 |
| MaxPool-1 | 27x27x96 | 0 | 0 | 0 |
| Conv-2 | 27x27x256 | 614,400 | 256 | 614,656 |
| MaxPool-2 | 13x13x256 | 0 | 0 | 0 |
| Conv-3 | 13x13x384 | 884,736 | 384 | 885,120 |
| Conv-4 | 13x13x384 | 1,327,104 | 384 | 1,327,488 |
| Conv-5 | 13x13x256 | 884,736 | 256 | 884,992 |
| MaxPool-3 | 6x6x256 | 0 | 0 | 0 |
| FC-1 | 4096×1 | 37,748,736 | 4,096 | 37,752,832 |
| FC-2 | 4096×1 | 16,777,216 | 4,096 | 16,781,312 |
| FC-3 | 1000×1 | 4,096,000 | 1,000 | 4,097,000 |
| Output | 1000×1 | 0 | 0 | 0 |
| **Total** | | | | **62,378,344** |

# Other Deep CNN Architectures



# Deep Learning

3 6

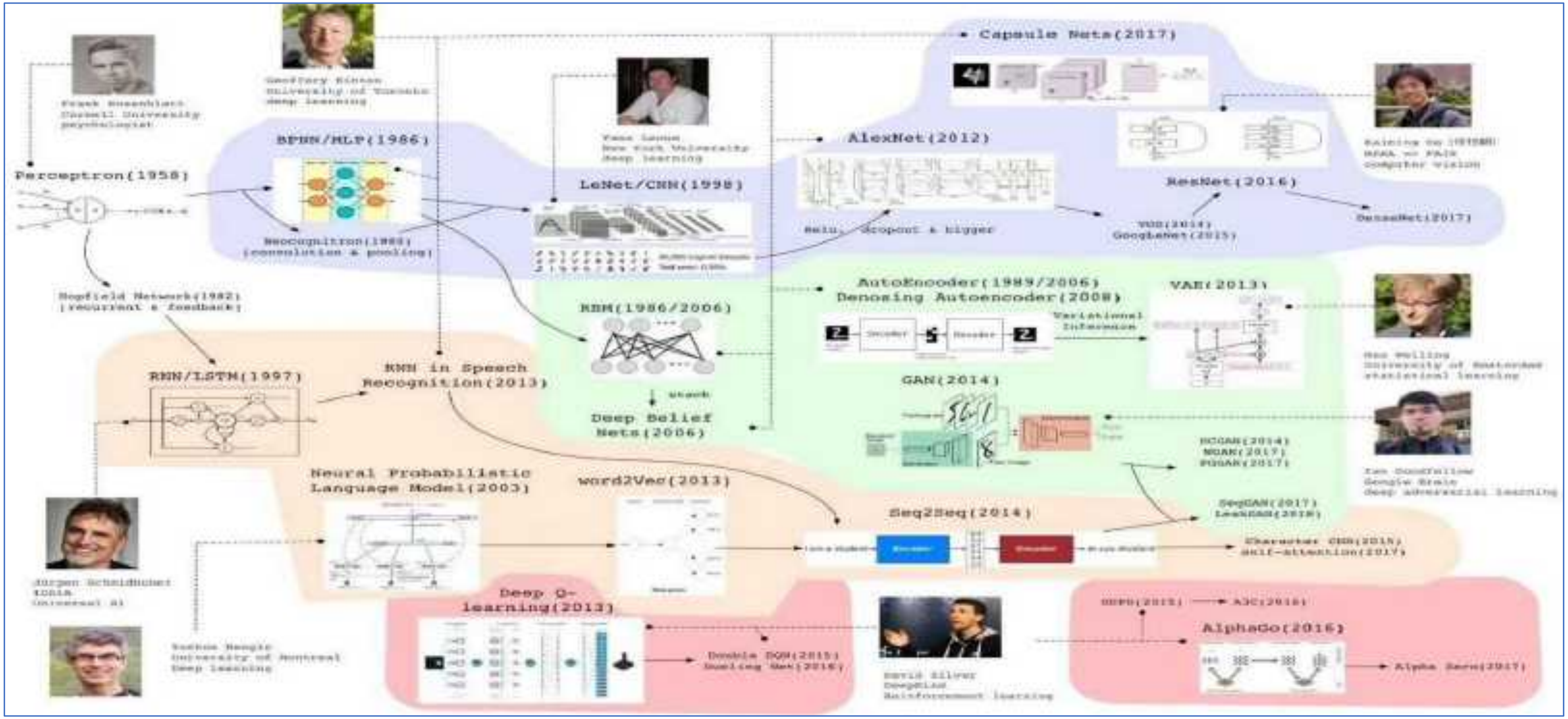
1 Deep Learning

2 Convolution

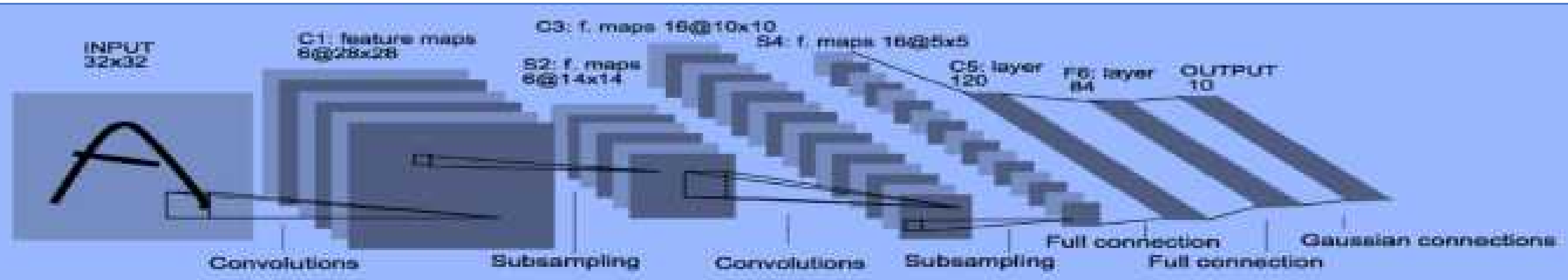
3 CNN Deep Learning

4 Deep Learning and Machine Learning

# Deep Learning Models and Algorithms



# LeNet (1998) – The Origin of Convolutional Neural Network



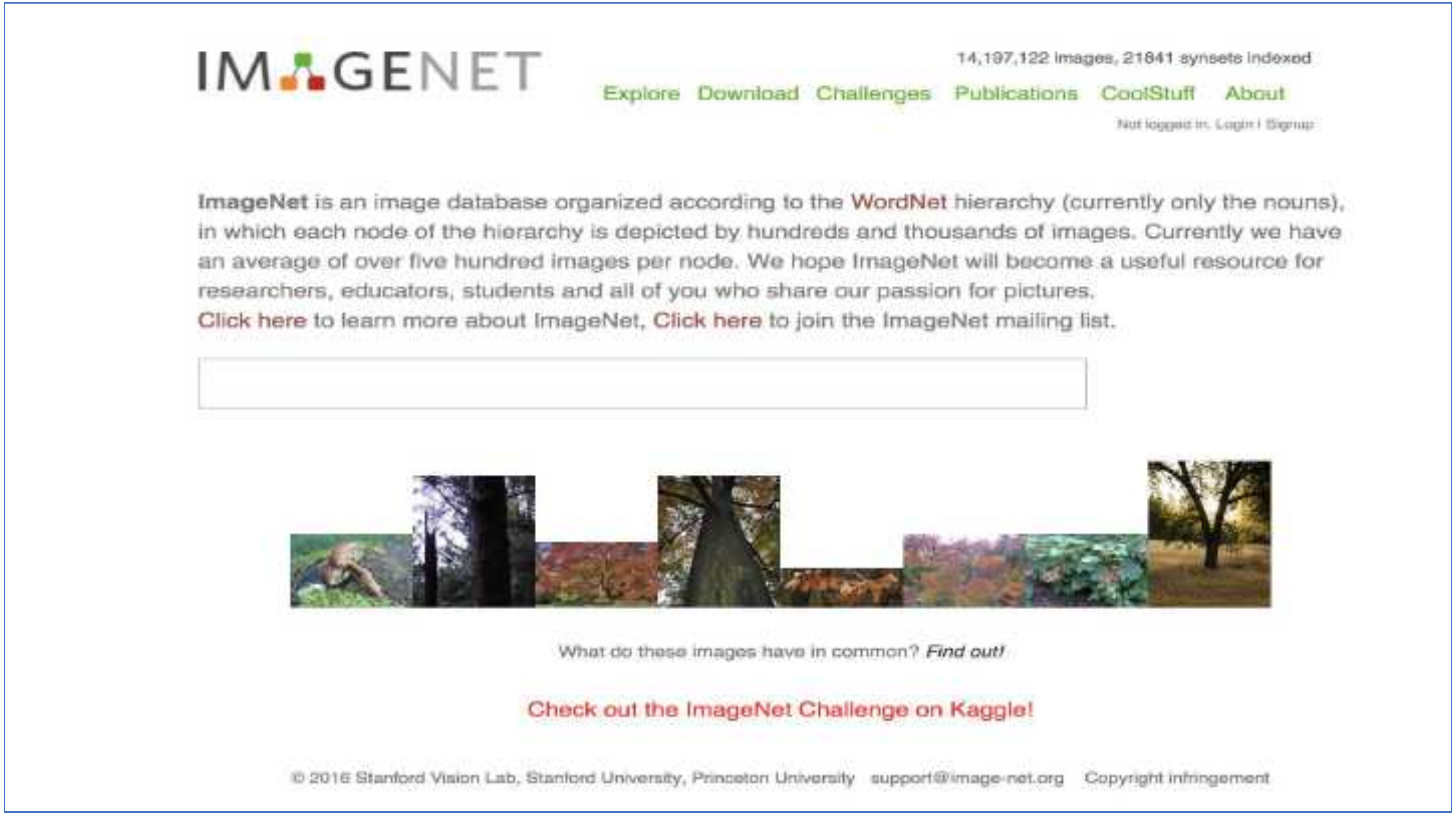

**CNN called LeNet by Yann LeCun (1998)**

| Characteristics | Key Contributions |
| --- | --- |
| • Repeat of Convolution – Pooling – Non Linearity<br>• Average pooling<br>• Sigmoid activation for the intermediate layer<br>• tanh activation at F6<br>• 5x5 Convolution filter<br>• 7 layers and less than 1M parameters | • Use of convolution to extract spatial features<br>• Subsample using spatial average of maps<br>• Sparse connection matrix between layers to avoid large computational cost |

# LeNet Configurations

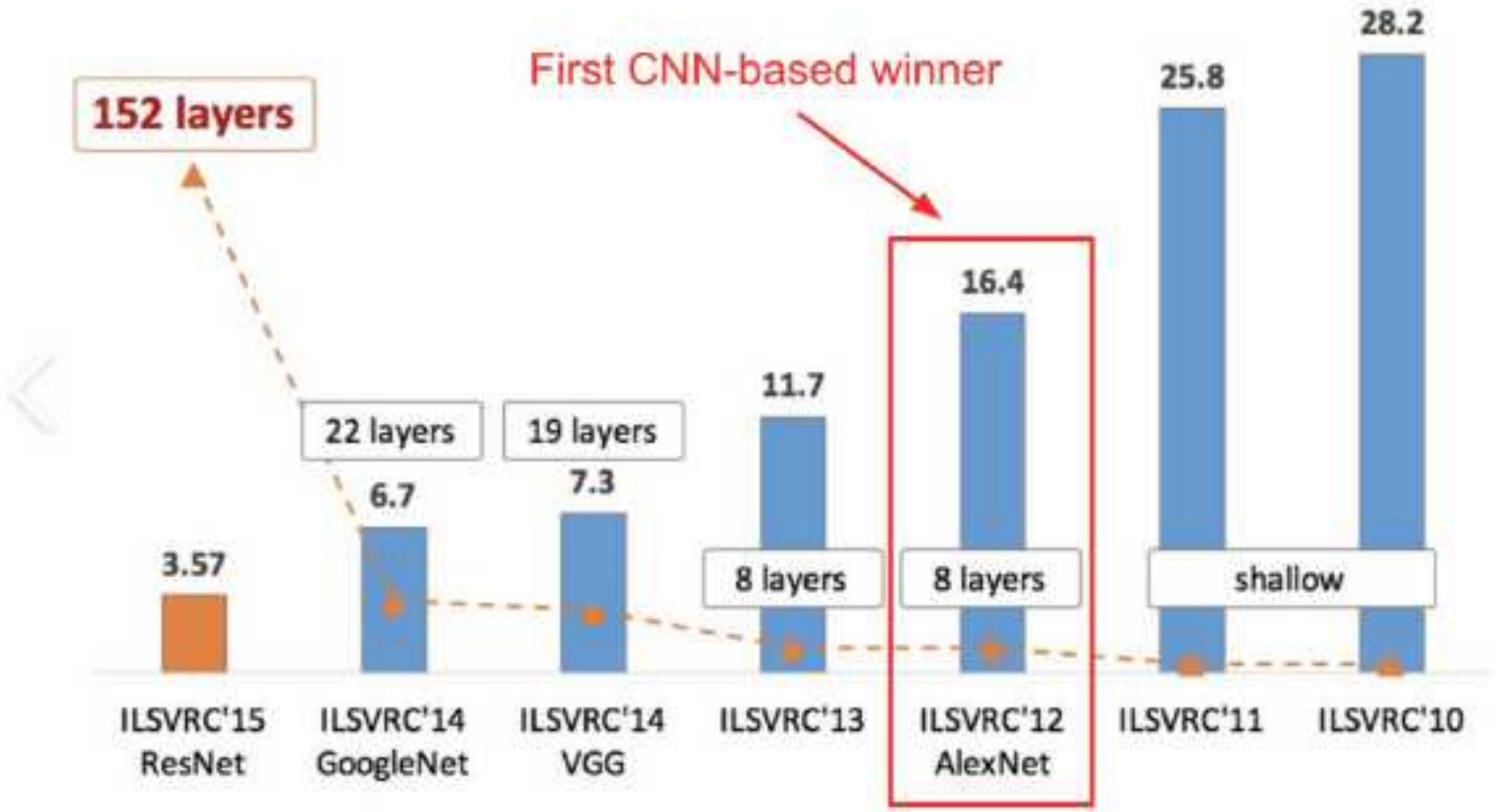
| Layer | | Feature Map | Size | Kernel Size | Stride | Activation |
|---|---|---|---|---|---|---|
| Input | Image | 1 | 32x32 | - | - | - |
| 1 | Convolution | 6 | 28x28 | 5x5 | 1 | tanh |
| 2 | Average Pooling | 6 | 14x14 | 2x2 | 2 | tanh |
| 3 | Convolution | 16 | 10x10 | 5x5 | 1 | tanh |
| 4 | Average Pooling | 16 | 5x5 | 2x2 | 2 | tanh |
| 5 | Convolution | 120 | 1x1 | 5x5 | 1 | tanh |
| 6 | FC | - | 84 | - | - | tanh |
| Output | FC | - | 10 | - | - | softmax |

# From Big Data to Deep Learning Models: IMAGENET Challenge- Computer Vision World Cup



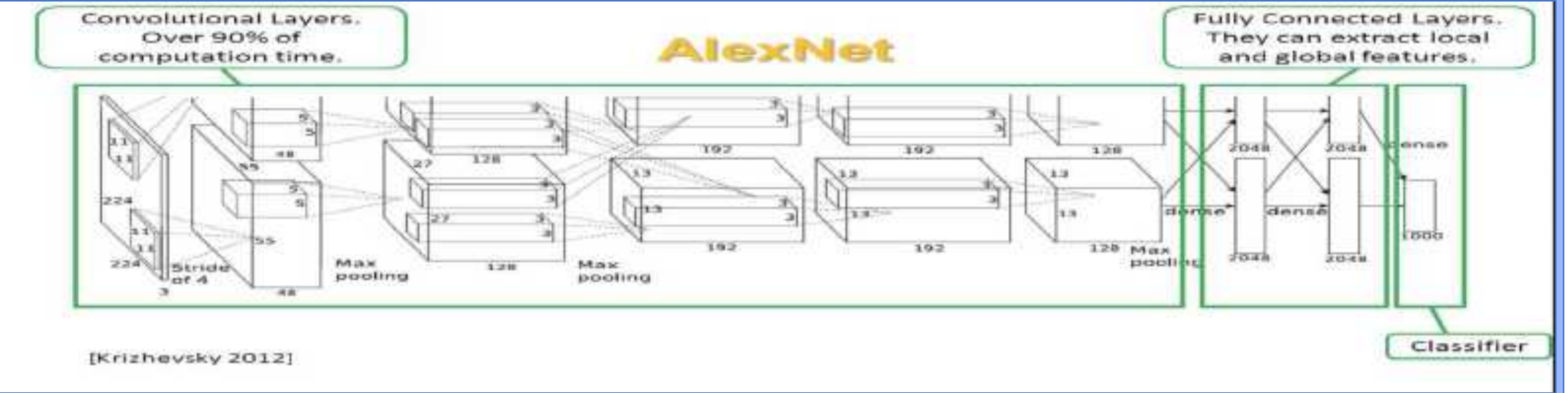
# ImageNet Large Scale Visual Recognition Challenge (ILSVRC)

# Some Popular Algorithms



# AlexNet

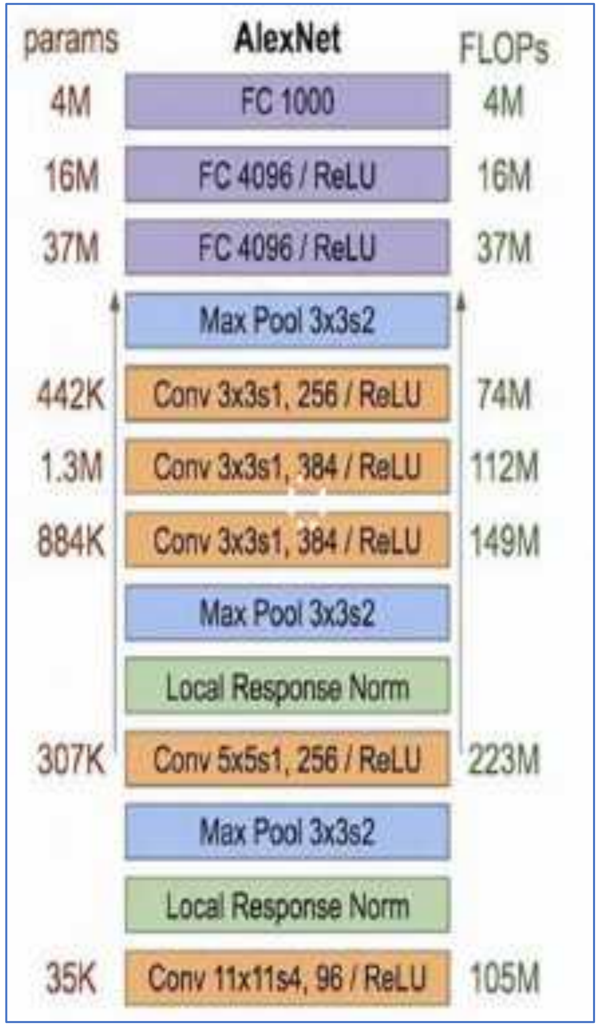
- Similar framework to LeCun'98 but:
  - Bigger model (7 hidden layers, 650,000 units, 60,000,000 params)
  - More data ($10^6$ vs. $10^3$ images)
  - GPU implementation (50x speedup over CPU)
    - Trained on two GPUs for a week



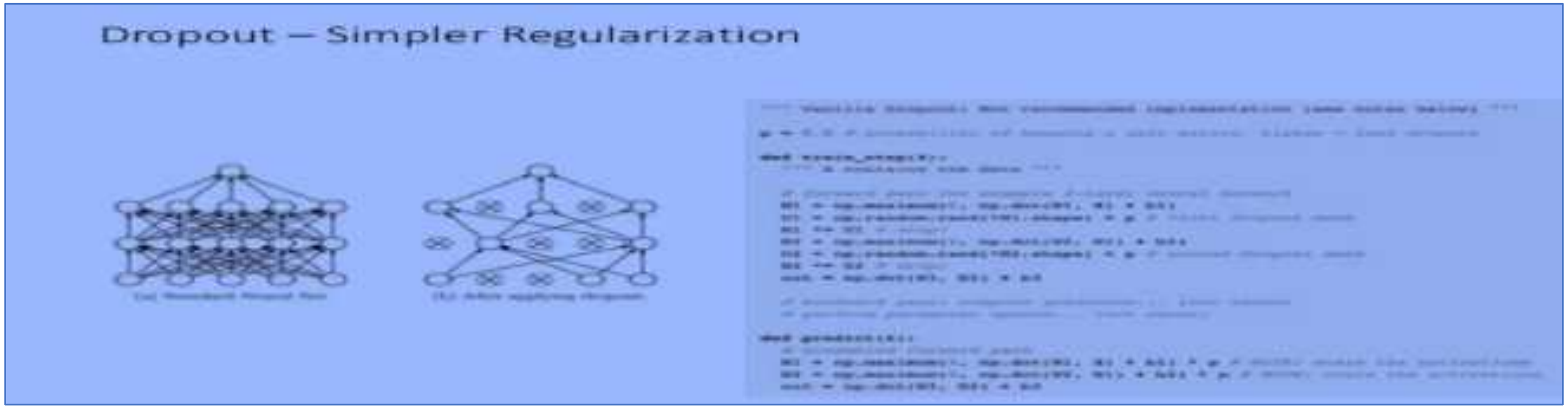

A. Krizhevsky, I. Sutskever, and G. Hinton,
ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012

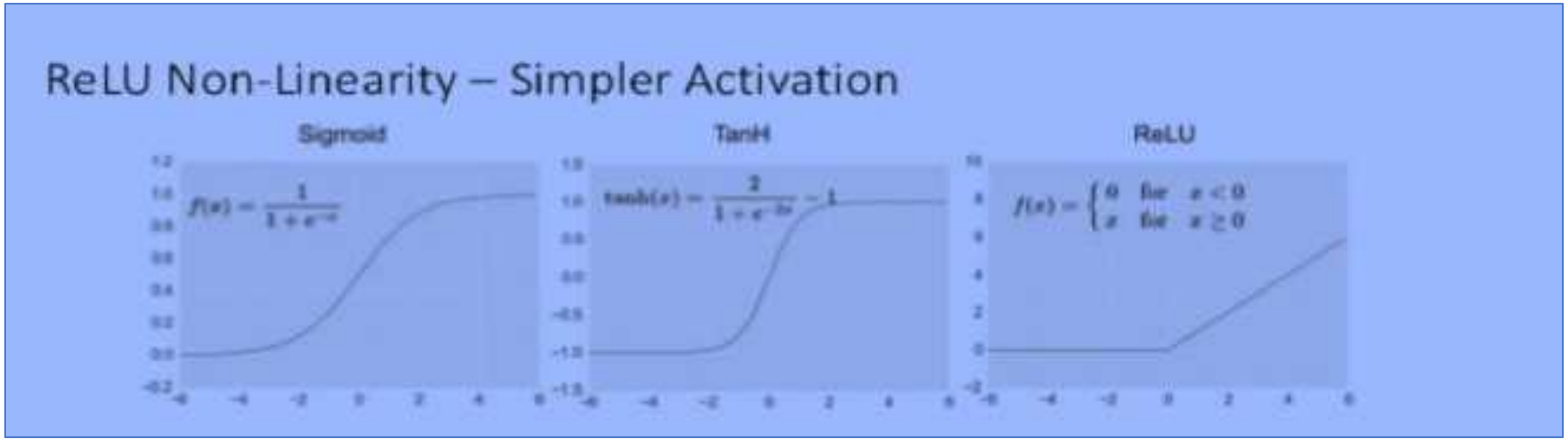
# AlexNet: Network and Data Augmentation


| Characteristics | Key Contributions |
| --- | --- |
| - 11x11, 5x5 and 3x3 Convolutions<br>- Max pooling<br>- 3 FC layers<br>- 60 Million parameters | • GPU and training in parallel<br>• ReLu Activation<br>• Dropout regularization<br>• Image Augmentation |

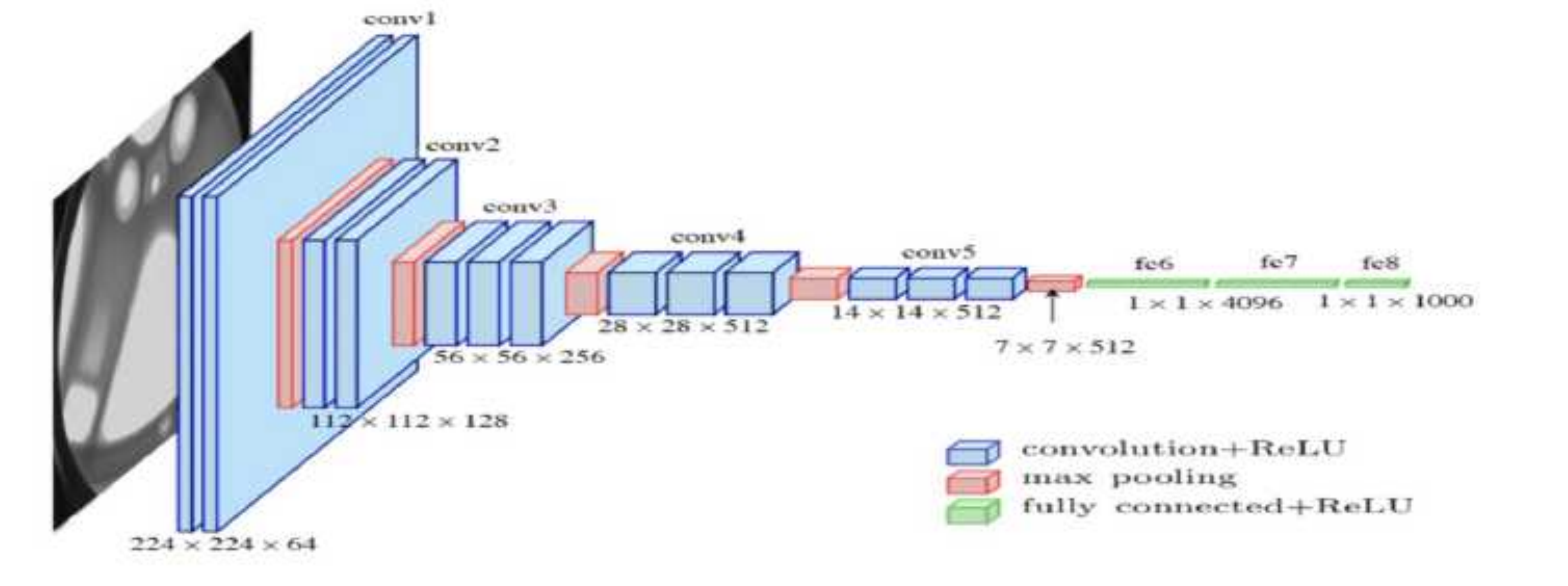


# Computation and non-linearity: Dropout and ReLU



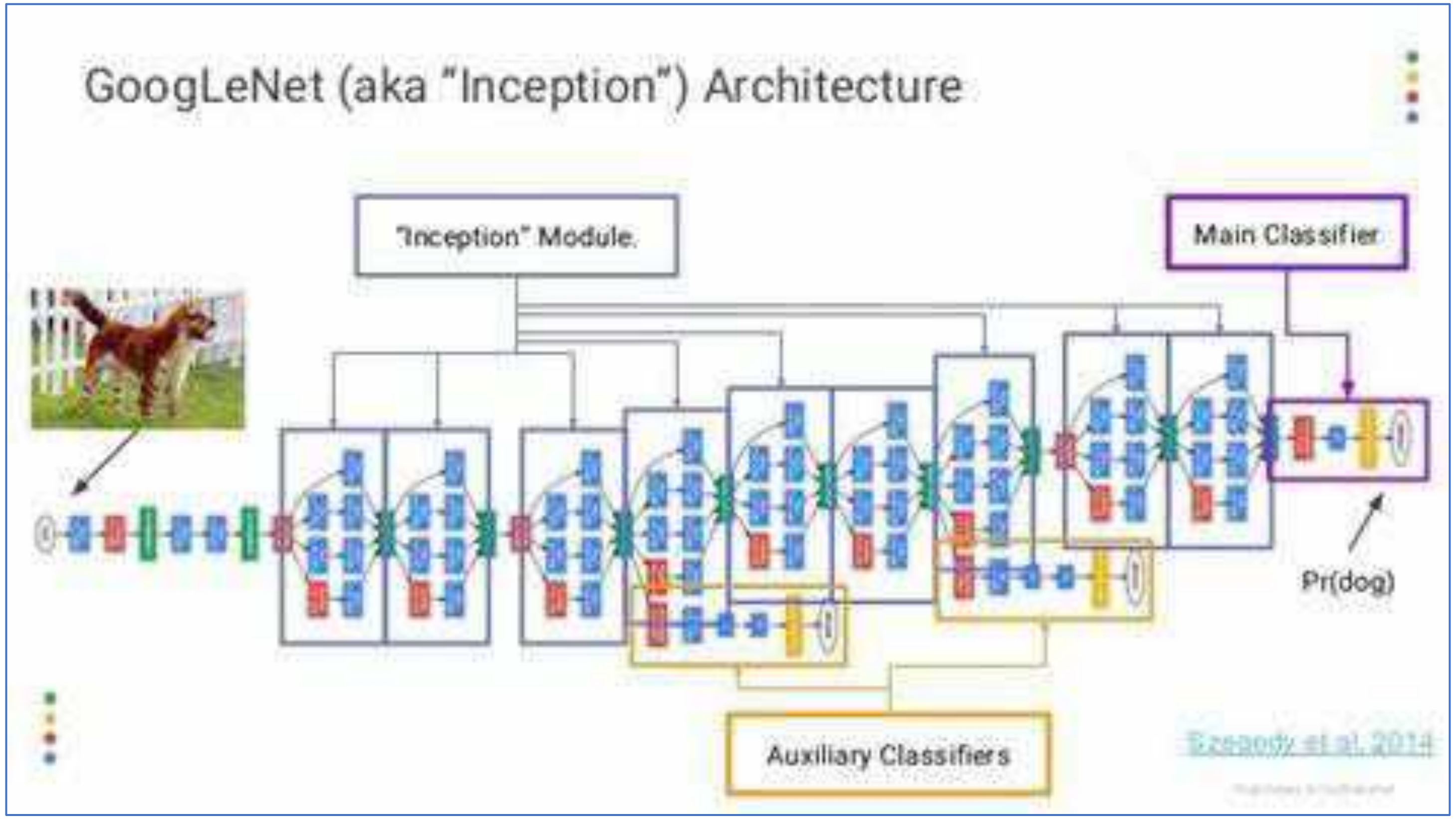



# VGG Network
# - Visual Geometry Group, University of Oxford



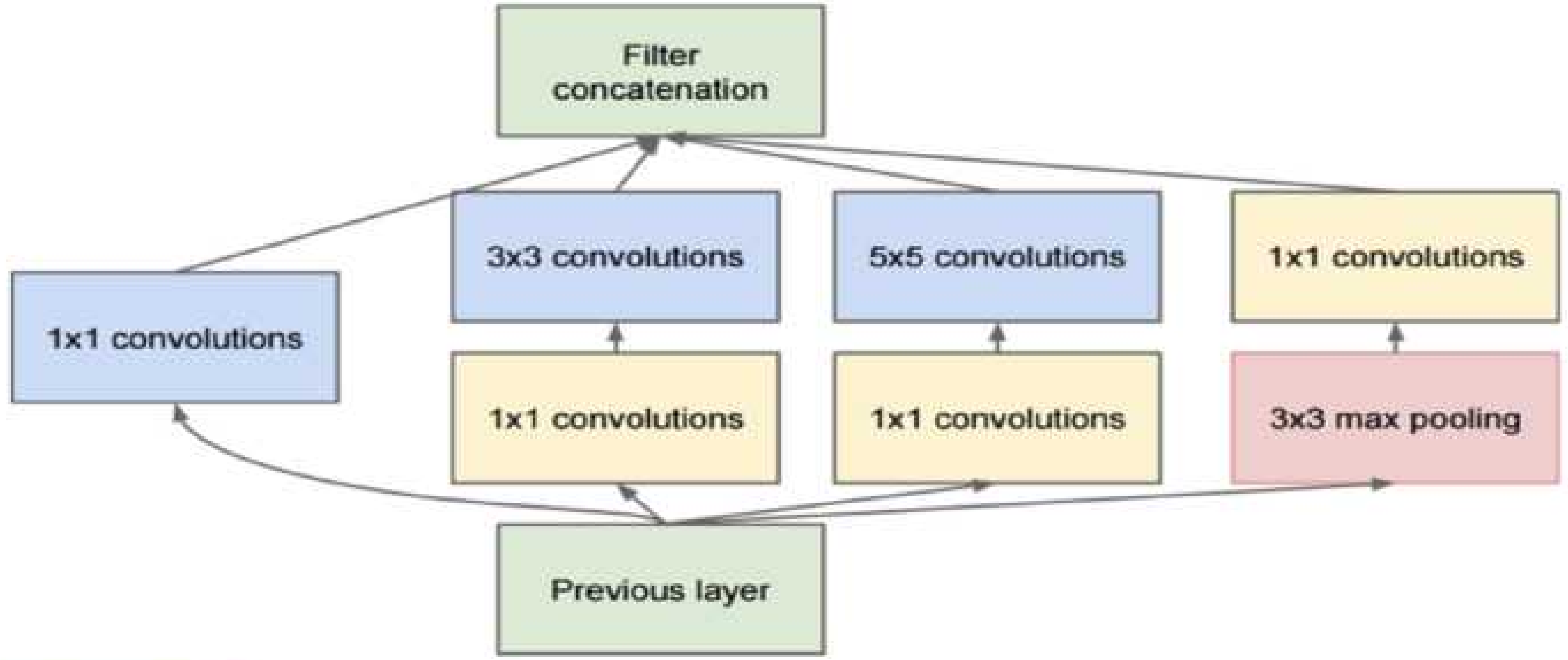

*The main contribution of VGG is to show that classification/localisation accuracy can be improved by increasing the depth of CNN in spite of using small receptive fields in the layers.*

# GoogleNet InceptionNet




Convolution
Pooling
Softmax
Other

# GoogleNet Inception Module

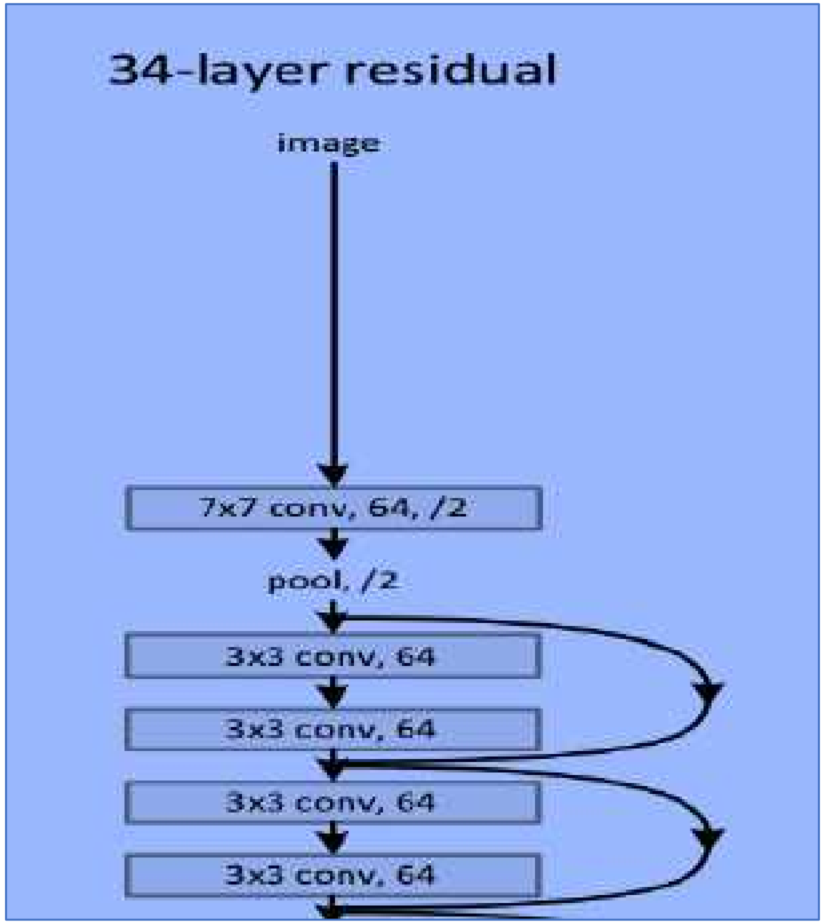



Full Inception module

*Its main contribution was the development of an Inception Module that dramatically reduced the number of parameters in the network (4M, compared to AlexNet with 60M).*

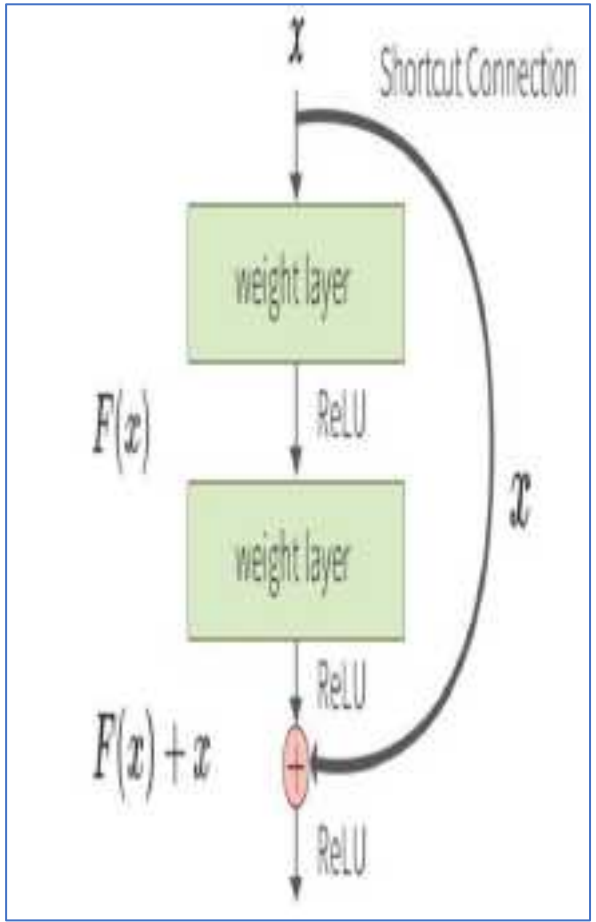
# ResNet

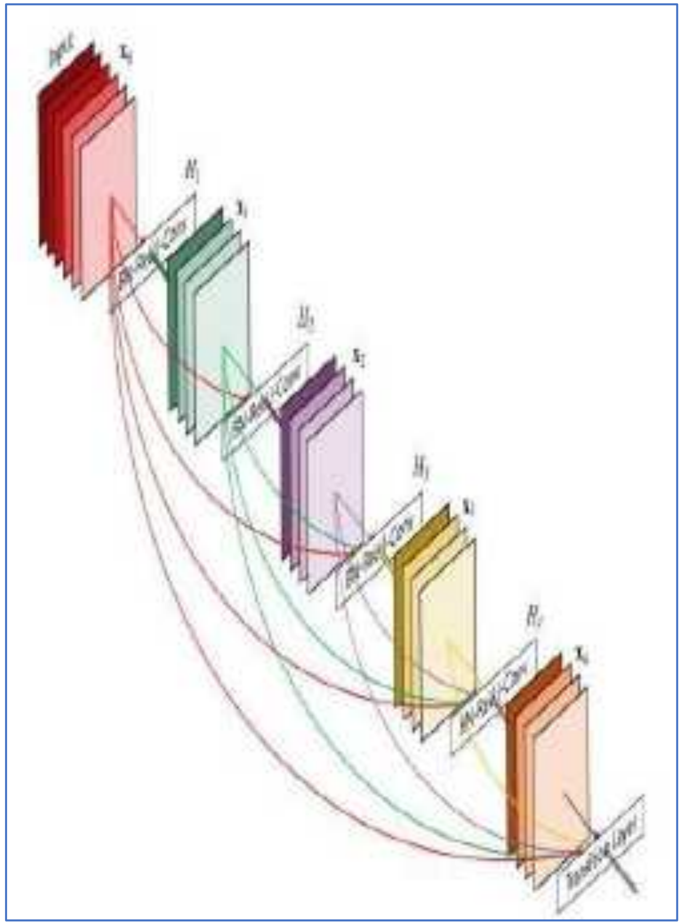
- A residual neural network (ResNet) is an artificial neural network (ANN) of a kind that builds on constructs known from pyramidal cells in the cerebral cortex. Residual neural networks do this by utilizing skip connections, or short-cuts to jump over some layers.



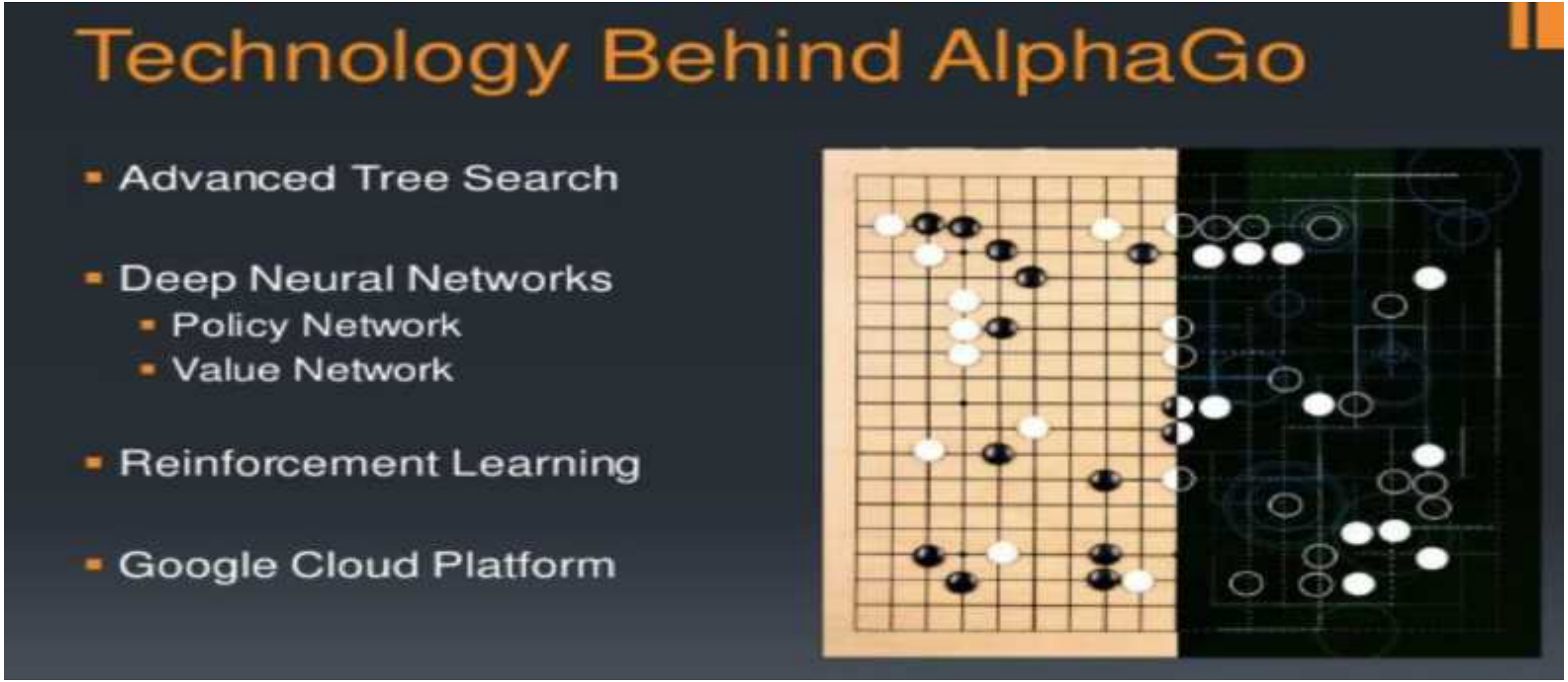
# ResNet 152 and DenseNet

- When a net is **very** deep it becomes very difficult for gradients to propagate backwards all the way. Skip connections offer "short cuts" for gradients to propagate further and allow for efficient training of very deep nets.

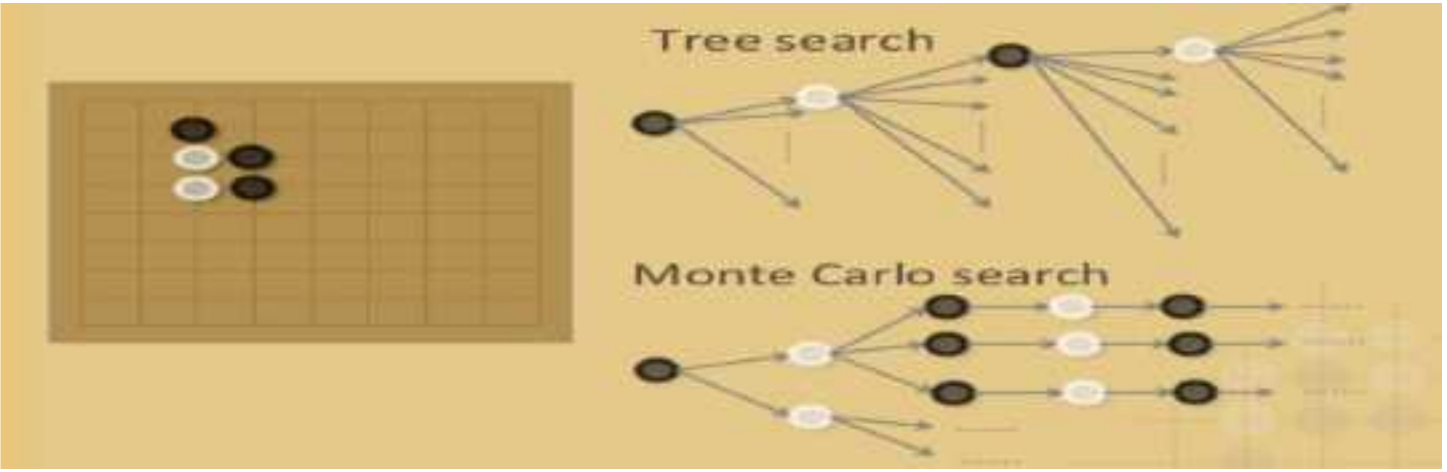


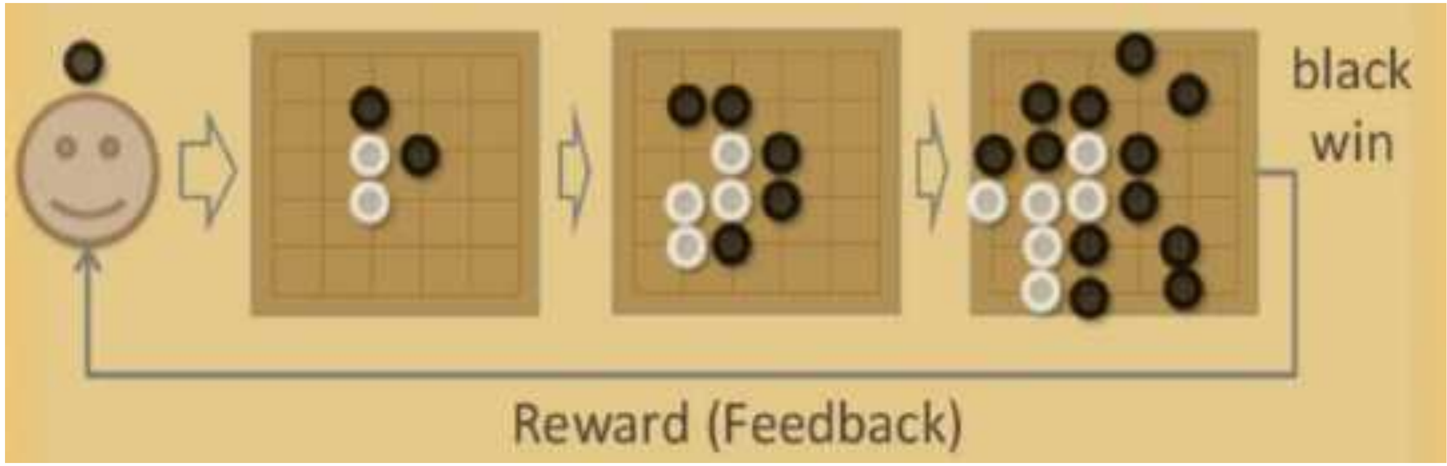



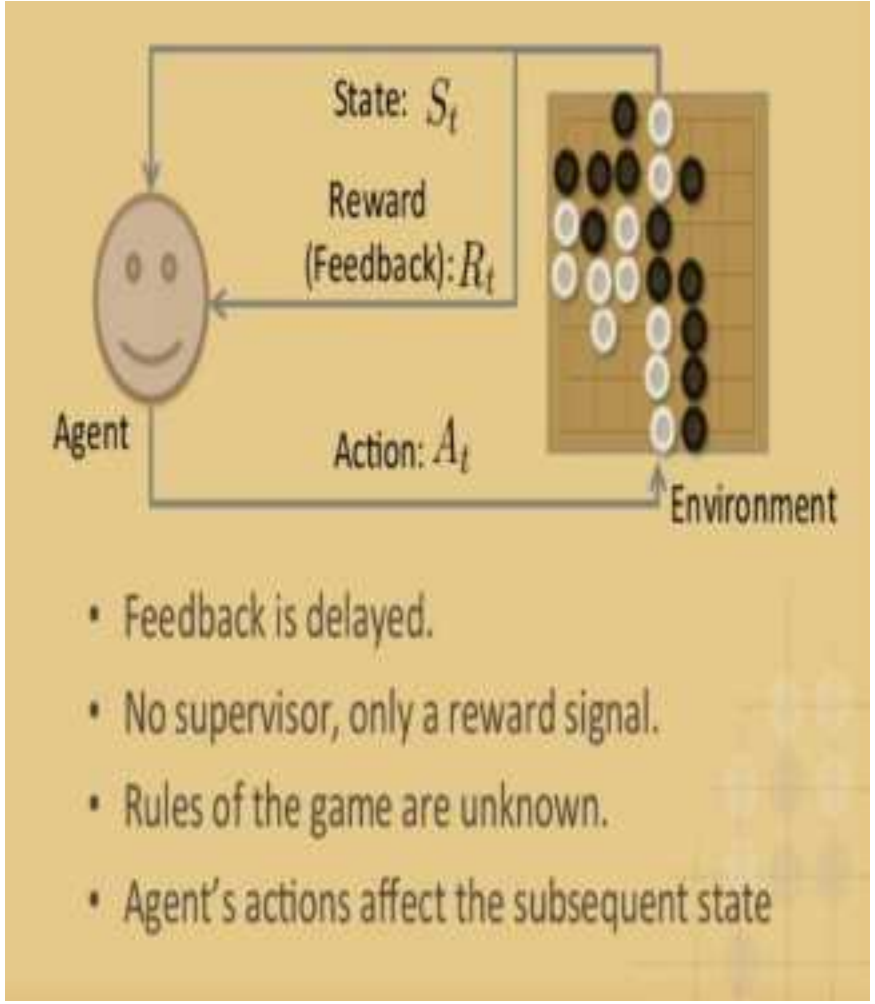
# AlphaGo and AlphaGo Zero Technologies



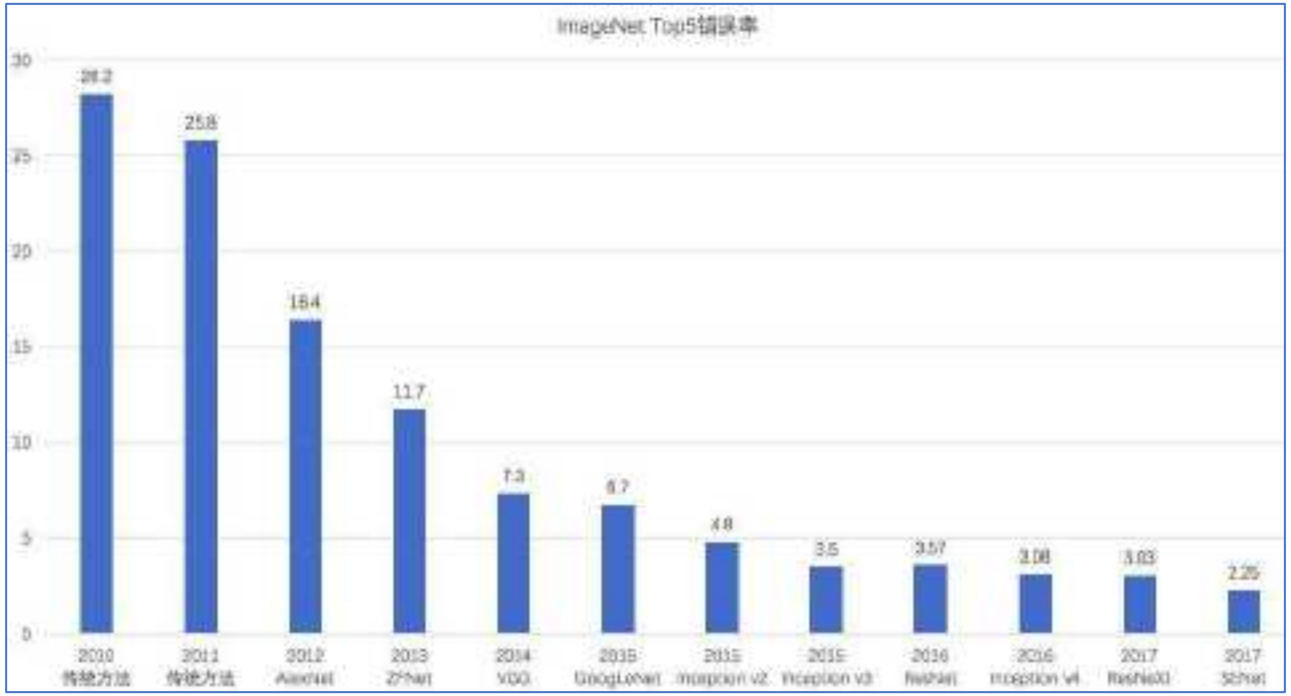
# Monte Carlo and Reinforcement Learning

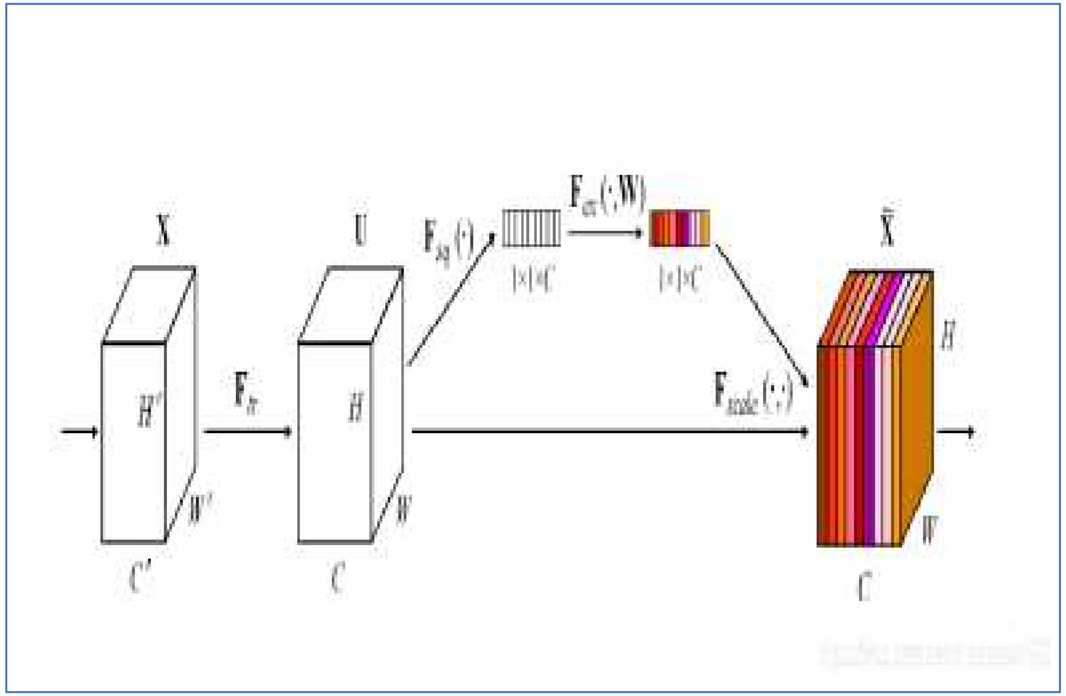


# SE Net




**Squeeze-and-excitation networks (2017)**

# AmoebaNet (2018): Neural Architecture Search

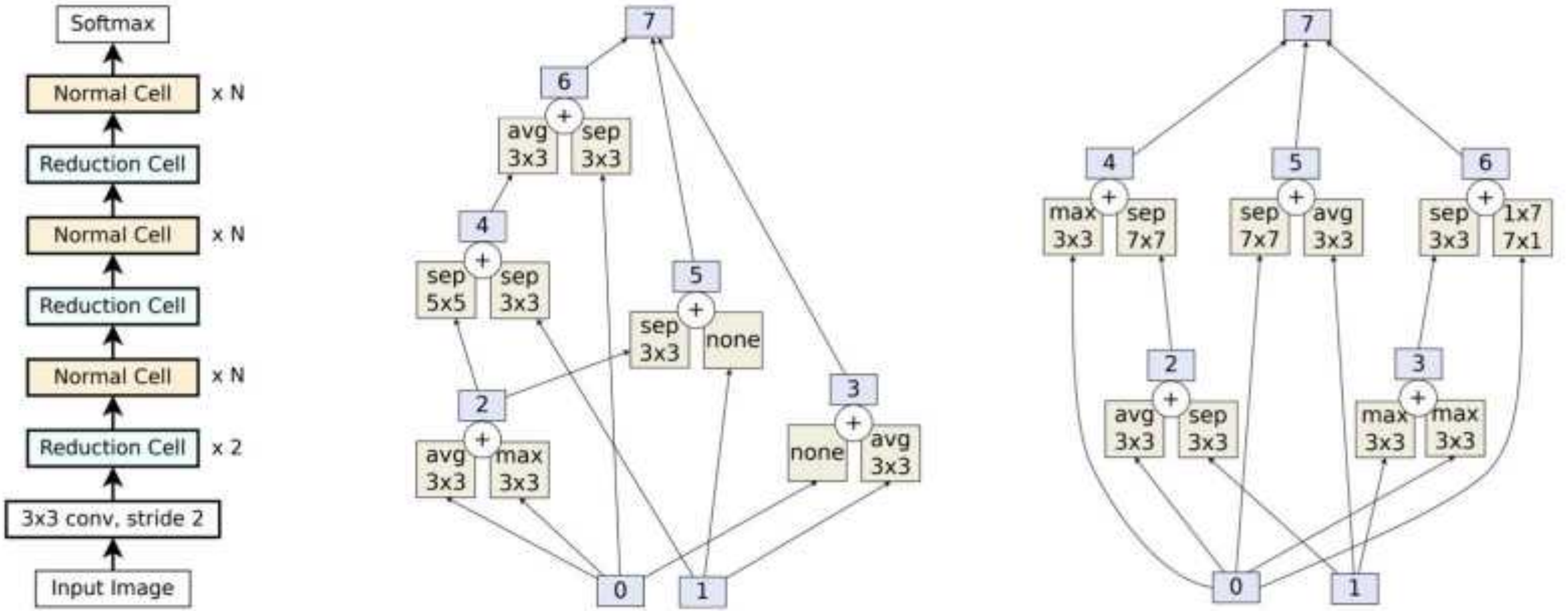
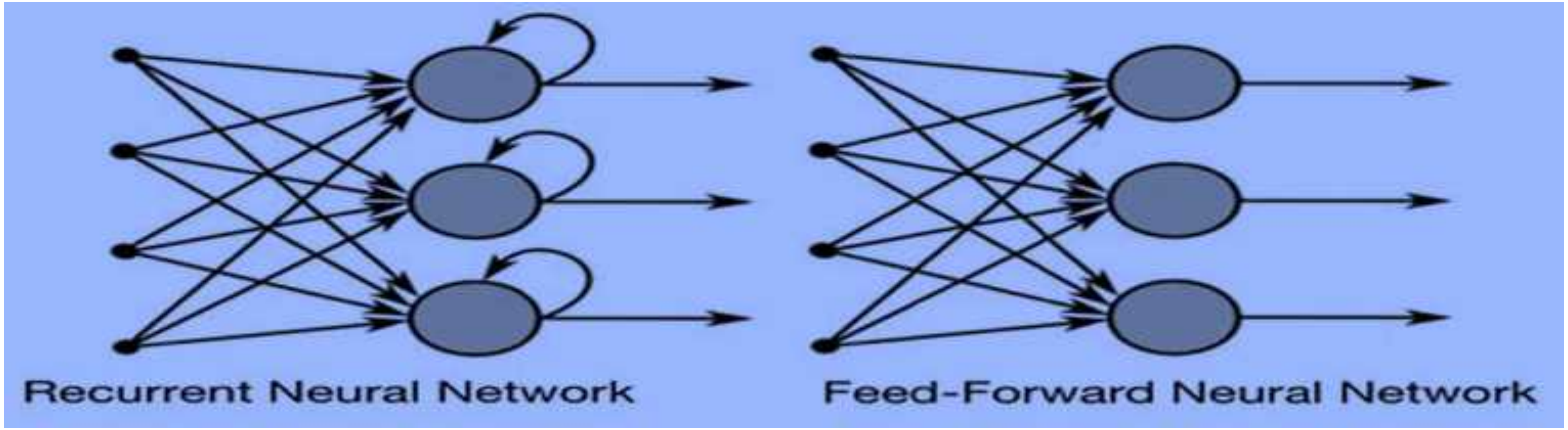



*Figure from Real et al. (2019)*

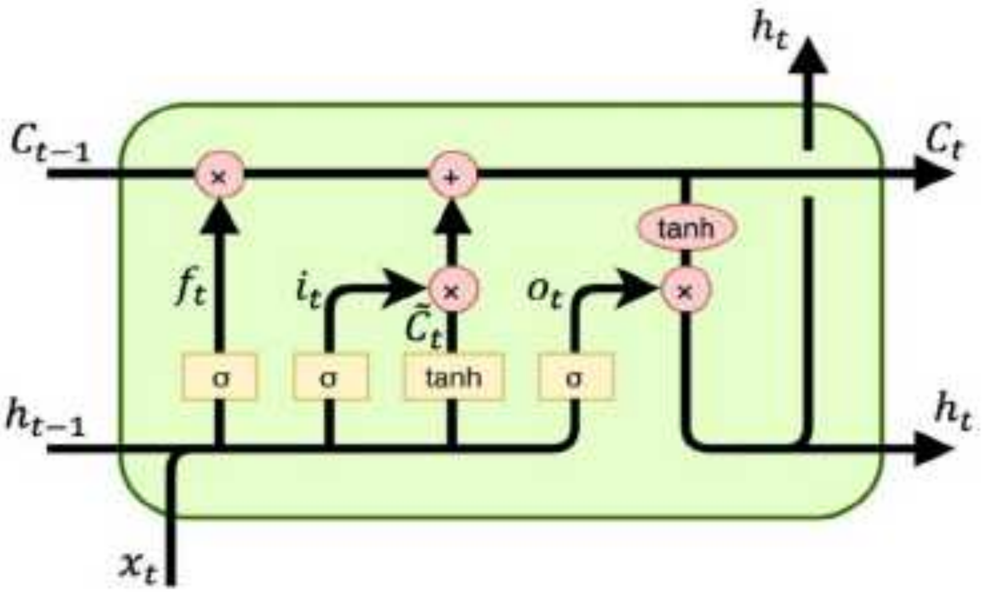
**Architecture found by evolution**

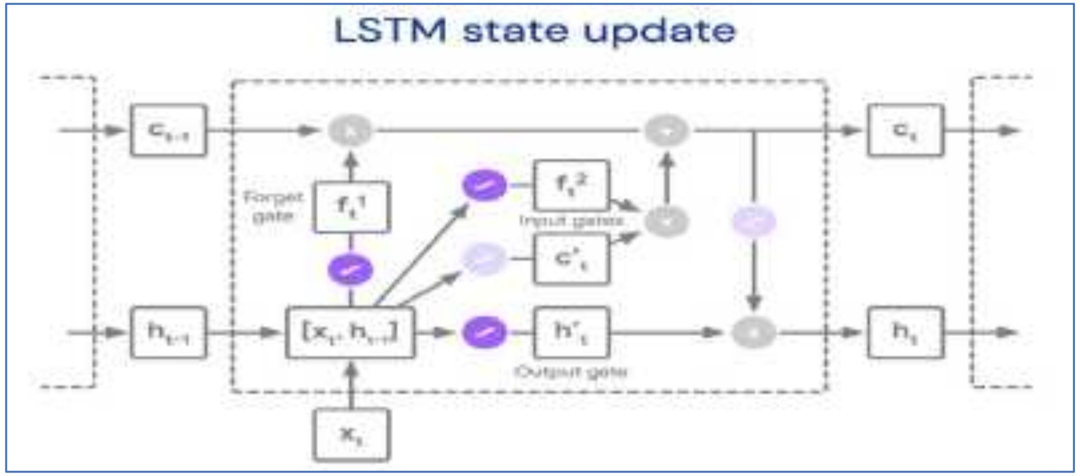
# RNN and LSTM




A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs.

# Long Short-Term Memory




$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

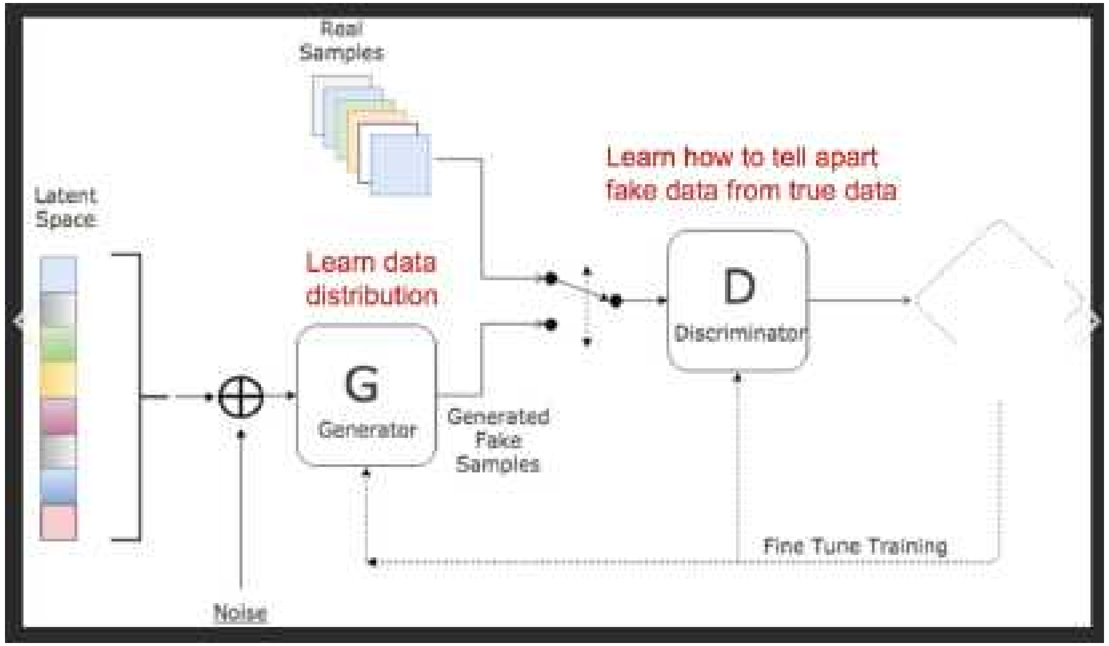
$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

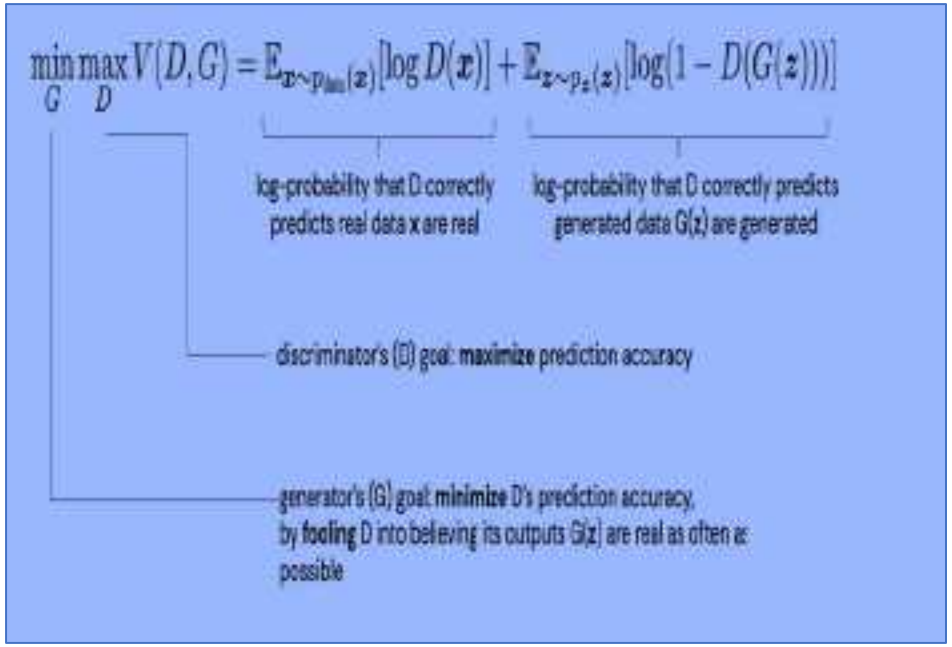
$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

$$h_t = o_t \odot \tanh(C_t)$$

- $x_t \in \mathbb{R}^d$: input vector to the LSTM unit
- $f_t \in \mathbb{R}^h$: forget gate's activation vector
- $i_t \in \mathbb{R}^h$: input/update gate's activation vector
- $o_t \in \mathbb{R}^h$: output gate's activation vector
- $h_t \in \mathbb{R}^h$: hidden state vector also known as output vector of the LSTM unit
- $\tilde{c}_t \in \mathbb{R}^h$: cell input activation vector
- $c_t \in \mathbb{R}^h$: cell state vector

# GAN Unsupervised Network

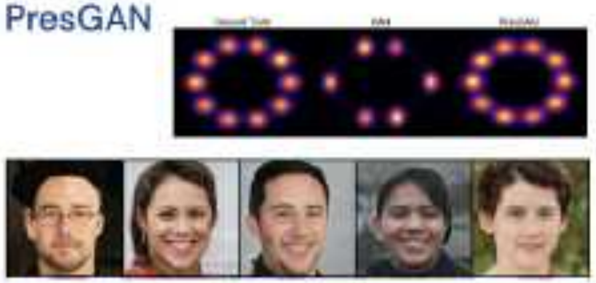



Generative Adversarial Networks (GAN) is one of the most promising recent developments in Deep Learning. GAN, introduced by Ian Goodfellow in 2014, attacks the problem of unsupervised learning by training two deep networks, called Generator and Discriminator, that compete and Cooperate with each other.

# Recent GANs Proposed in 2019

## PresGAN

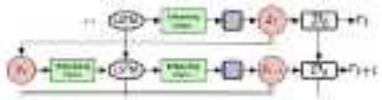


Dieng et al. **Prescribed Generative Adversarial Networks.** *arXiv:1910.04302* (2019)

## Effectiveness of Adam on Cycles

Gemp & McWilliams. **The Unreasonable Effectiveness of Adam on Cycles.** *NeurIPS Smooth Games Optimization and Machine Learning Workshop* (2019)

## ScratchGAN



de Masson d'Autume et al. **Training language GANs from Scratch.** *Neural Information Processing Systems* (2019)

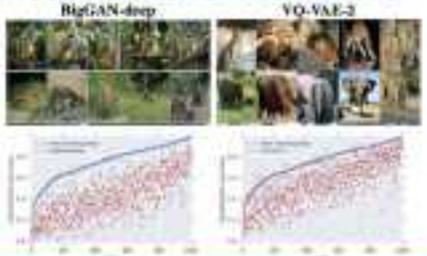
## Copy-Pasting GAN

Arandjelovic & Zisserman. **Object Discovery with a Copy-Pasting GAN.** *arXiv:1905.11369* (2019)

## Improved SPIRAL

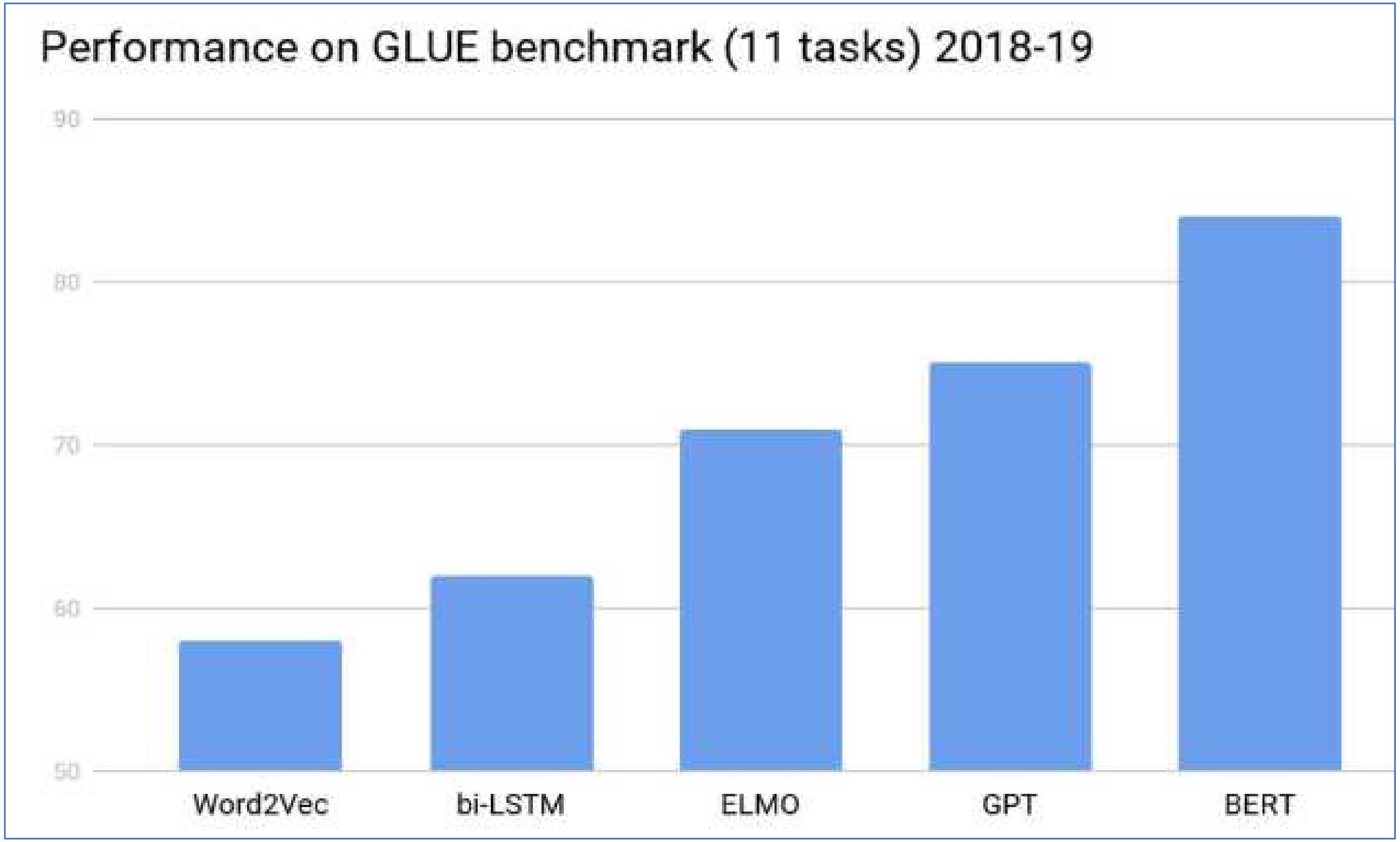
Mellor et al. **Unsupervised Doodling and Painting with Improved SPIRAL.** *arXiv:1910.01007* (2019)

## Classification Accuracy Score




Ravuri & Vinyals. **Classification Accuracy Score for Conditional Generative Models.** *Neural Information Processing Systems* (2019)

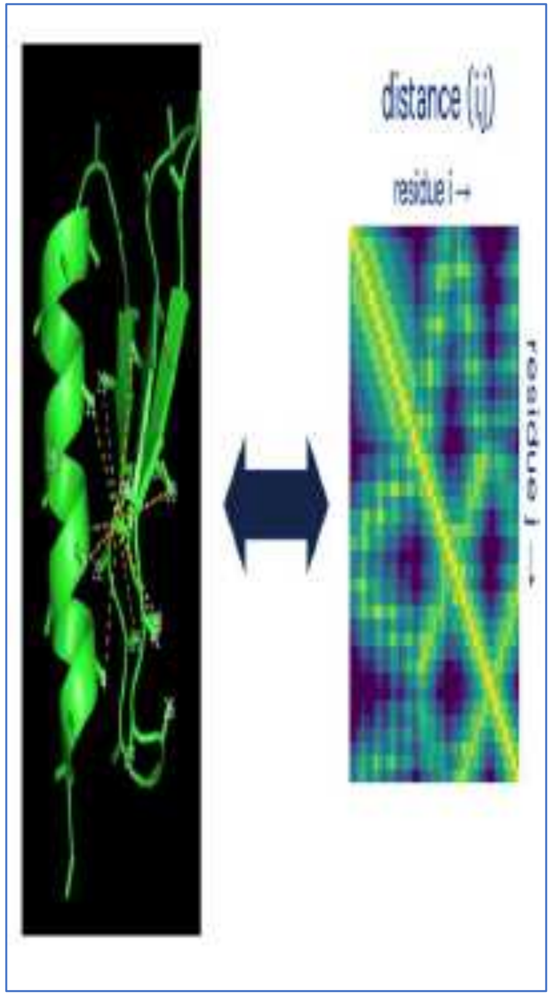
# GPT3 General Pre-trained Transformer 3-2019



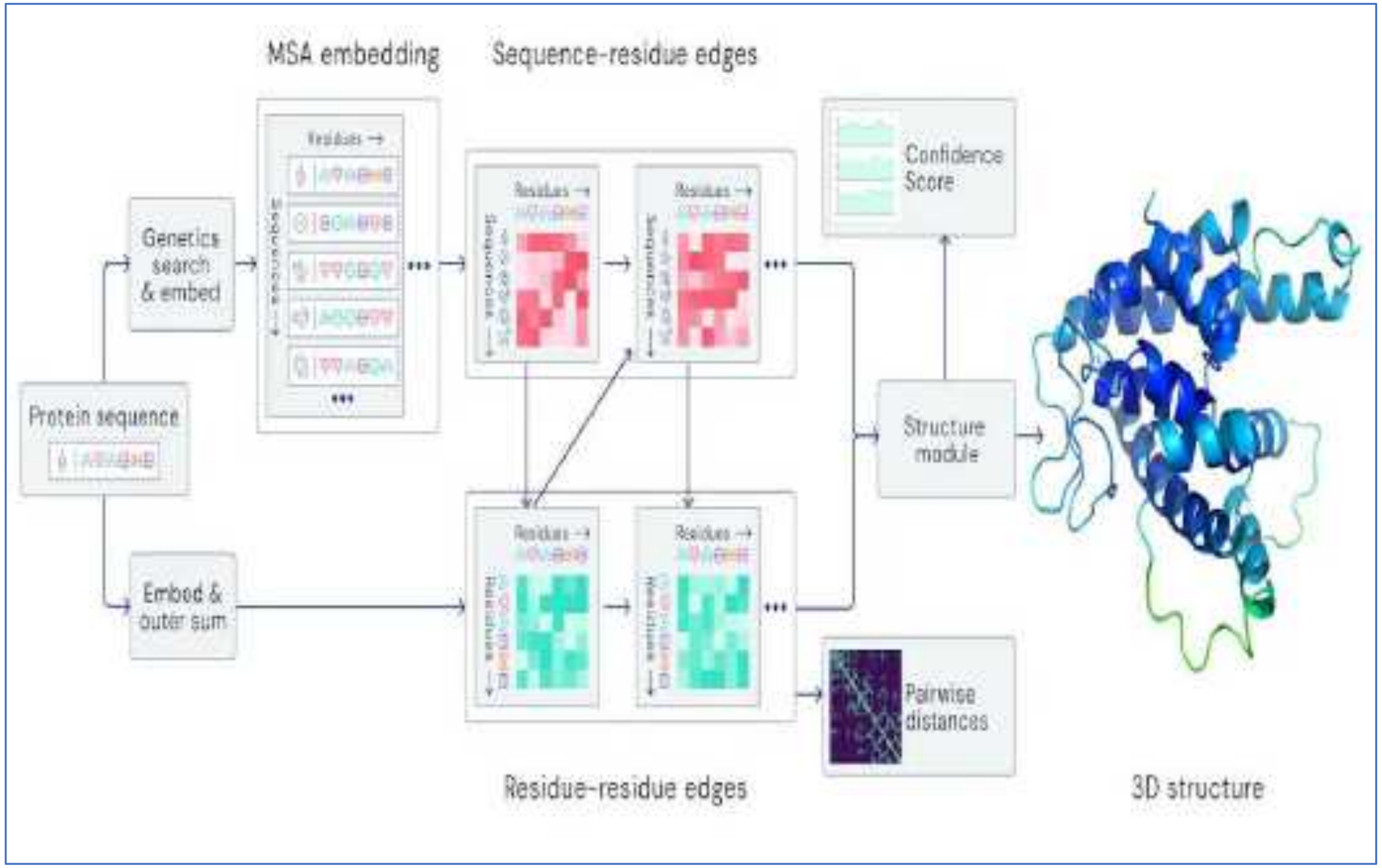

GPT-2 and 3

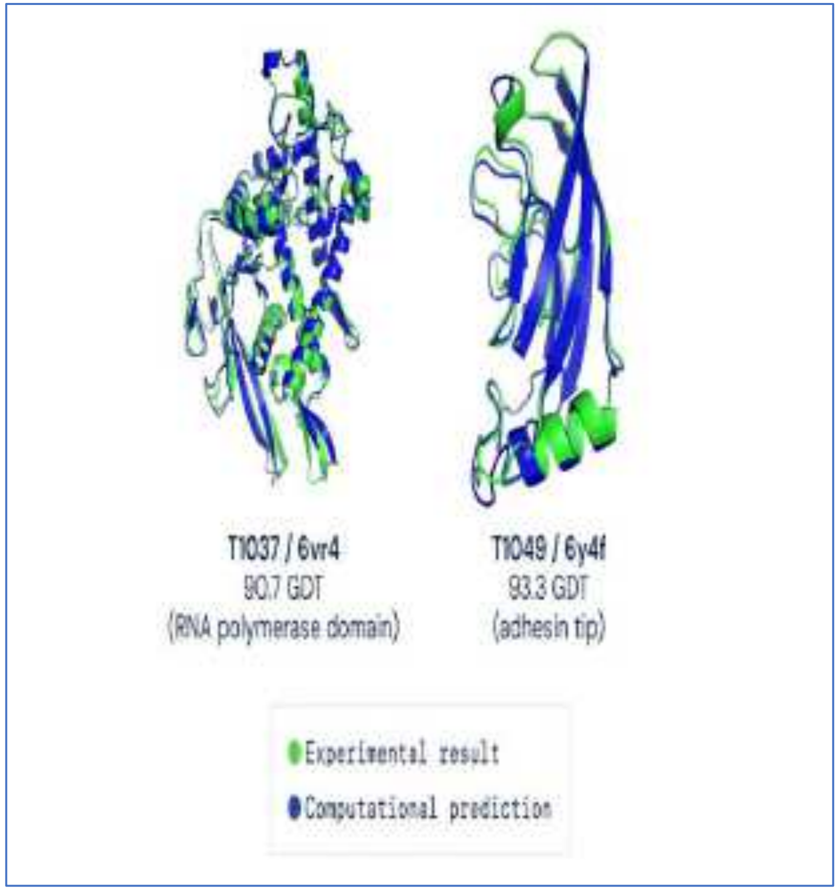
GPT-3 2020: Largest ANN. 175B:1.5B Parameters

# AlphaFold - 2020

# AlphaFold -2020



# Unsupervised Learning - 2020

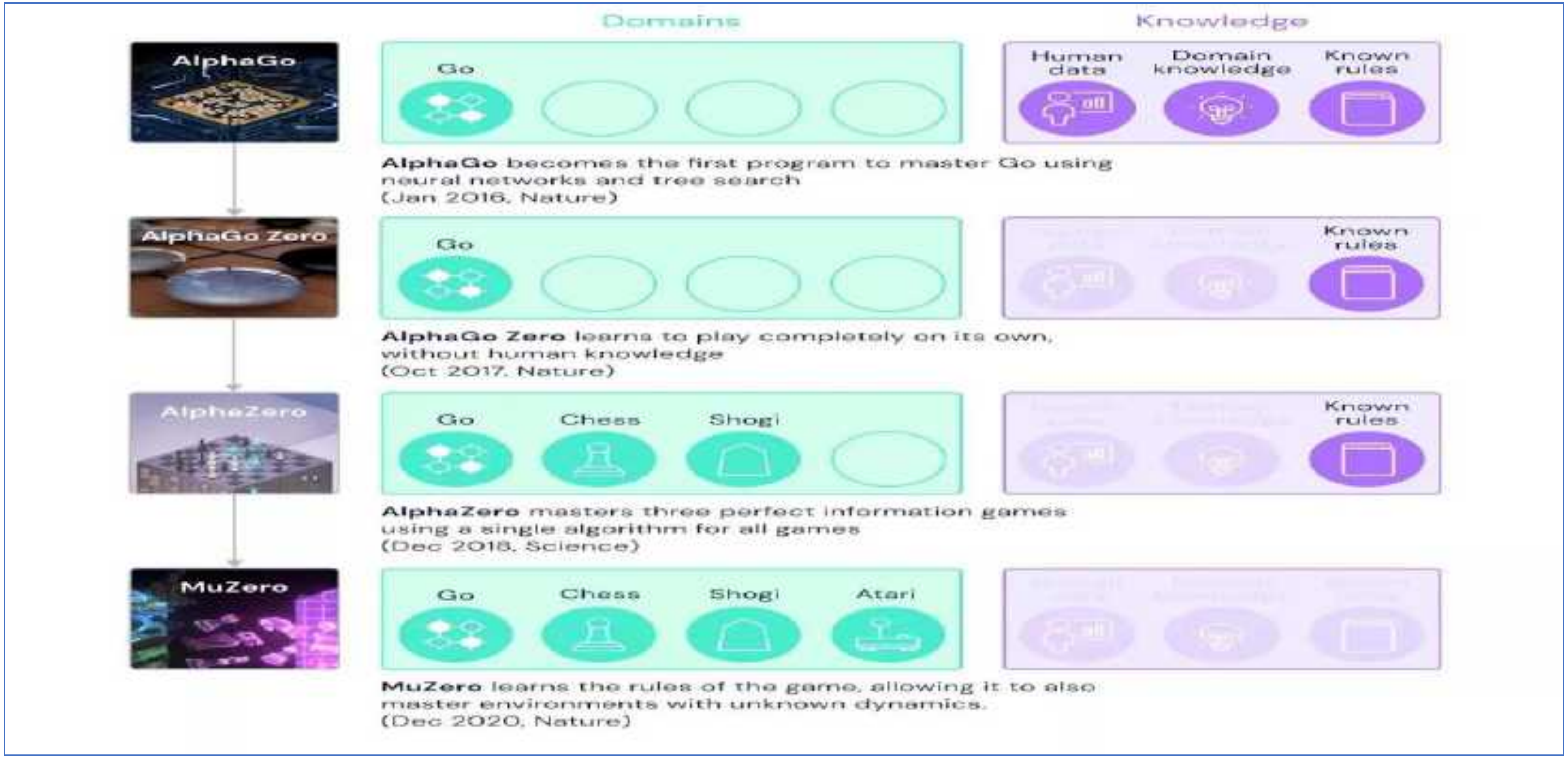
"I always knew unsupervised learning was the right thing to do
— Geoff Hinton

"Basically it's the idea of learning to represent the world before learning a task — and this is what babies do
— Yann LeCun

"And so if we can build models of the world where we have the right abstractions, where we can pin down those changes to just one or a few variables, then we will be able to adapt to those changes because we don't need as much data, as much observation in order to figure out what has changed.
— Yoshua Bengio

Turing Award winners at AAAI 2020

# MuZero – Dec 2020 Nature

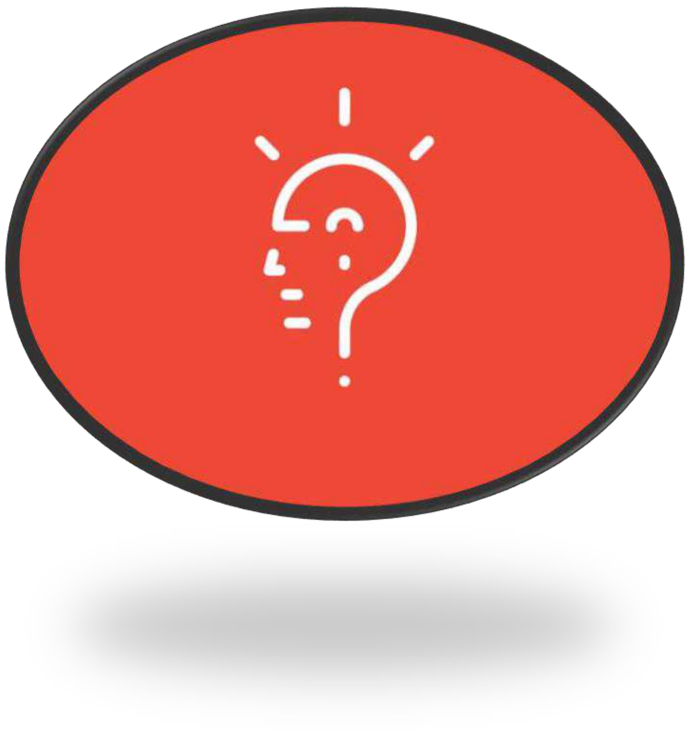


# Open Topic:
# Deep Learning and Machine Learning

Deep Learning Neural Networks replaced handcrafted features with handcrafted architectures.

Prior knowledge is not obsolete: it is merely incorporated at a higher level of abstraction.

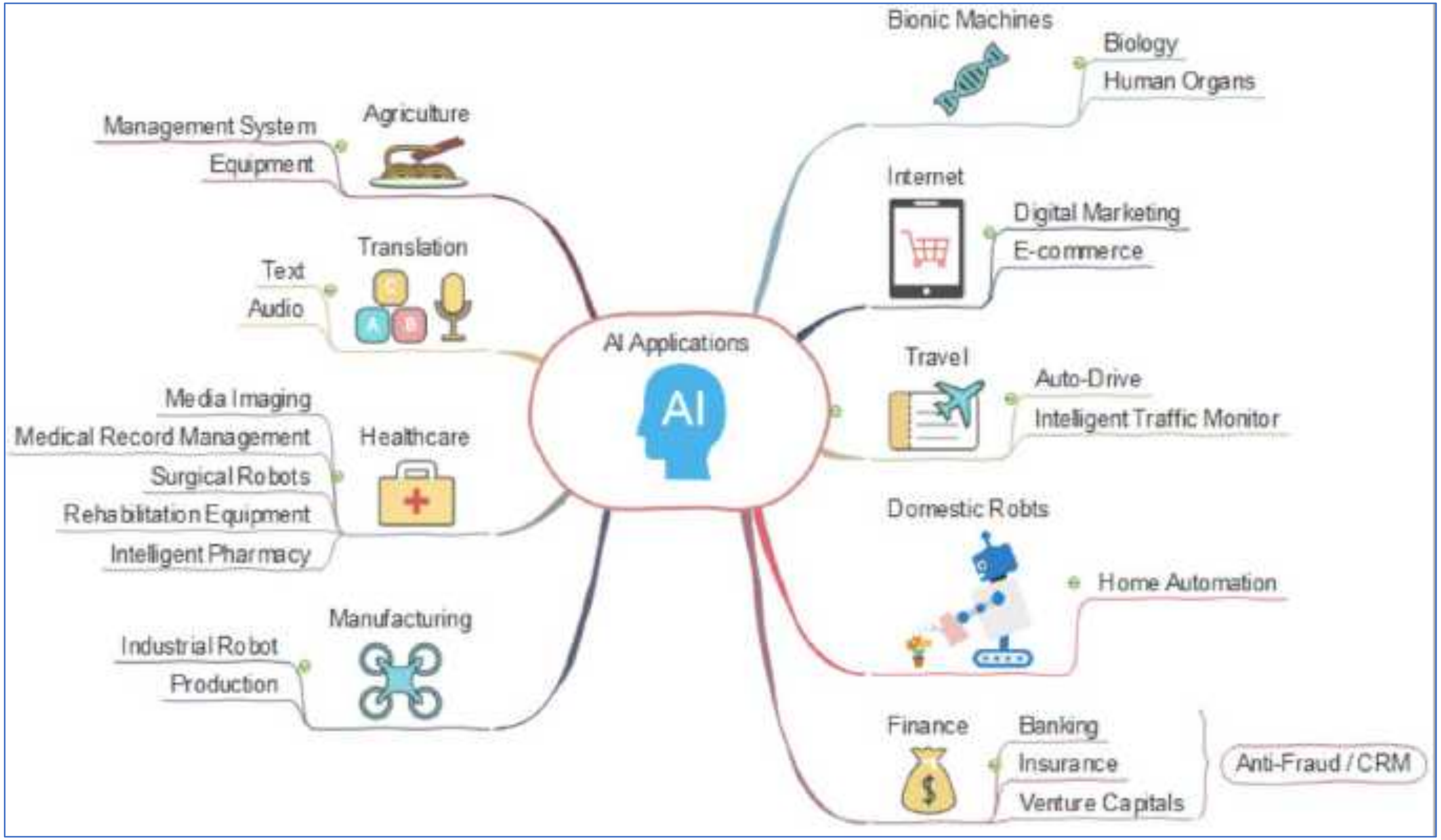
# Any Question?



# Topics

1. CS 103 Module Introduction And Class Rules
2. AI Concepts
3. AI Algorithms
4. AI Applications (AI+)

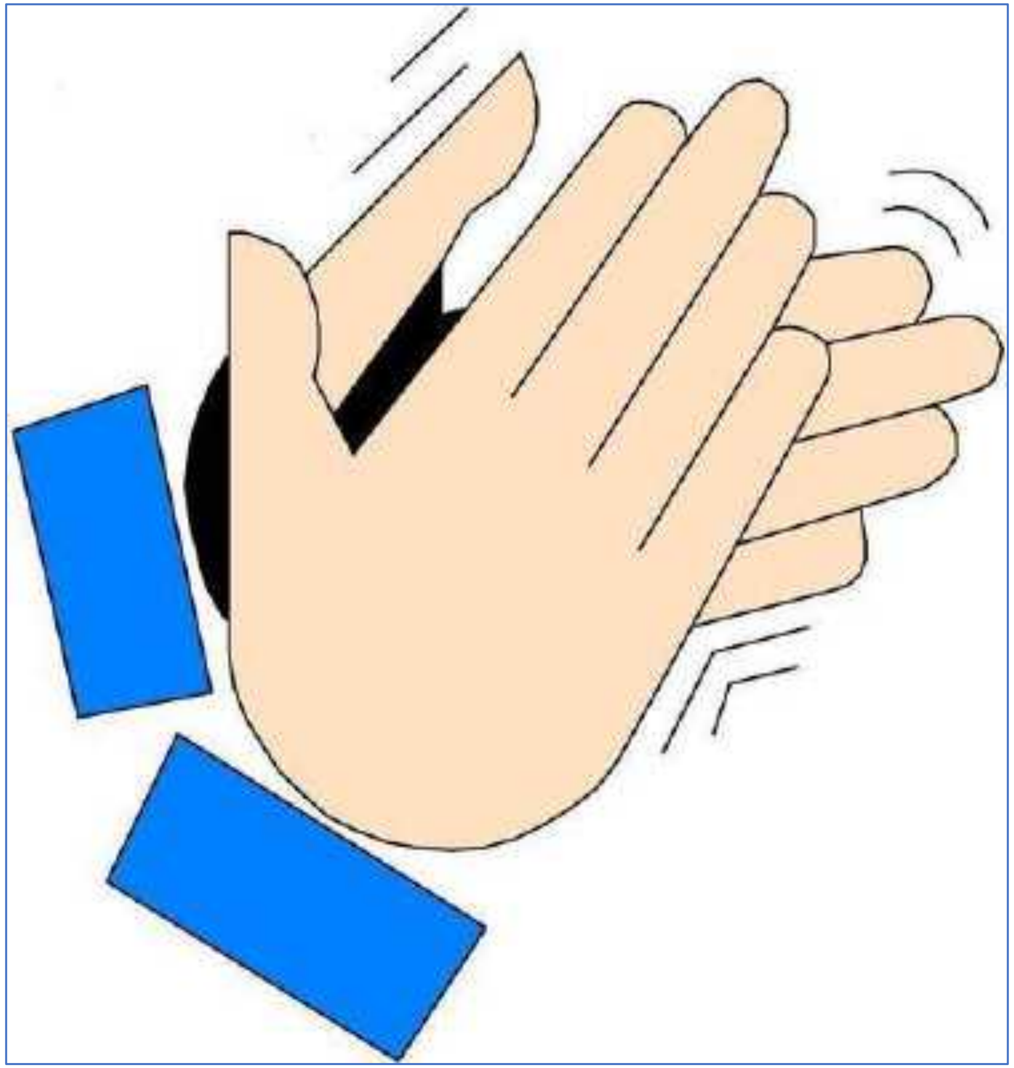
# AI Applications (AI+)



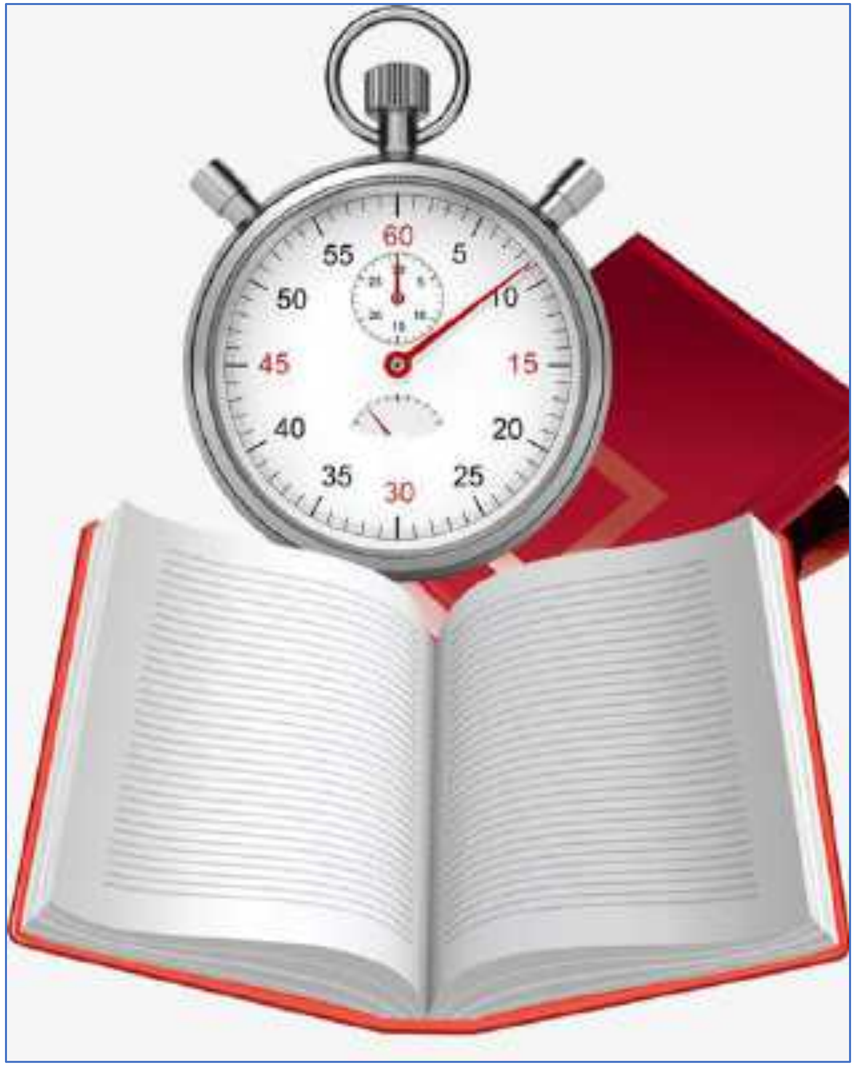
# Your “AI+” Practices

| 序号 | 题目 | 成员 |
|---|---|---|
| 1 | AI+斗地主 | 孙永康、李怀武、胡鸿飞、吴一凡、杨光、张习之、金肇轩（组长）、于佳宁 |
| 2 | AI+五子棋 | 周贤玮、韩梓辰（组长）、赵云龙、张坤龙、夏星晨 |
| 3 | High Score Gamer | 易辰朗、许天淇、黄北辰（组长）、赵思源、朱佳伟、宛清源 |
| 4 | AI application on diabetes | 周钰奇、李仪轩、董叔文、湛掌、胡钧淇（组长）、裴鸿婧 |
| 5 | AI in lung cancer | 夏瑞浩、李悦明、龚颖璇、吴云潇潇（组长）、姜欣瑜、王英豪 |
| 6 | 基于MRI图像的阿尔茨海默症分类 | 董廷臻、郑英炜（组长）、李博翱、朱嘉楠、李杨燊 |
| 7 | AI Applications in Breast Cancer Imaging | 林文心、翟靖蕾（组长）、孙瀛、林宝月、陈帅名、冀鹏宇 |
| 8 | Applications of artificial intelligence in covid-19 patients | 罗岁岁（组长）、周雅雯、肖雨馨、程旸、尹子宜 |
| 9 | 基于OCT图像的眼部多种疾病诊断和分析的调研 | 何忧、郭煜煊、朱寒旭、赵子璇（组长）、王子杰、张晓新 |
| 10 | 人工智能对白内障分级的算法综述 | 赵宇航、徐格蕾、陈星宇、祖博瀛、黄弋骞（组长） |
| 11 | 句子图片的文本情感分析 | 唐云龙、刘叶充、刘旭坤、马卓远、陈子蔚（组长）、江欣乐、陈浩然 |
| 12 | gesture recognition | 车文心、张静远、张骥霄（组长）、杜鹏辉 |
| 13 | AI in Lab | 孙含曦、于松琦、罗西（组长）、唐家豪、孙杰欣 |
| 14 | 人脸识别算法的发展与应用 | 易翔（组长）、陈俊滔、罗景南、胡泰玮、文颖潼、吴杰翰 |
| 15 | 人工智能在无障碍设施领域中的使用调查 | 马子晗（组长）、陈沐尧、林小璐、任艺伟、王增义 |
| 16 | identification of handwriting elements | 刘通、谈思序、赵伯航、张皓淇 |
| 17 | AI虚拟主播制作计划 | 王标、张倚凡（组长）、李康欣、何泽安、曾宇祺、 Zhang Kenneth |
| 18 | 人工智能技术在个性化推荐系统上的应用与研究 | 谭雅静、刘思岑、Ooi Yee Jing、孟宇阳、杨锦涛（组长） |
| 19 | 校园巴士路线优化 | 王祥辰、何鸿杰、吴子彧、樊青远（组长）、方琪涵、袁通 |
| 20 | 给线稿上色的强大AI的算法研究 | 韩晗（组长）、刘思语、赵晓蕾、陈松斌 |
| 21 | 人工智能应用于病理分析的前景与挑战 | 刘宇欣、李修治（组长）、沈睿琦 |
| 22 | 深度学习在自动驾驶中的应用 | 王晓轩 |

# Proud of CS 103 Students



# Course Review



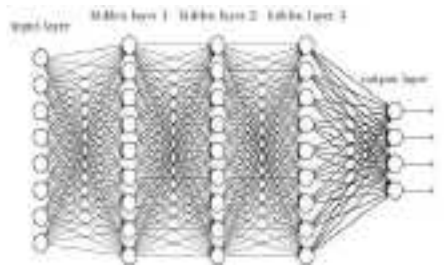
# Learn and Study

Active learning: It is about how much you think and learn

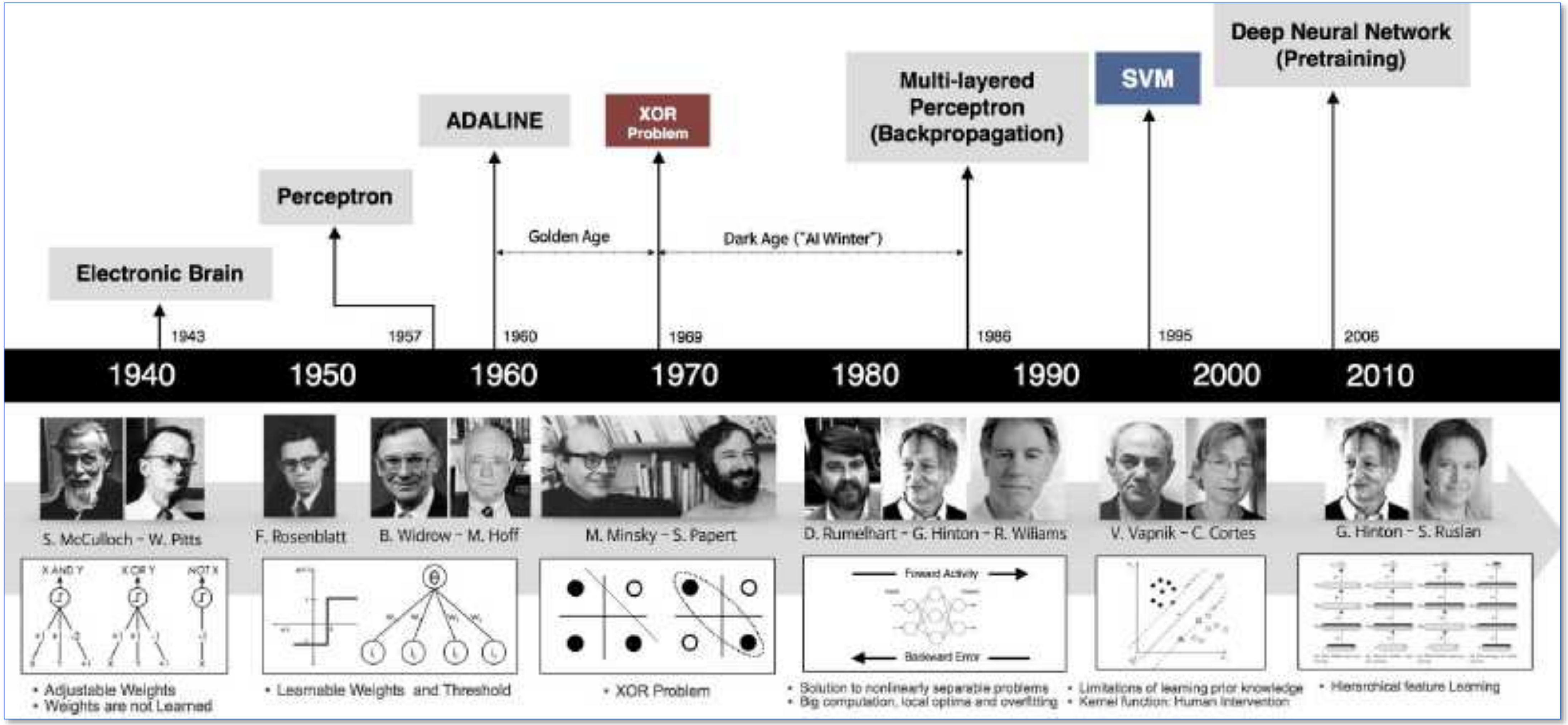
Collective study: Let us study together

# CS 103 Module Coverage

## What CS 103 will cover?

| AI Concepts | AI Algorithms | AI Application |
| --- | --- | --- |
|  |  |  |

# Computer Algorithm and AI algorithm Development Stages and Future Direction



# CS 103 -15
# Knowledge and Deep Learning

Jimmy Liu 刘江

2020-12-25