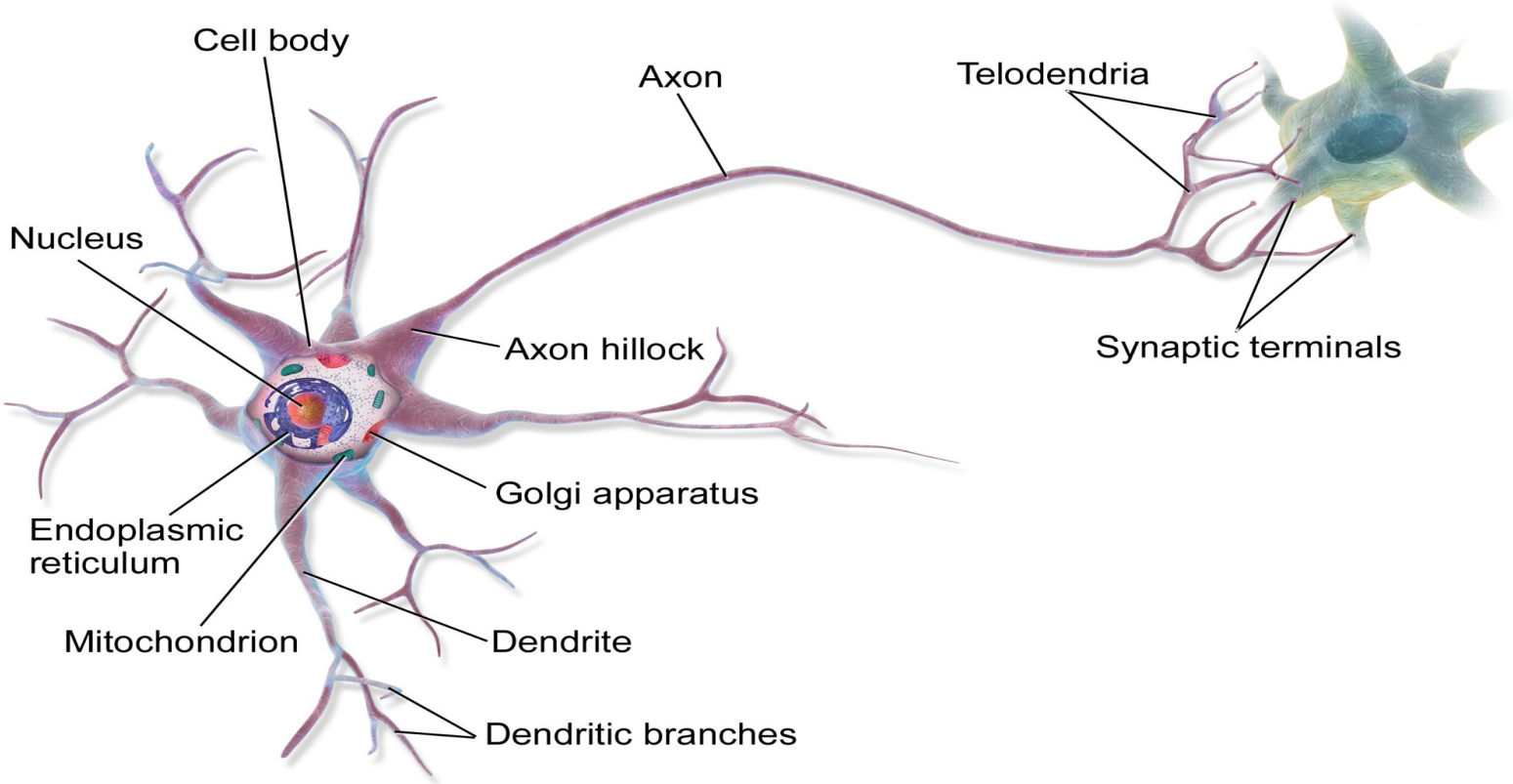
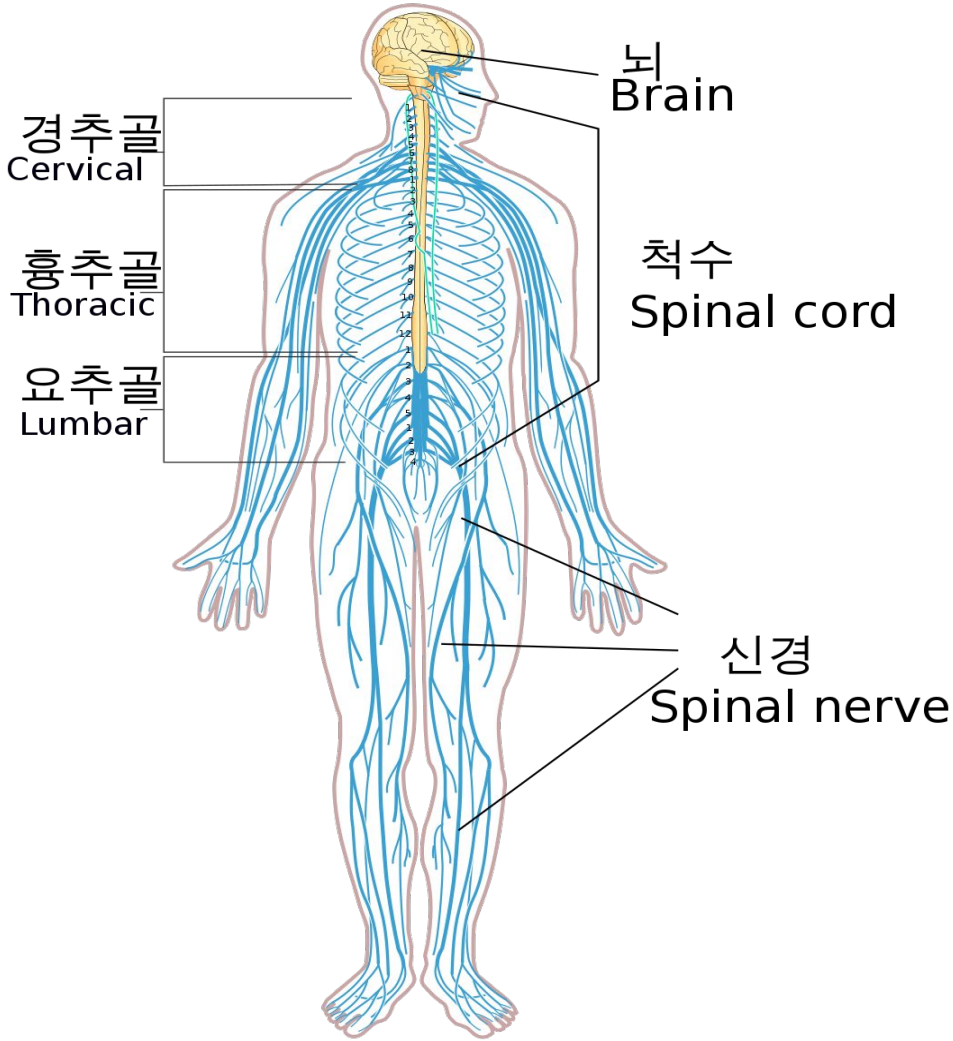
AI & DNN

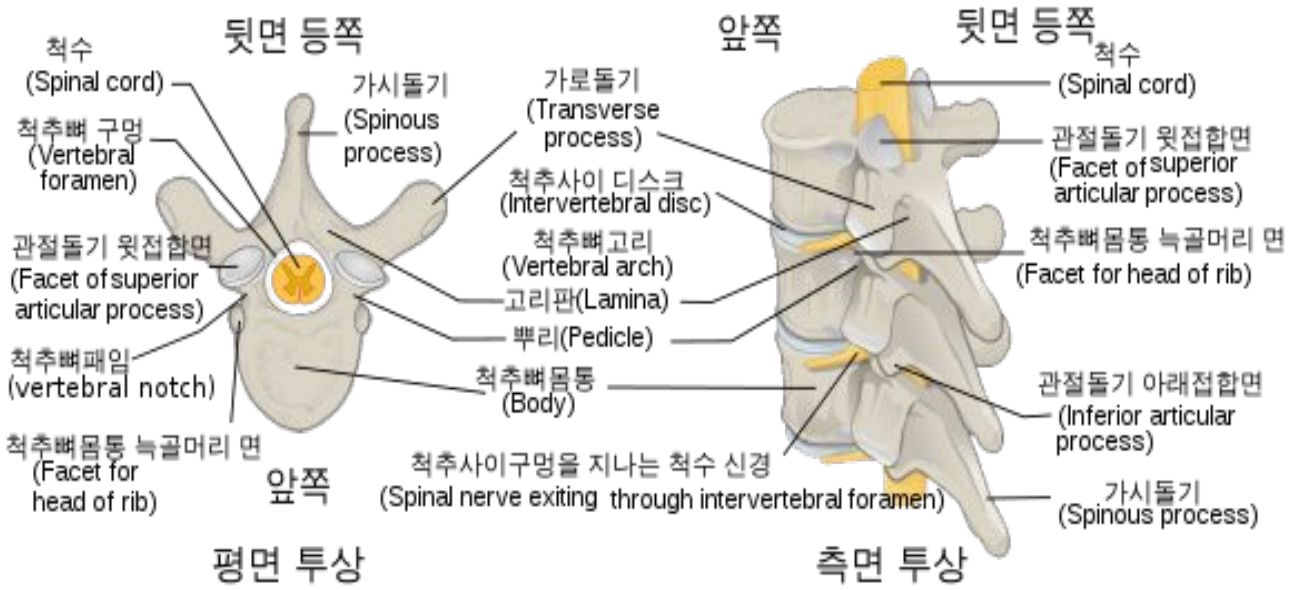
Timeline of artificial intelligence

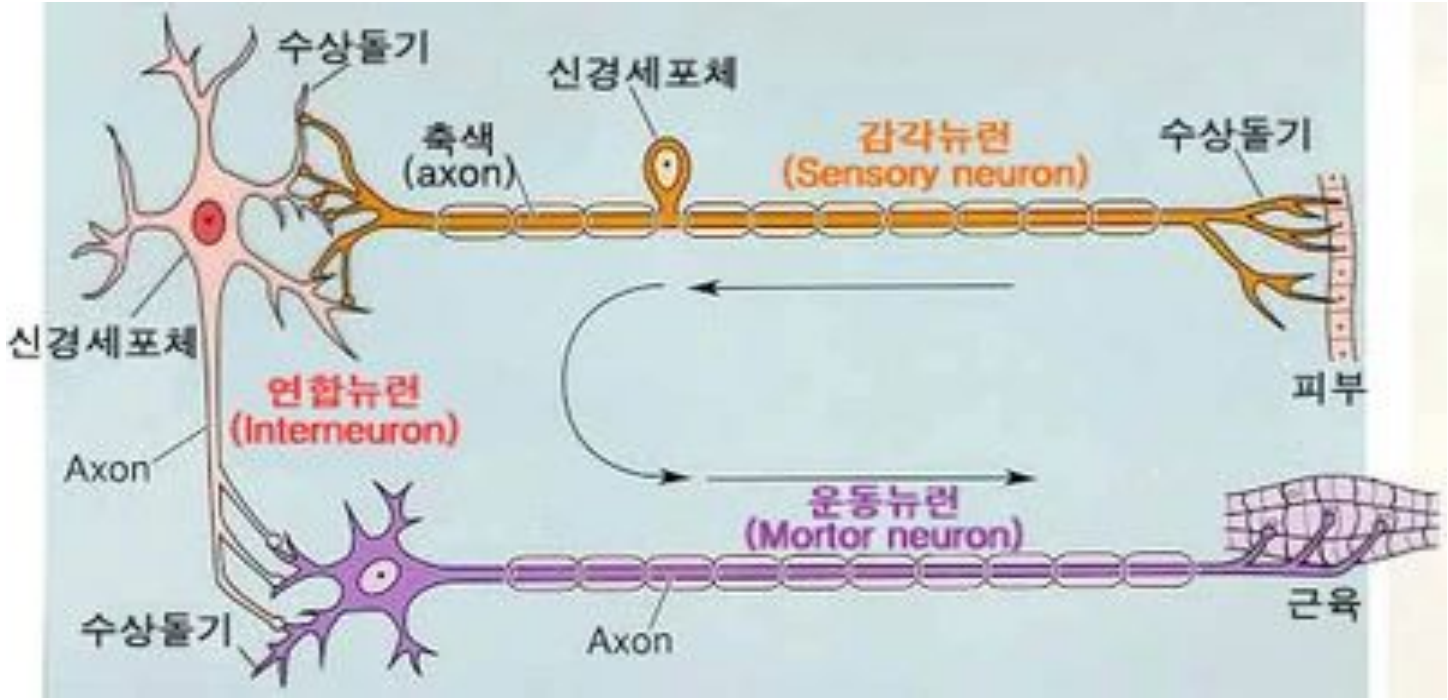
https://en.wikipedia.org/wiki/Timeline_of_artificial_intelligence

Neuron

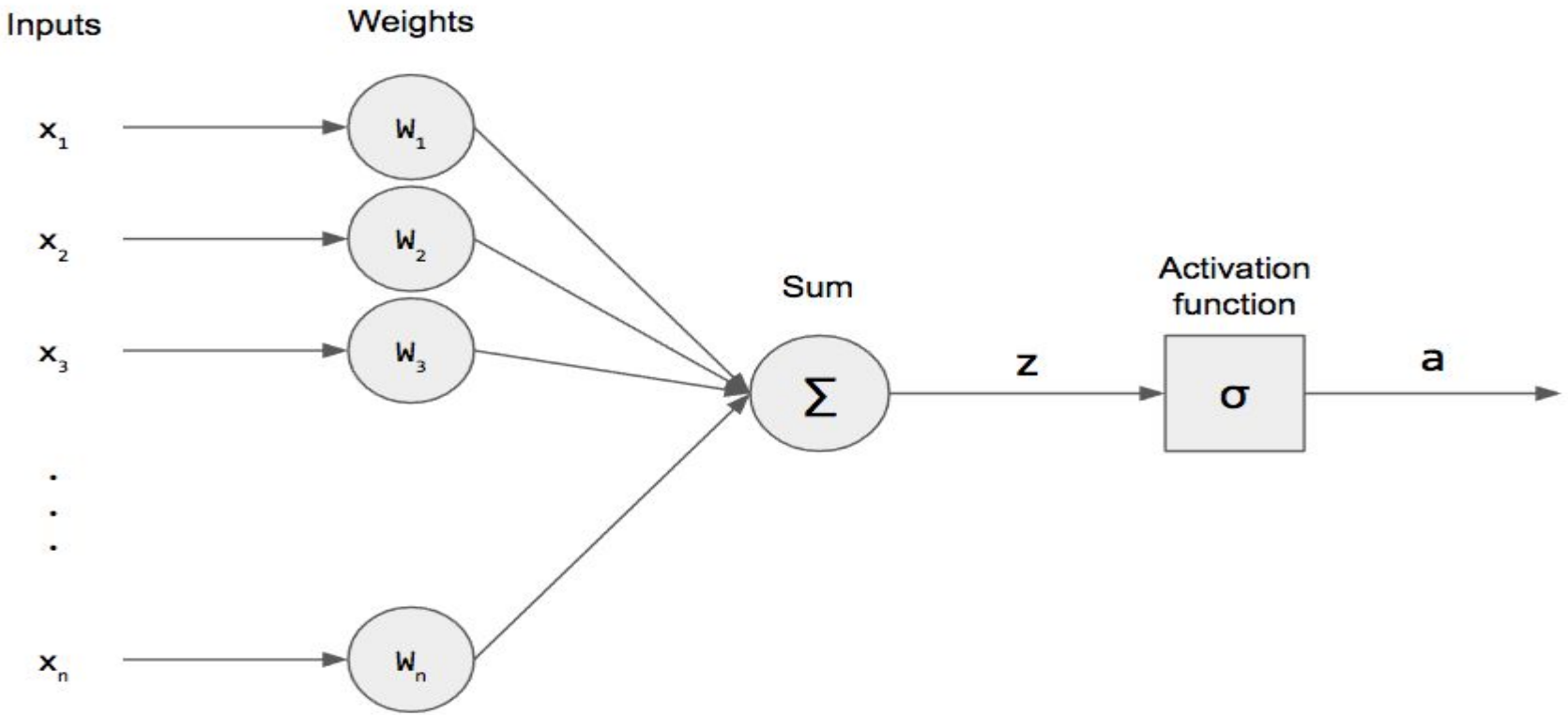

Cell body
Axon
Telodendria
Nucleus
Axon hillock
Synaptic terminals
Golgi apparatus
Endoplasmic reticulum
Mitochondrion
Dendrite
Dendritic branches


뇌
Brain
경추골
Cervical
흉추골
Thoracic
요추골
Lumbar
척수
Spinal cord
신경
Spinal nerve


뒷면 등쪽
척수
(Spinal cord)
가시돌기
(Spinous process)
척추뼈 구멍
(Vertebral foramen)
관절돌기 윗접합면
(Facet of superior articular process)
척추뼈패임
(vertebral notch)
척추뼈몸통 늑골머리 면
(Facet for head of rib)
앞쪽
평면 투상
앞쪽
가로돌기
(Transverse process)
척추사이 디스크
(Intervertebral disc)
척추뼈고리
(Vertebral arch)
고리판(Lamina)
뿌리(Pedicle)
척추뼈몸통
(Body)
척추사이구멍을 지나는 척수 신경
(Spinal nerve exiting through intervertebral foramen)
뒷면 등쪽
척수
(Spinal cord)
관절돌기 윗접합면
(Facet of superior articular process)
척추뼈몸통 늑골머리 면
(Facet for head of rib)
관절돌기 아래접합면
(Inferior articular process)
가시돌기
(Spinous process)
측면 투상

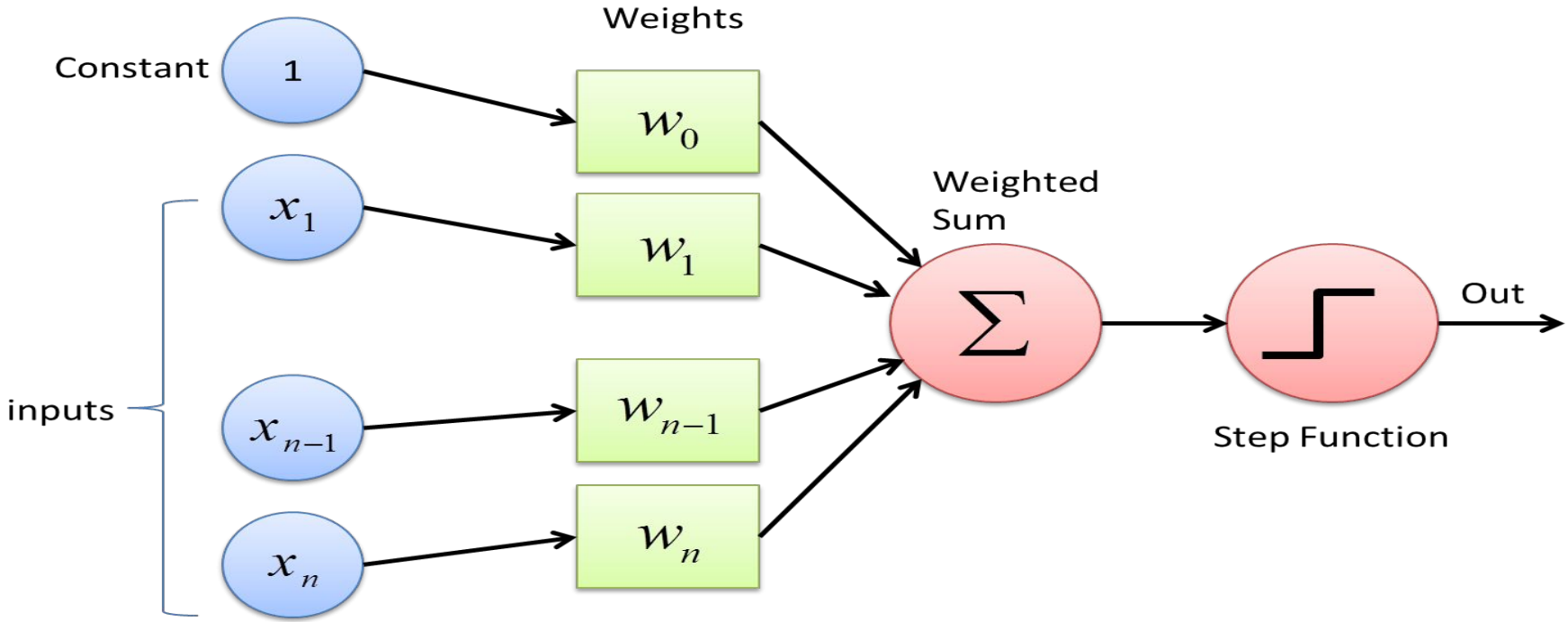

수상돌기
신경세포체
축색
(axon)
감각뉴런
(Sensory neuron)
수상돌기
신경세포체
피부
연합뉴런
(Interneuron)
Axon
운동뉴런
(Mortor neuron)
수상돌기
Axon
근육

Neural Representation

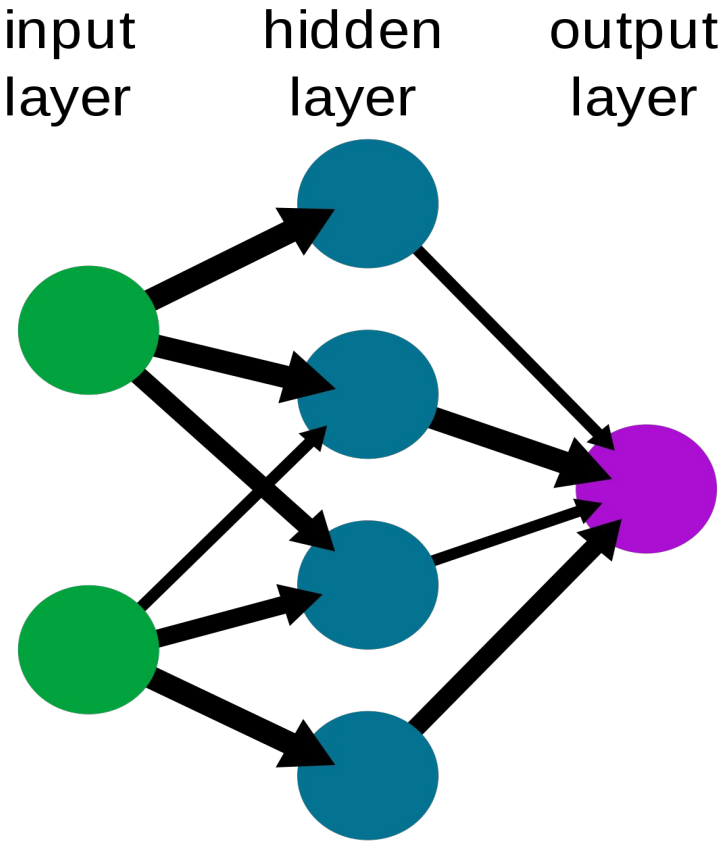


Perceptron

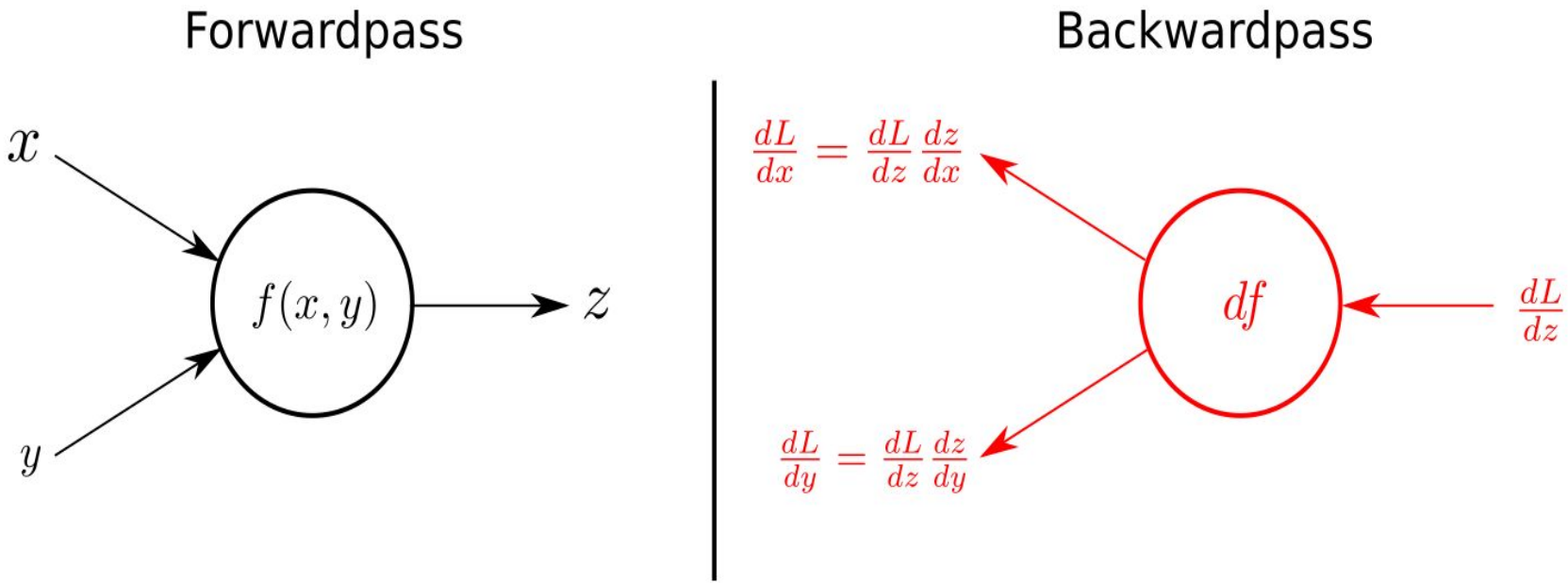
a neural network unit (an artificial neuron) that does certain computations to detect features or business intelligence in the input data.

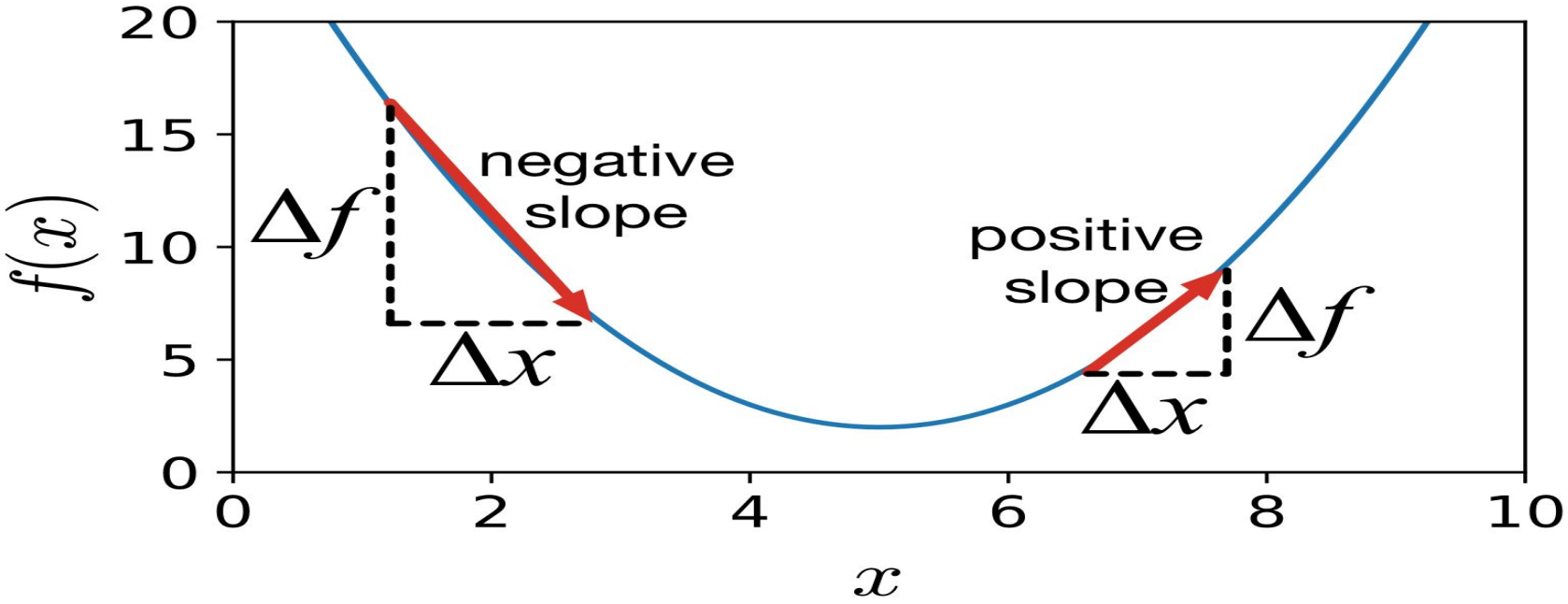


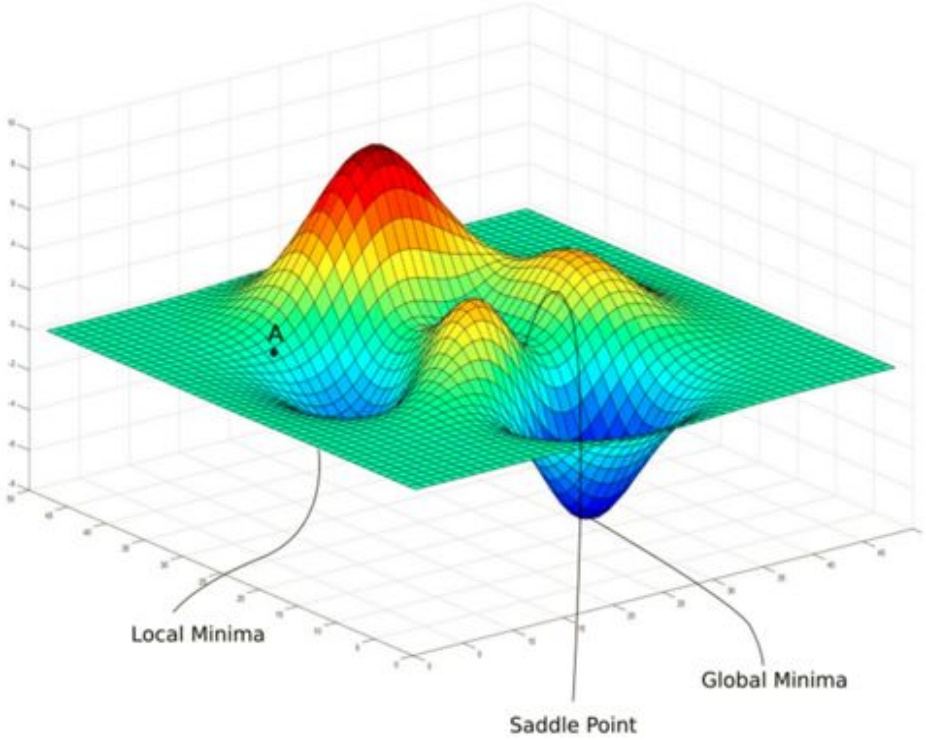
A simple neural network

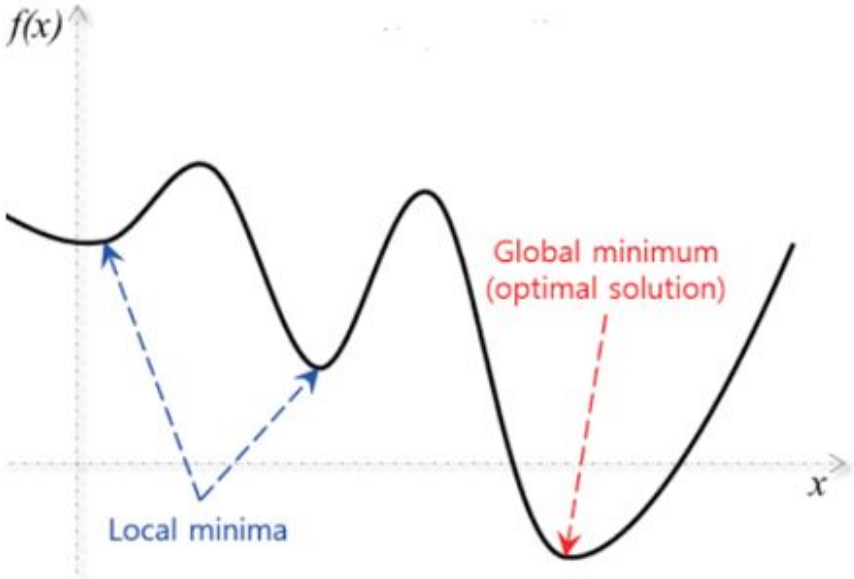


Propagation & Backpropagation

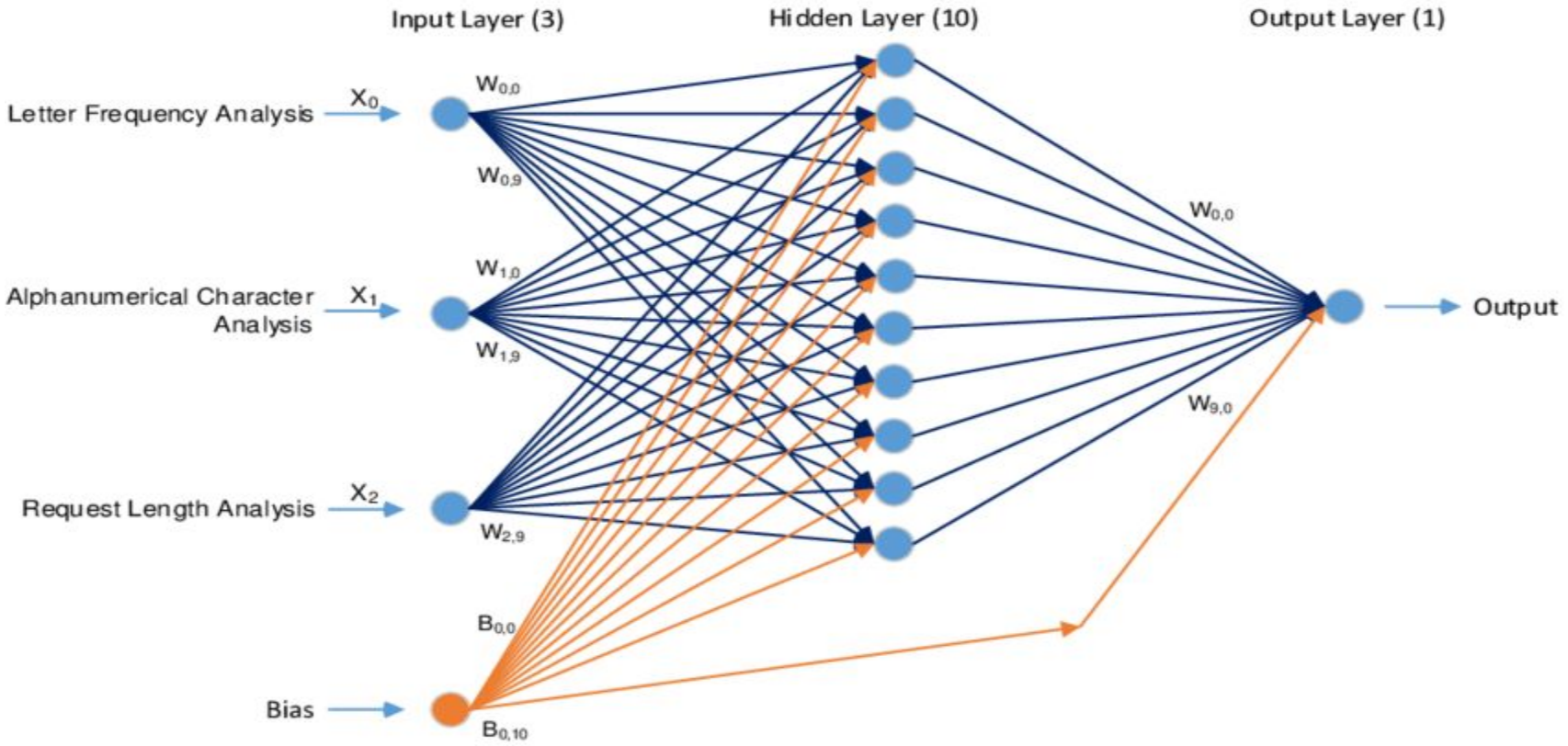



20
15
10
5
0
0
2
4
6
8
10
$f(x)$
x
Δf
Δx
negative
slope
positive
slope
Δf
Δx

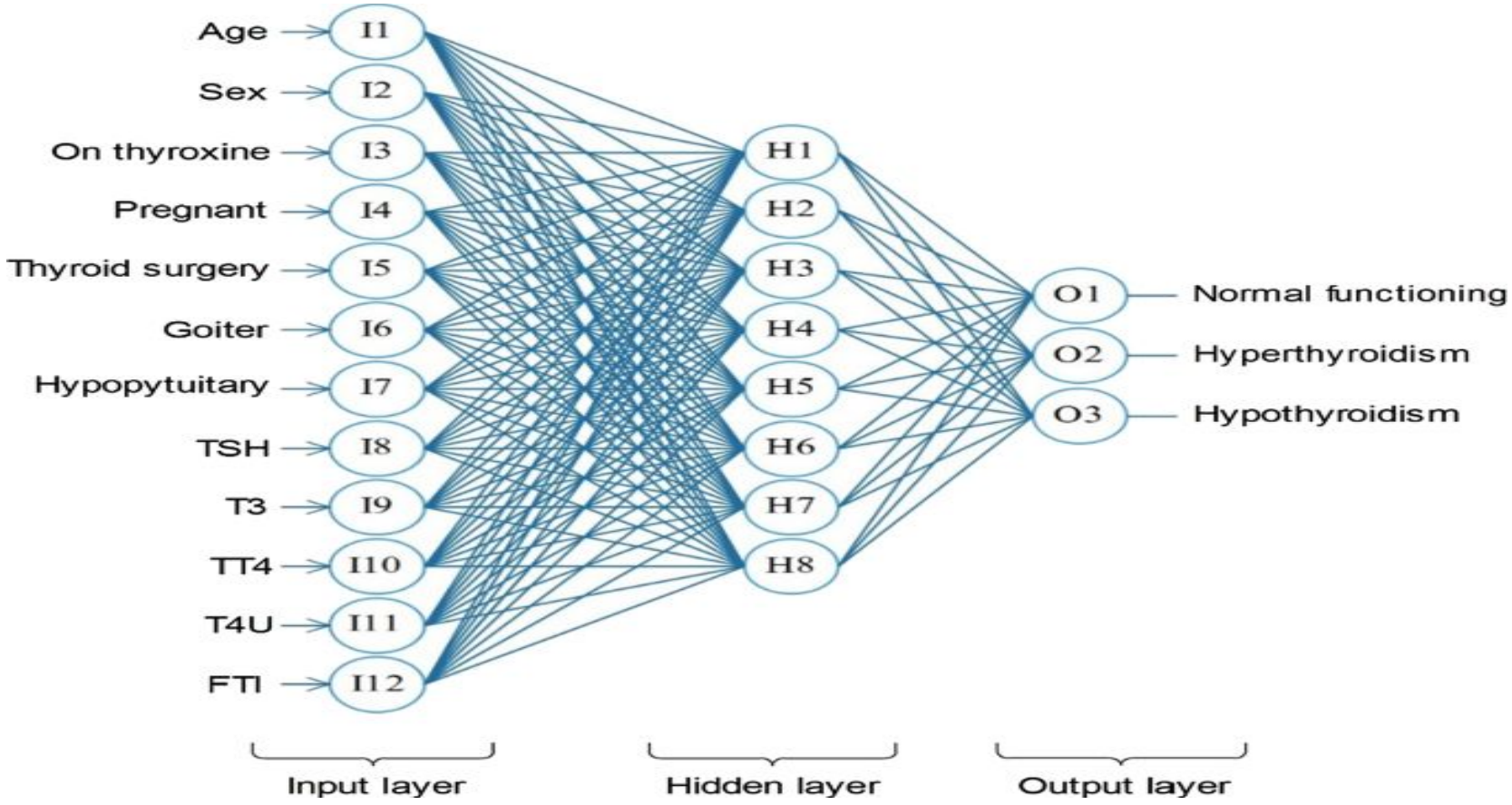

A
Local Minima
Saddle Point
Global Minima


f(x)
Global minimum
(optimal solution)
Local minima
x

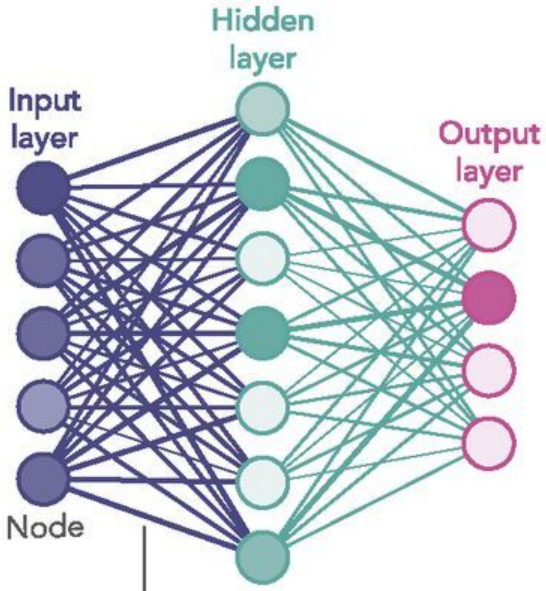
Computational neural networks



A multilayer perceptron neural network for thyroid disease diagnosis

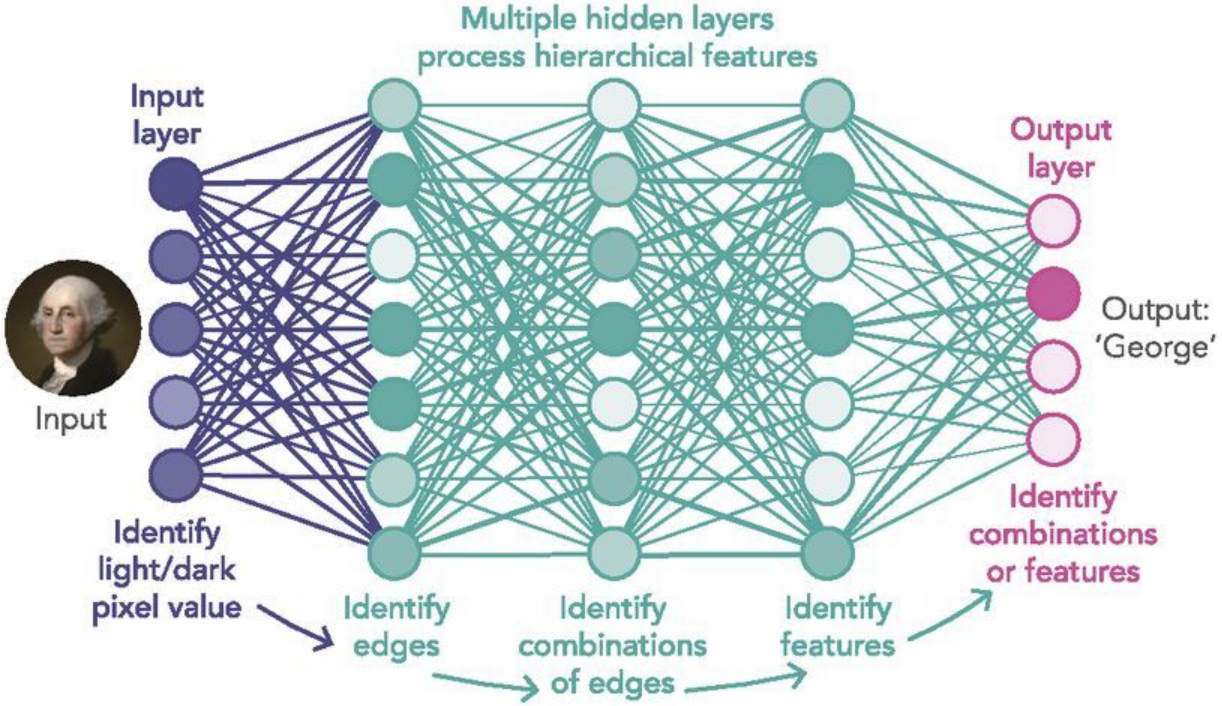


1980S-ERA NEURAL NETWORK



Links carry signals from one node to another, boosting or damping them according to each link's 'weight'.

DEEP LEARNING NEURAL NETWORK



Edge detection

1	2	3
4	5	6
7	8	9

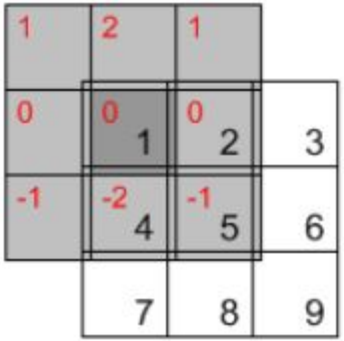
Input

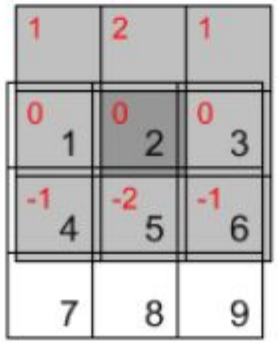
n \ m	-1	0	1
-1	-1	-2	-1
0	0	0	0
1	1	2	1

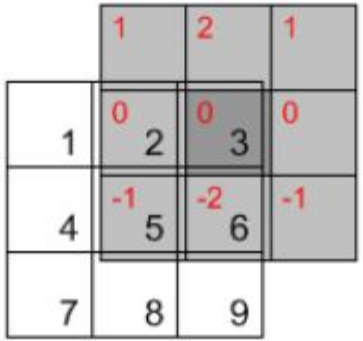
Kernel

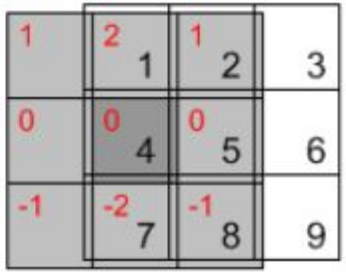
-13	-20	-17
-18	-24	-18
13	20	17

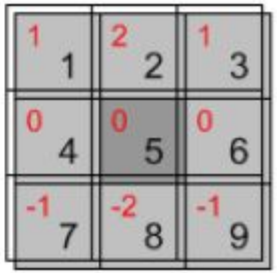
Output



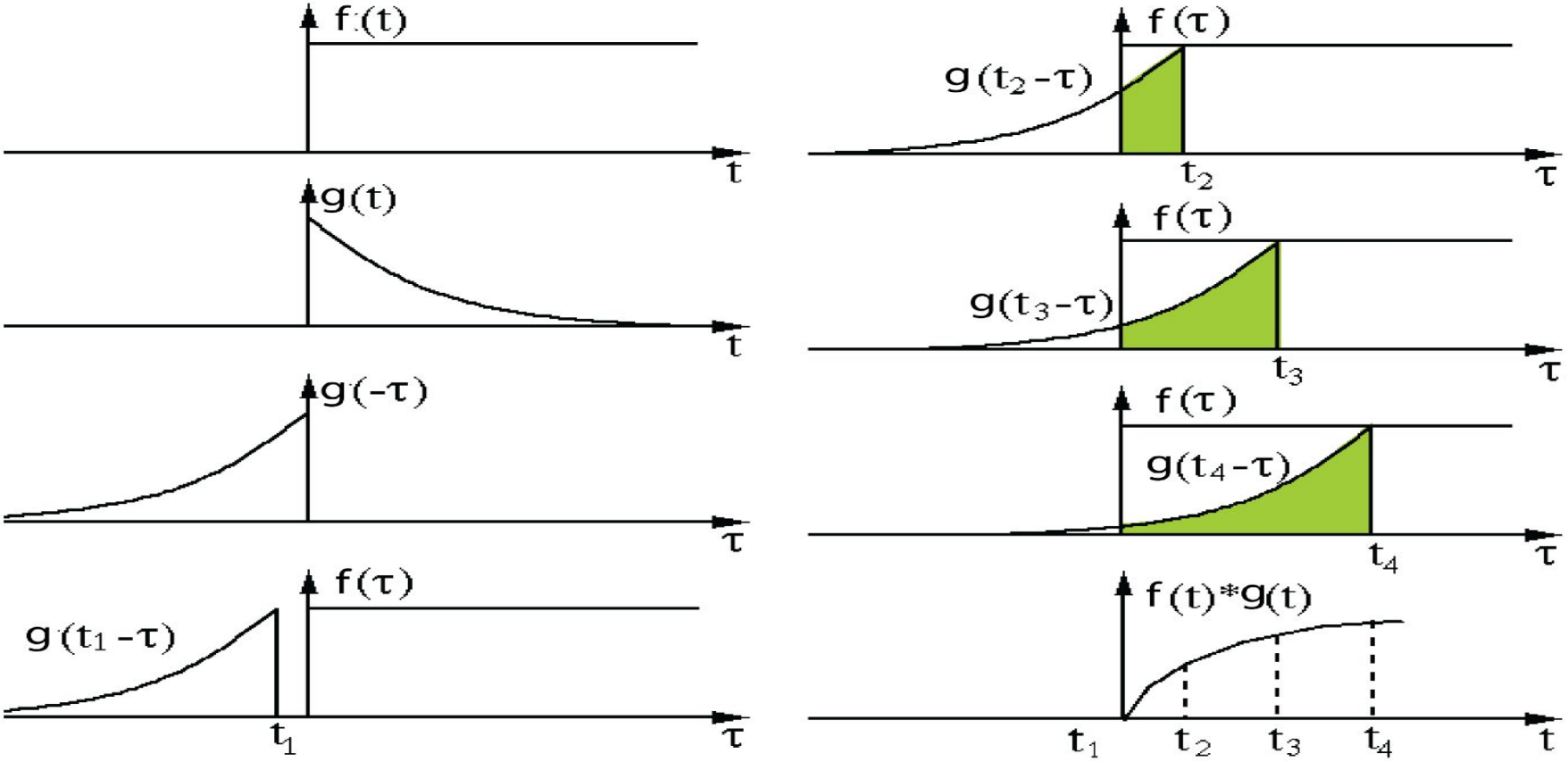






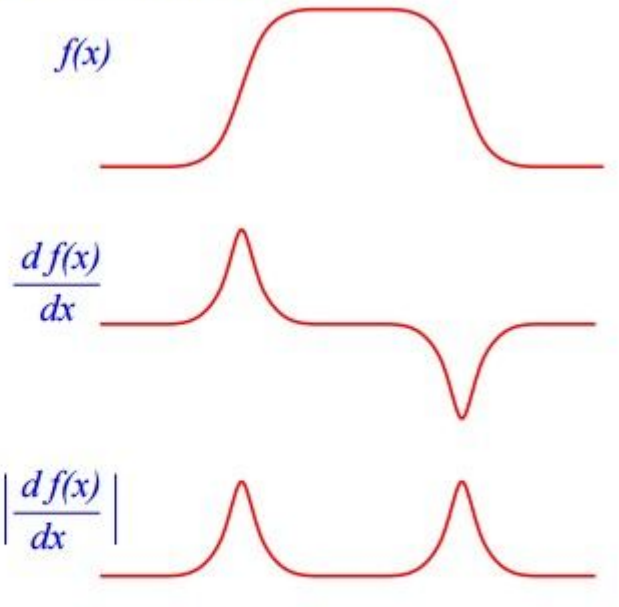


Convolution



Edge detection

First Order Differentials: In One-Dimension we have

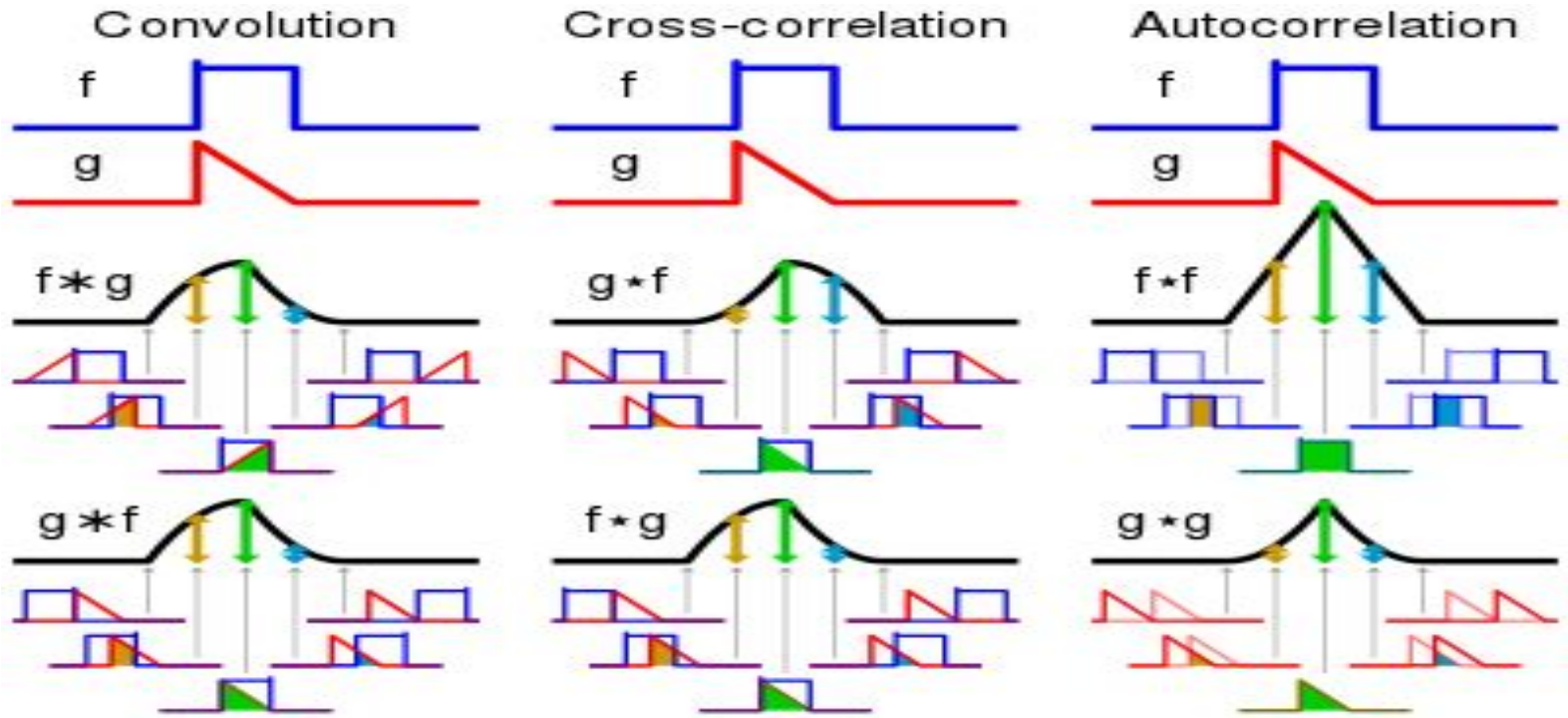


We can then detect the edge by a simple threshold of

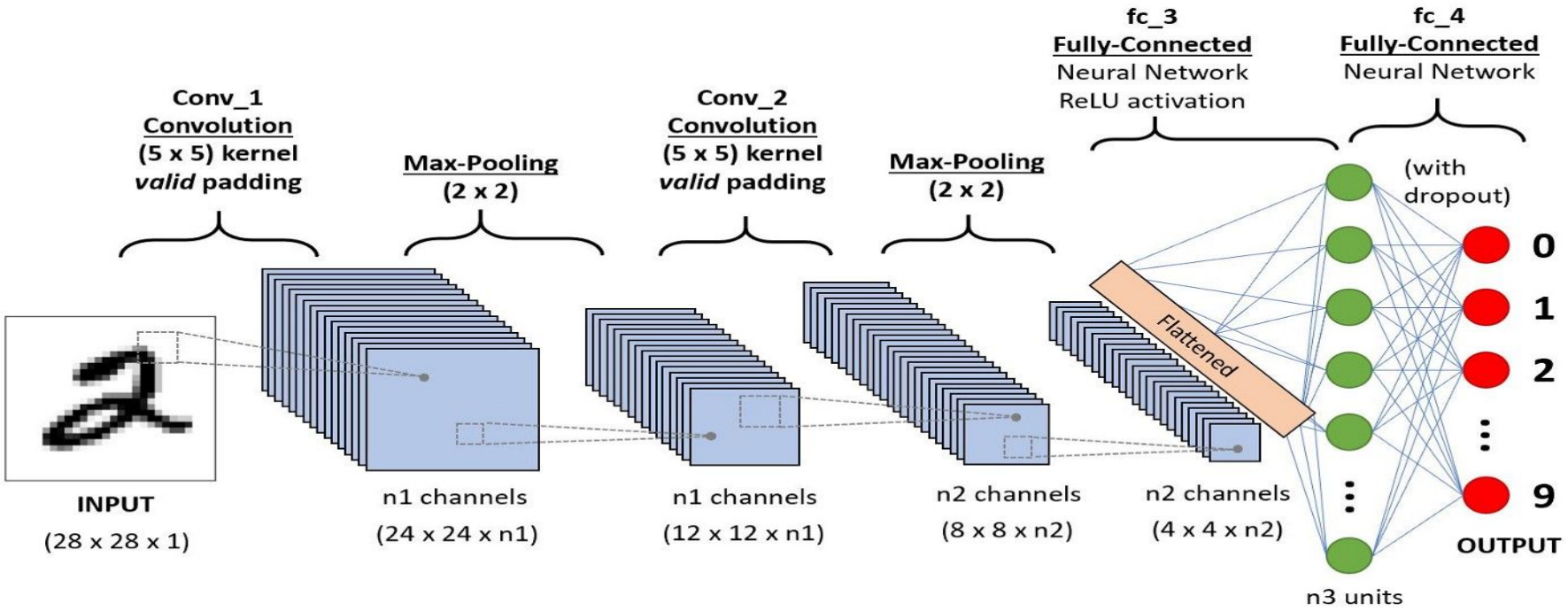
$$\left|\frac{\mathrm{d}f(x)}{\mathrm{d}x}\right| > T \quad \Rightarrow \text{Edge}$$

Edge Filter

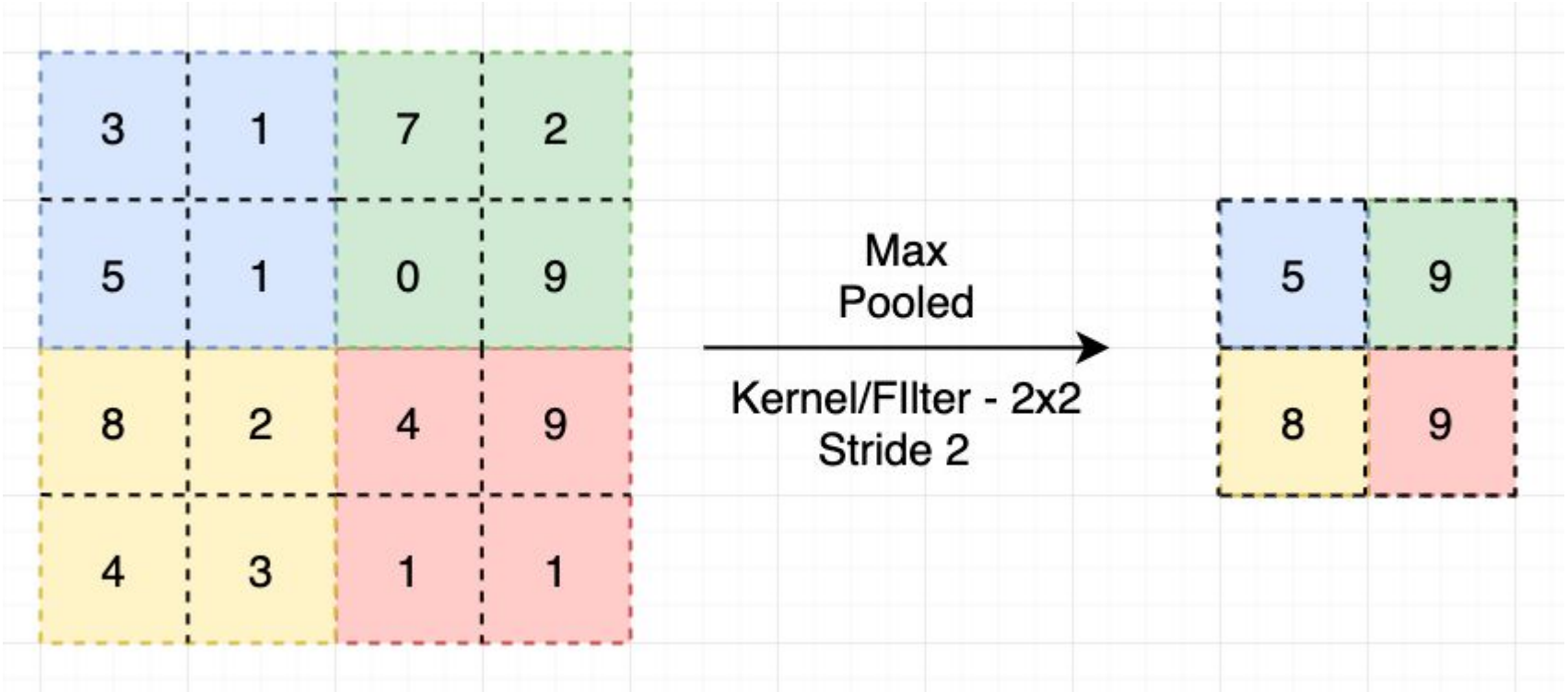
CNN…


Convolution
Cross-correlation
Autocorrelation
f
g
f∗g
g⋆f
f⋆f
g∗f
f⋆g
g⋆g

CNN


Conv_1
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
Conv_2
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
fc_3
Fully-Connected
Neural Network
ReLU activation
fc_4
Fully-Connected
Neural Network
(with dropout)
Flattened
INPUT
(28 x 28 x 1)
n1 channels
(24 x 24 x n1)
n1 channels
(12 x 12 x n1)
n2 channels
(8 x 8 x n2)
n2 channels
(4 x 4 x n2)
n3 units
0
1
2
9
OUTPUT

Max Pooling, Filter & Stride



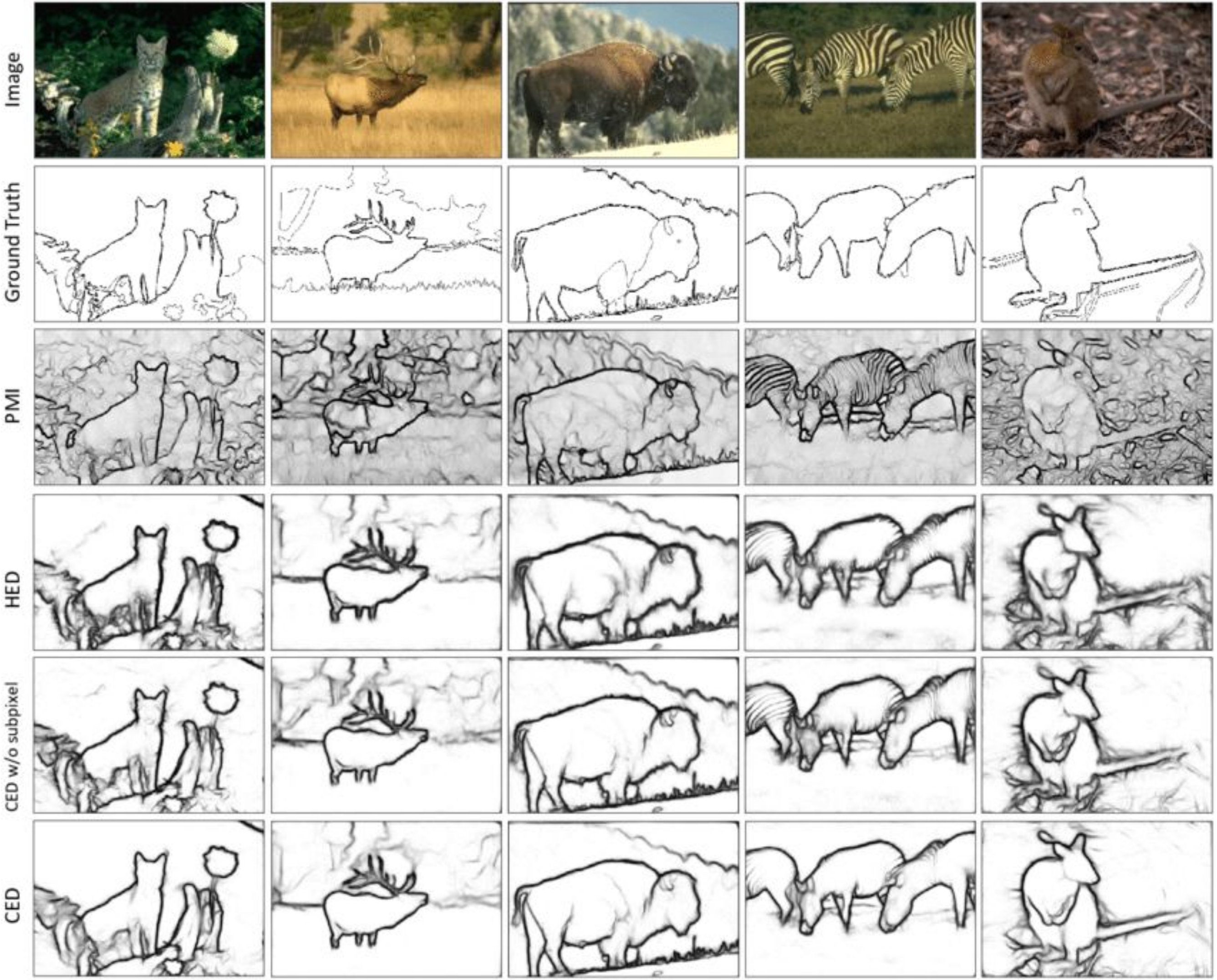
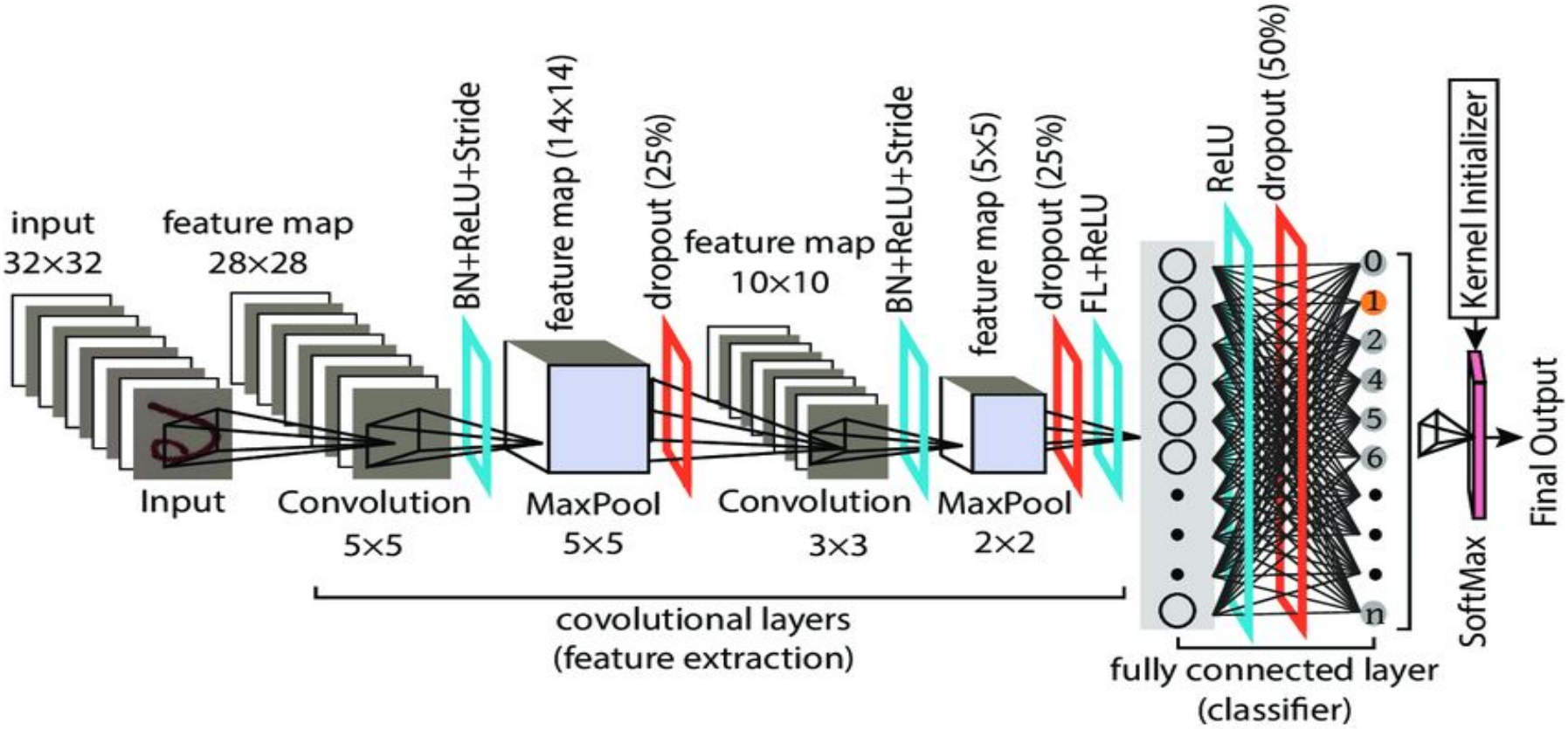
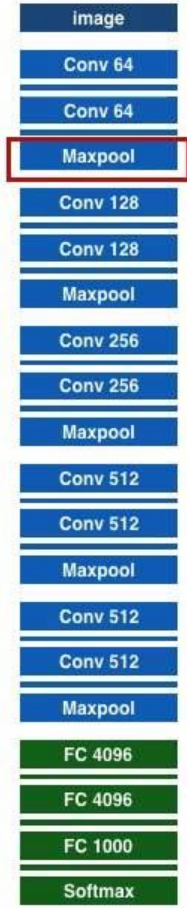


Image
Ground Truth
PMI
HED
CED w/o subpixel
CED

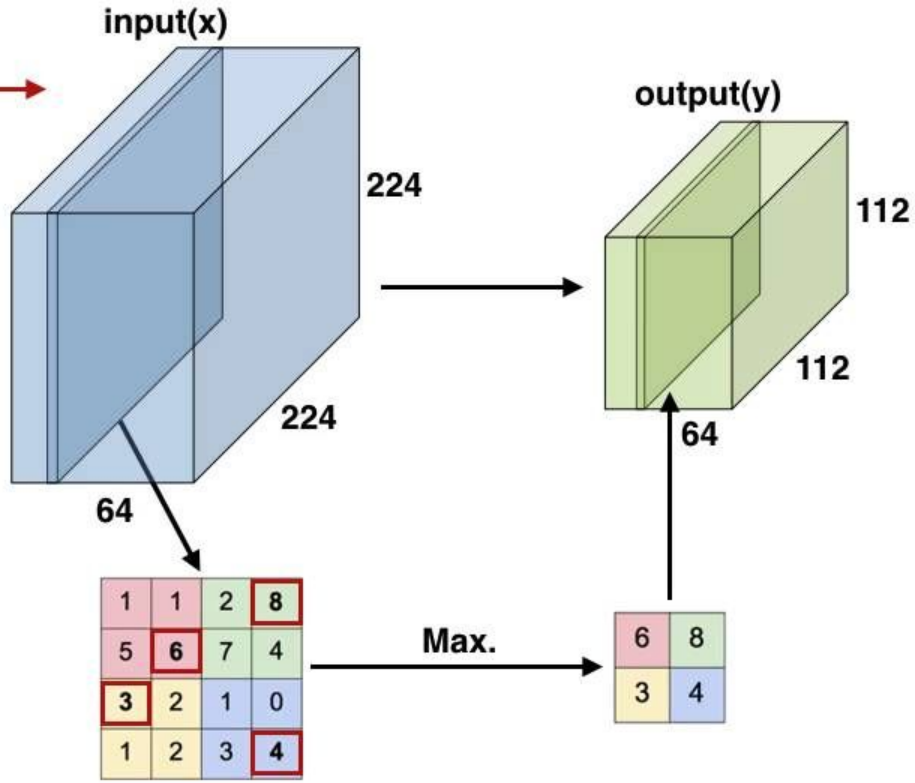
CNN


input
32×32
feature map
28×28
BN+ReLU+Stride
feature map (14×14)
dropout (25%)
feature map
10×10
BN+ReLU+Stride
feature map (5×5)
dropout (25%)
FL+ReLU
ReLU
dropout (50%)
Kernel Initializer
Input
Convolution
5×5
MaxPool
5×5
Convolution
3×3
MaxPool
2×2
0
1
2
4
5
6
n
SoftMax
Final Output
covolutional layers
(feature extraction)
fully connected layer
(classifier)

CNN

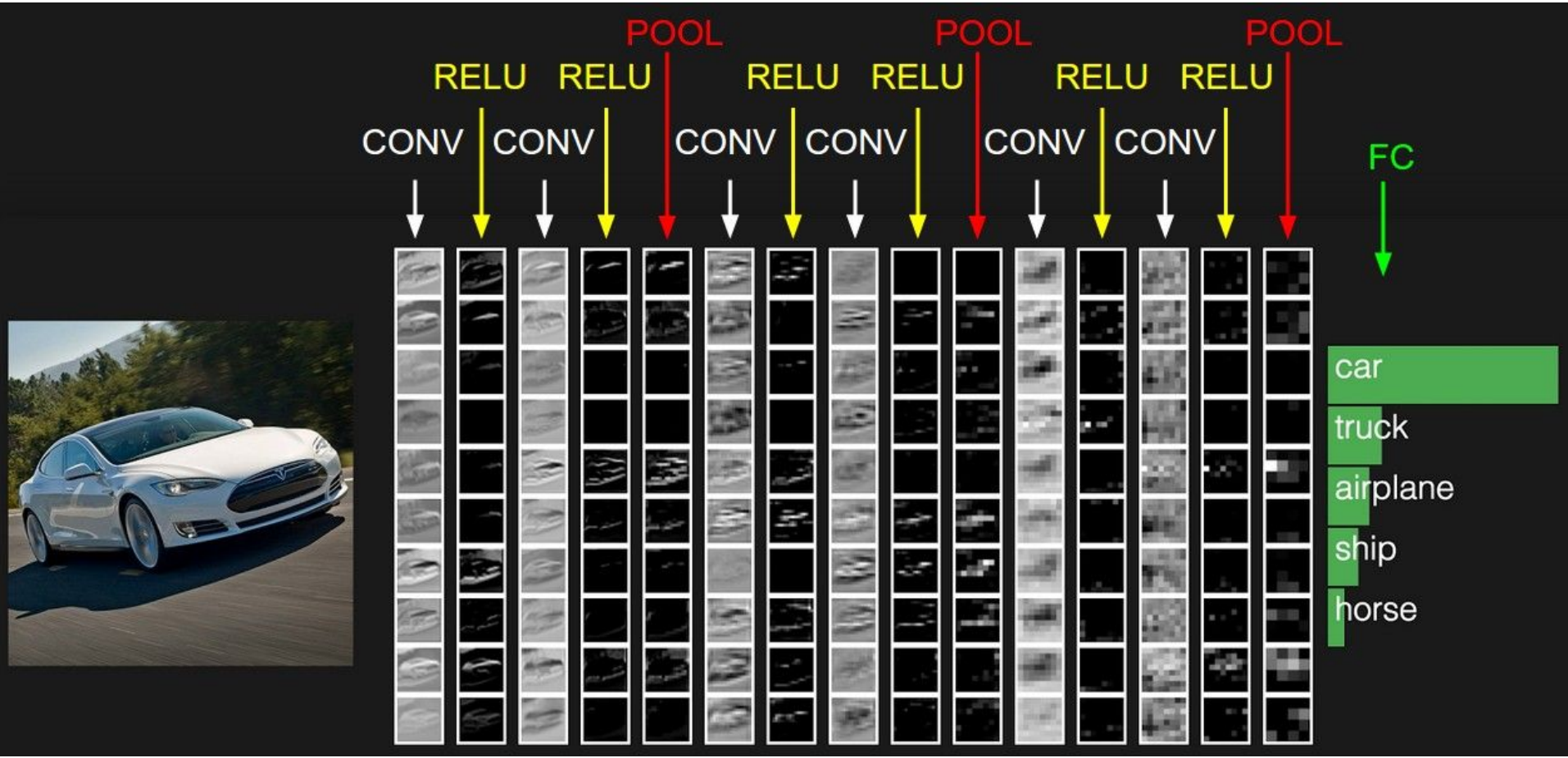


max pooling

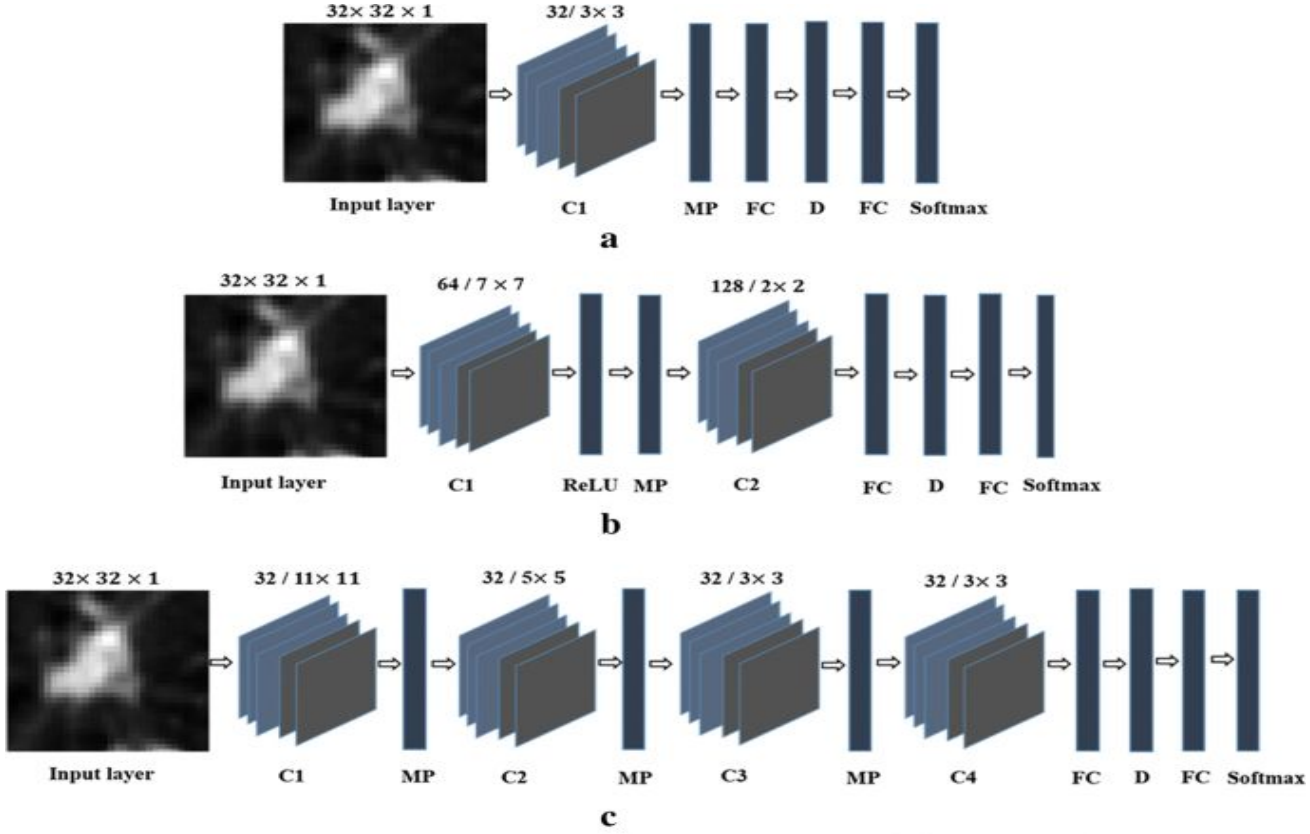


slides courtesy of Jonghoon Jin

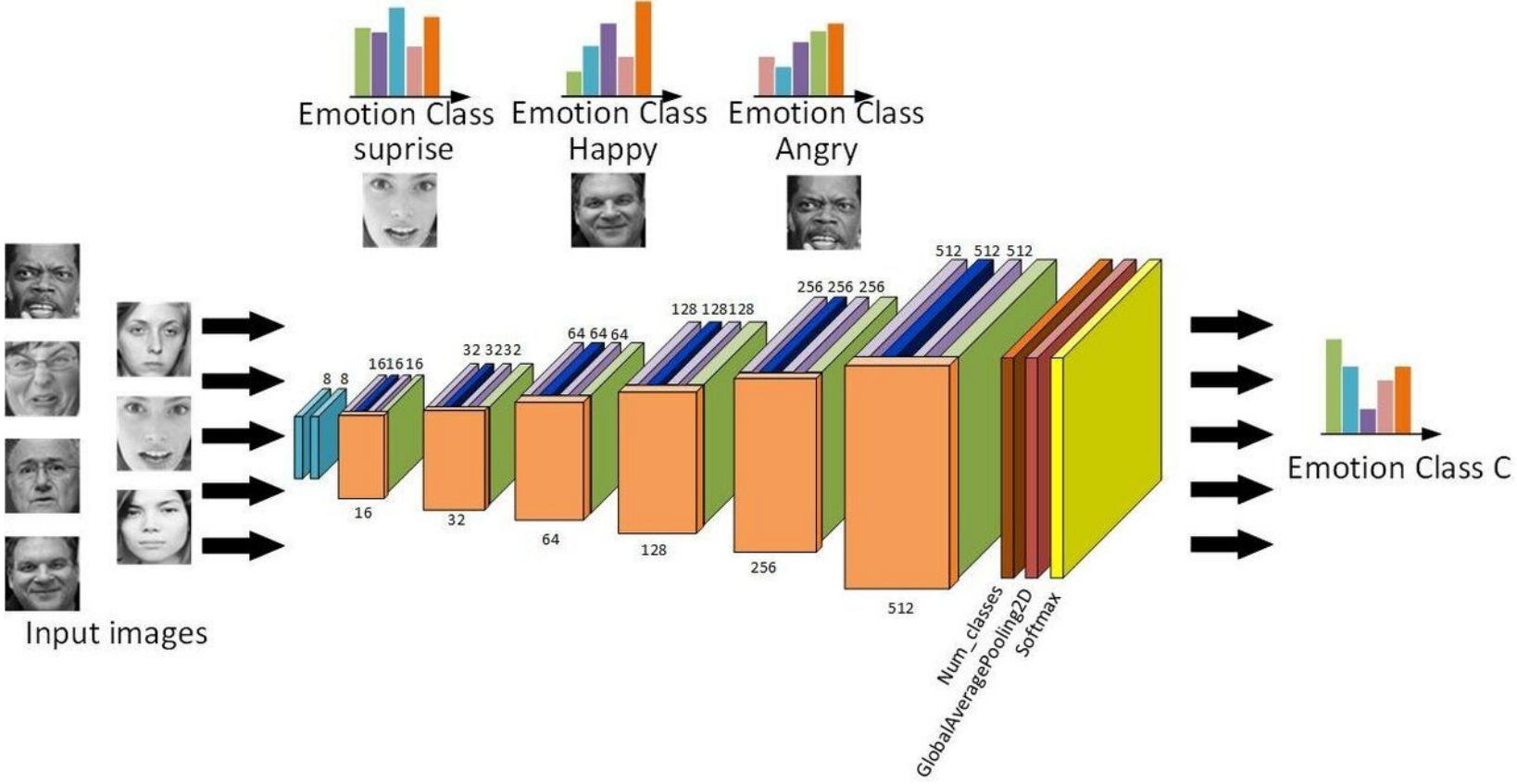
Eugenio Culurciello
© 2016


POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

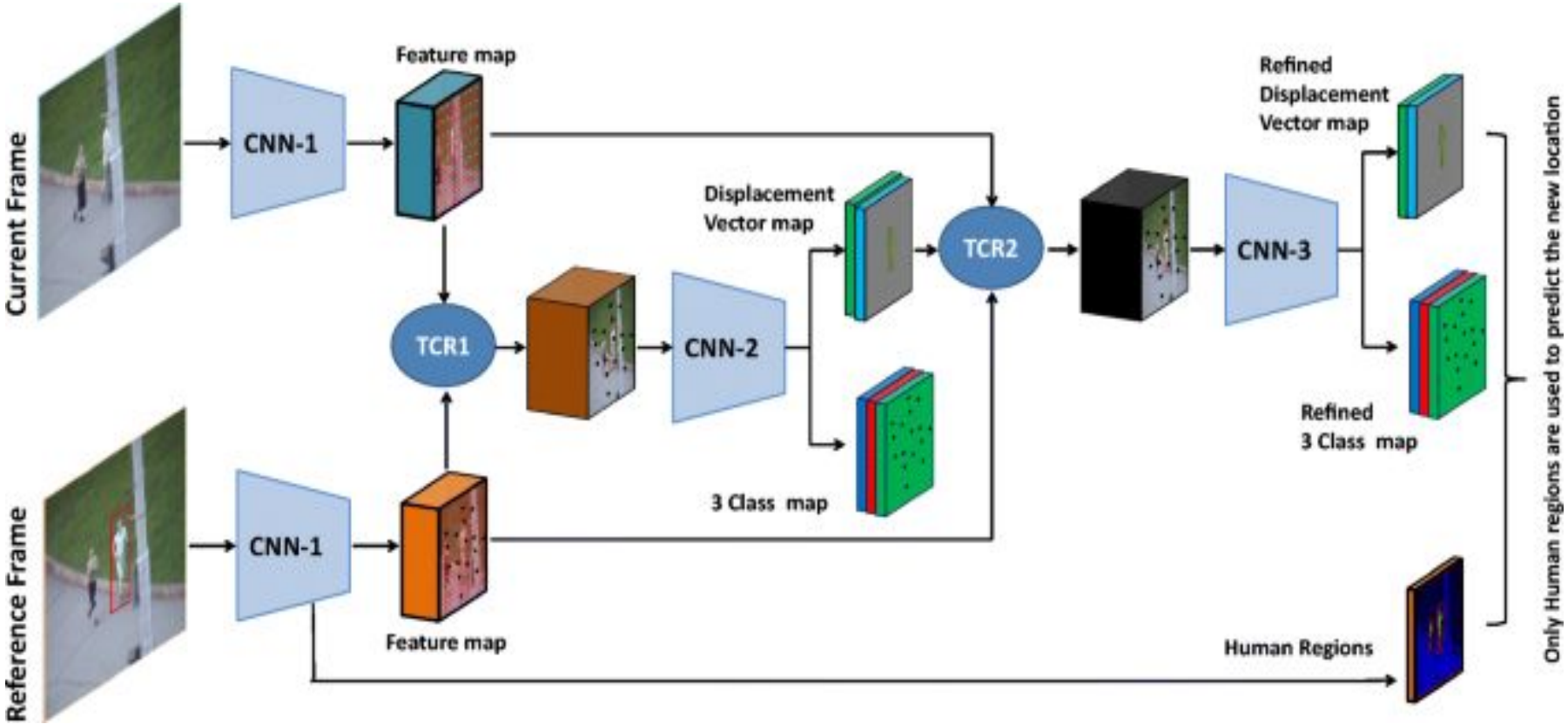
automatic detection of pulmonary nodules from CT lung screening (lung cancer)



C: convolution, MP: MaxPooling, D: dropout, FC: fully connected.


Emotion Class
suprise
Emotion Class
Happy
Emotion Class
Angry
Input images
8 8
16 16 16
32 32 32
64 64 64
128 128 128
256 256 256
512 512 512
16
32
64
128
256
512
Num_classes
GlobalAveragePooling2D
Softmax
Emotion Class C
Conv2D+Batch Normalisation+ReLU
SeparableConv2D+Batch Normalisation+ReLU
MaxPooling2D
MaxPooling2D
Conv2D
Softmax

Human tracking and localization



Vision & NLP…

CNN

RNN

LSTM

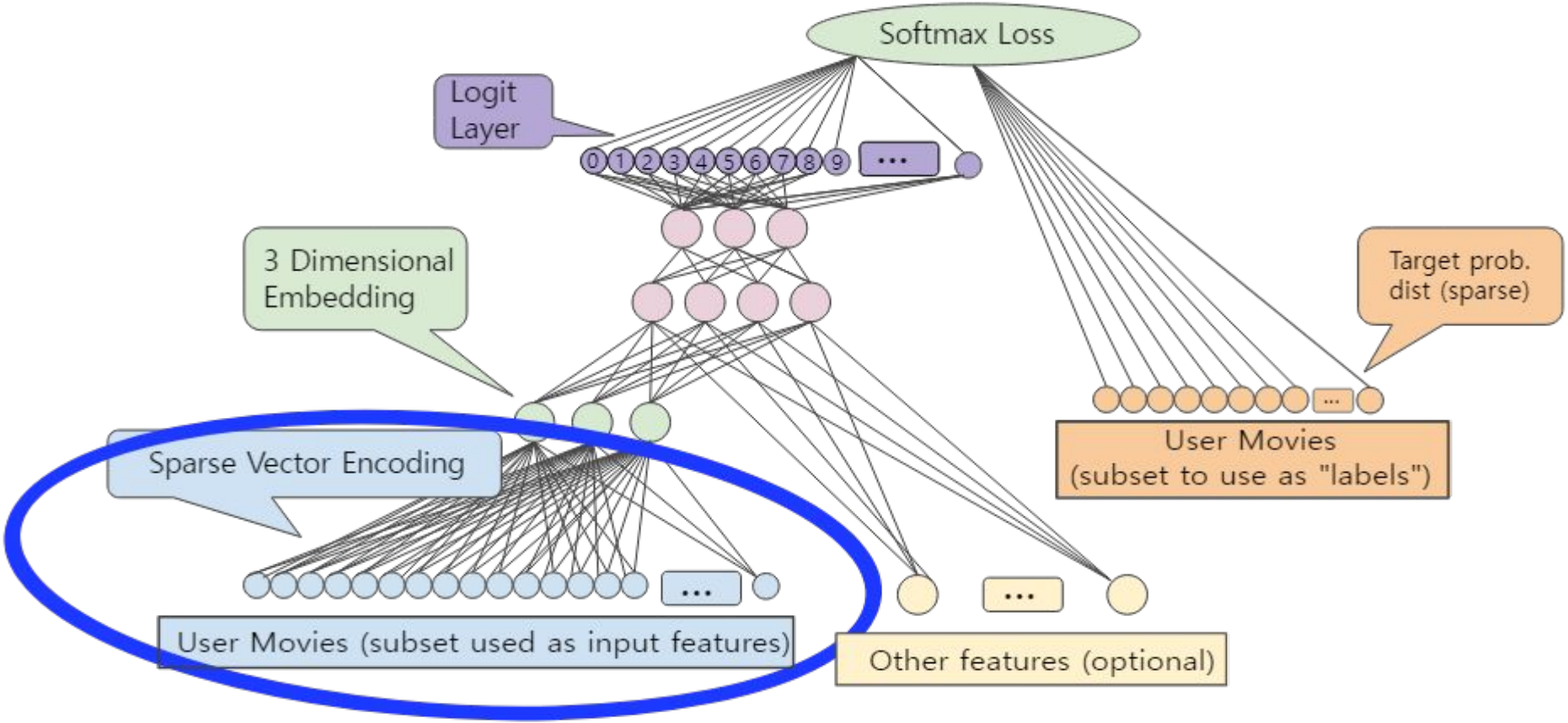
GAN

Transfer

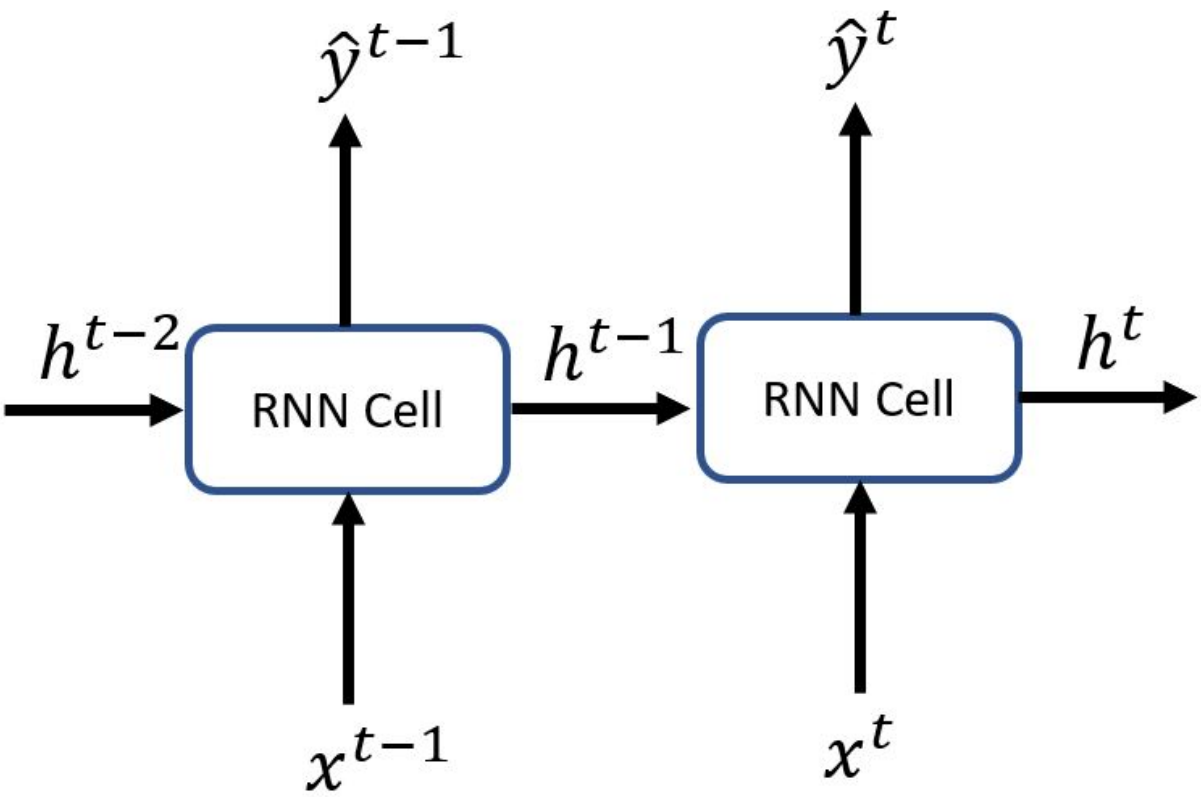
BERT

GPT n

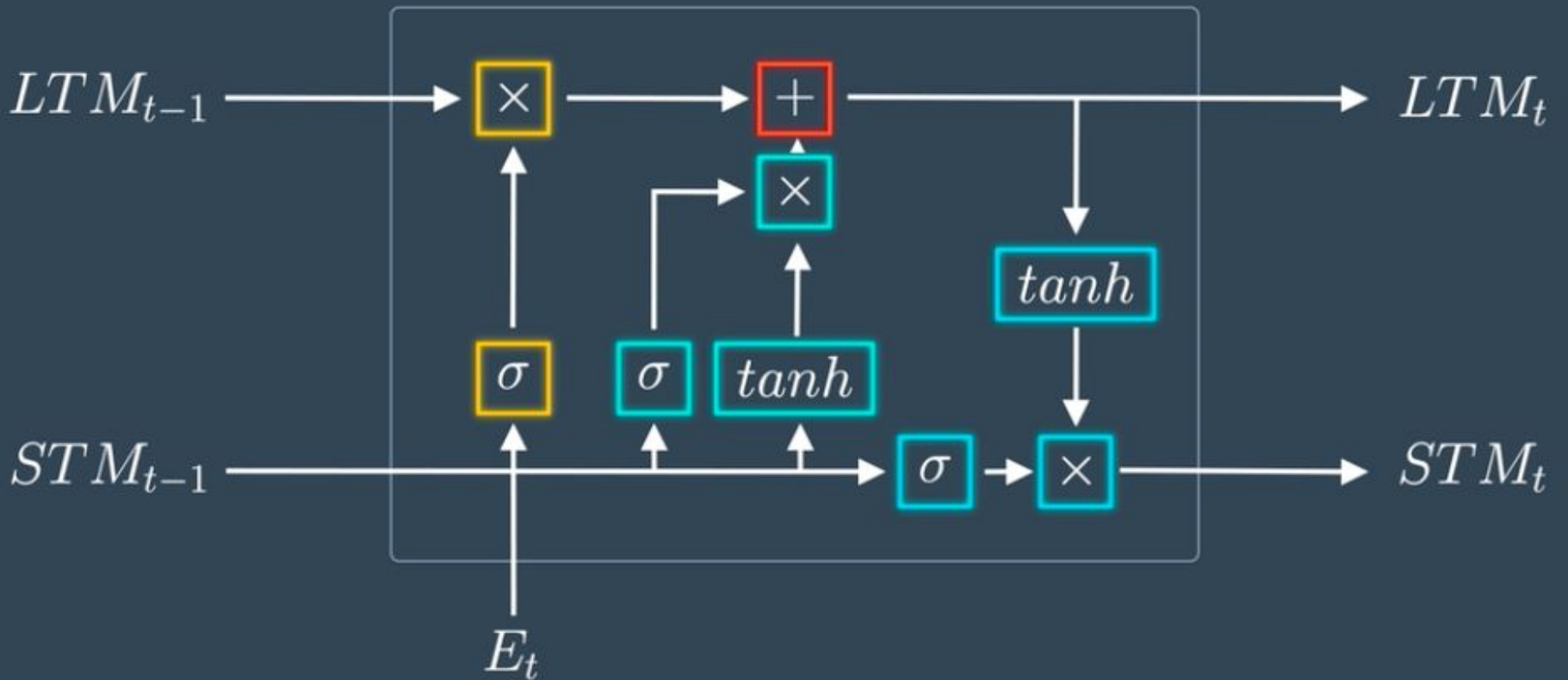
Embedding
https://developers.google.com/machine-learning/crash-course/embeddings/video-lecture



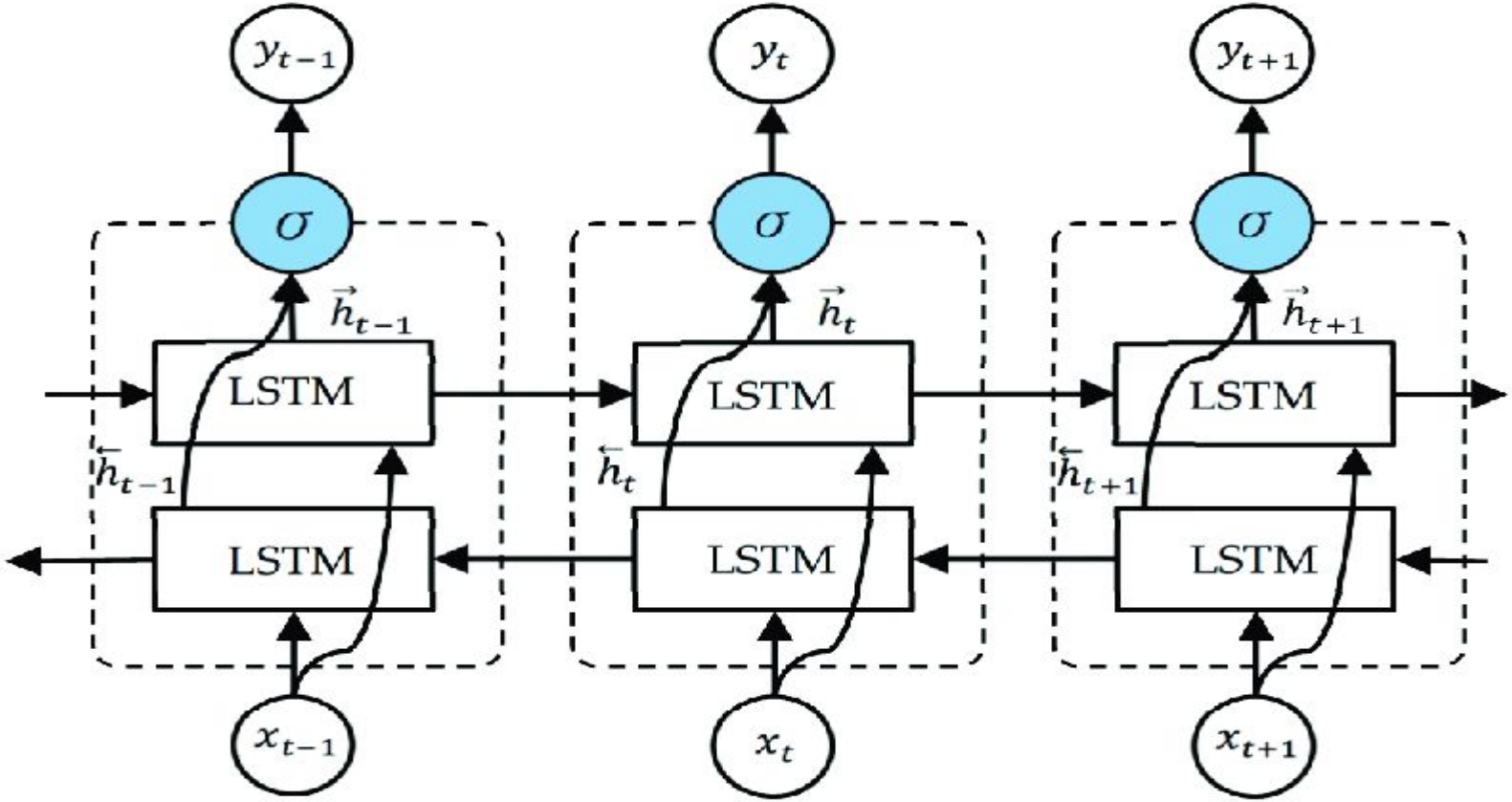
RNN



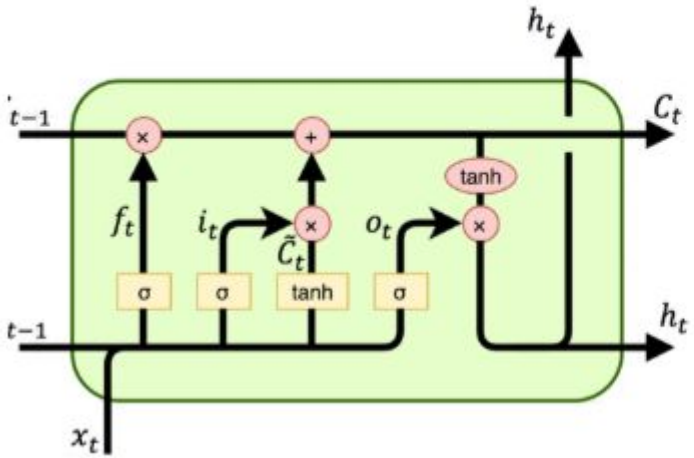
LSTM



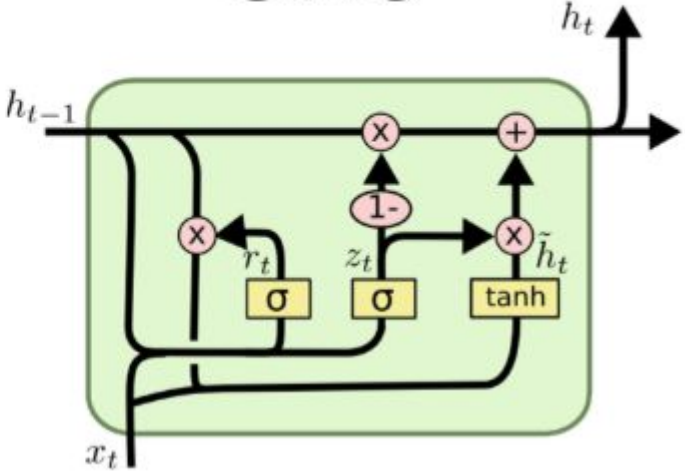
The unfolded architecture of Bidirectional LSTM (BiLSTM) with three consecutive steps.



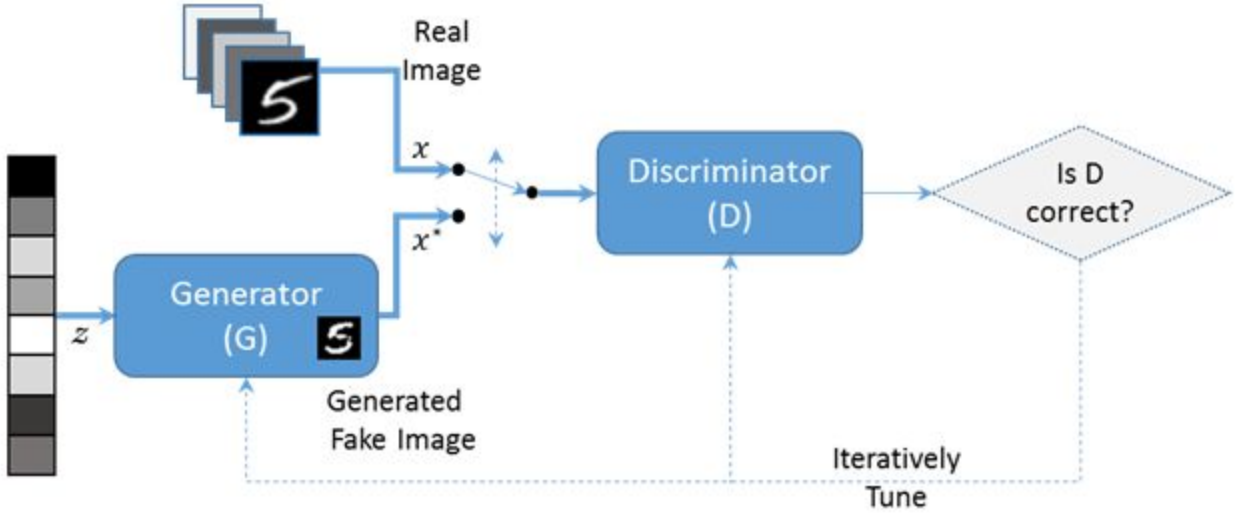
LSTM


h_t
C_t
$t-1$
f_t
i_t
$\tilde{C}_t$
o_t
tanh
σ
σ
tanh
σ
$t-1$
h_t
x_t

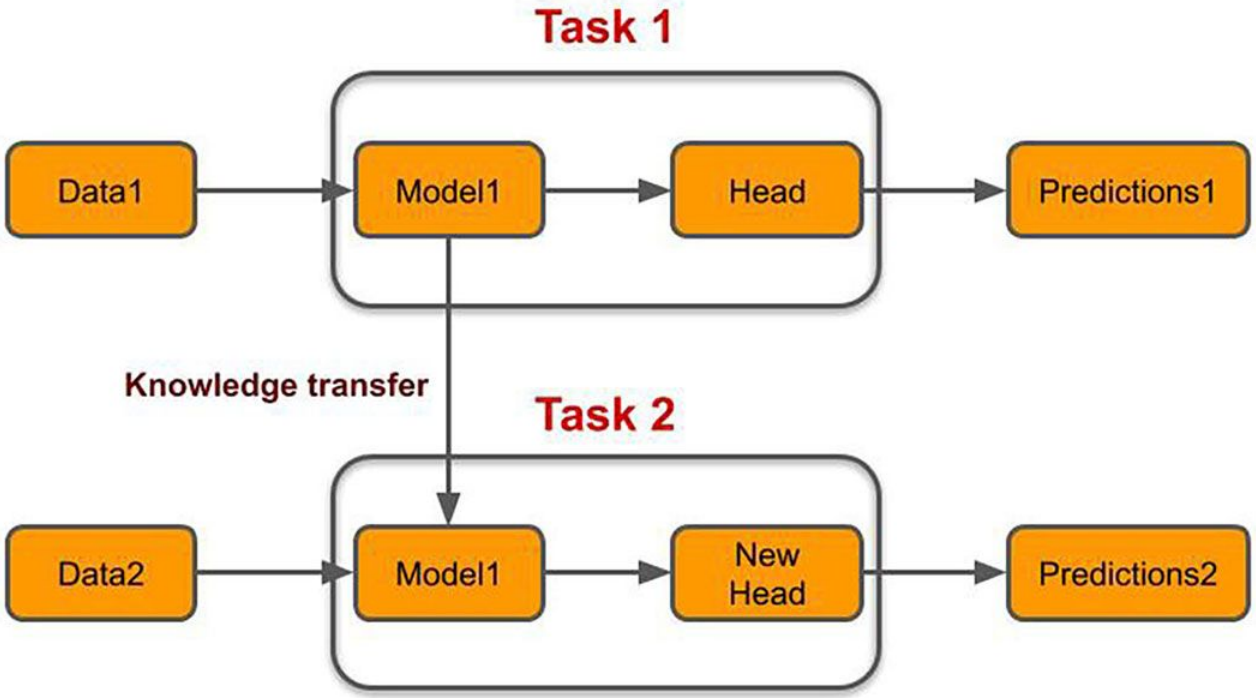
GRU

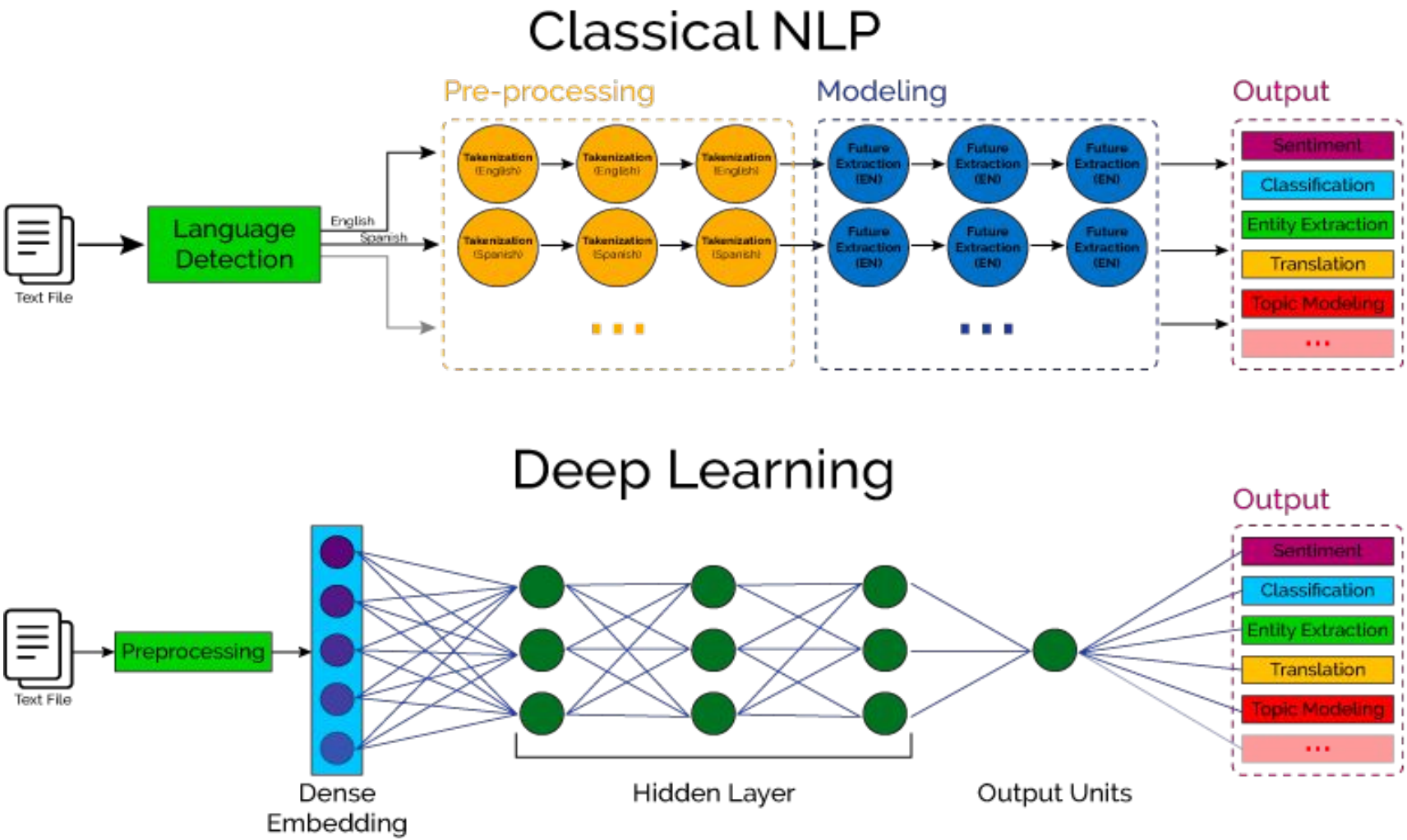

h_t
h_{t-1}
1-
r_t
z_t
$\tilde{h}_t$
σ
σ
tanh
x_t

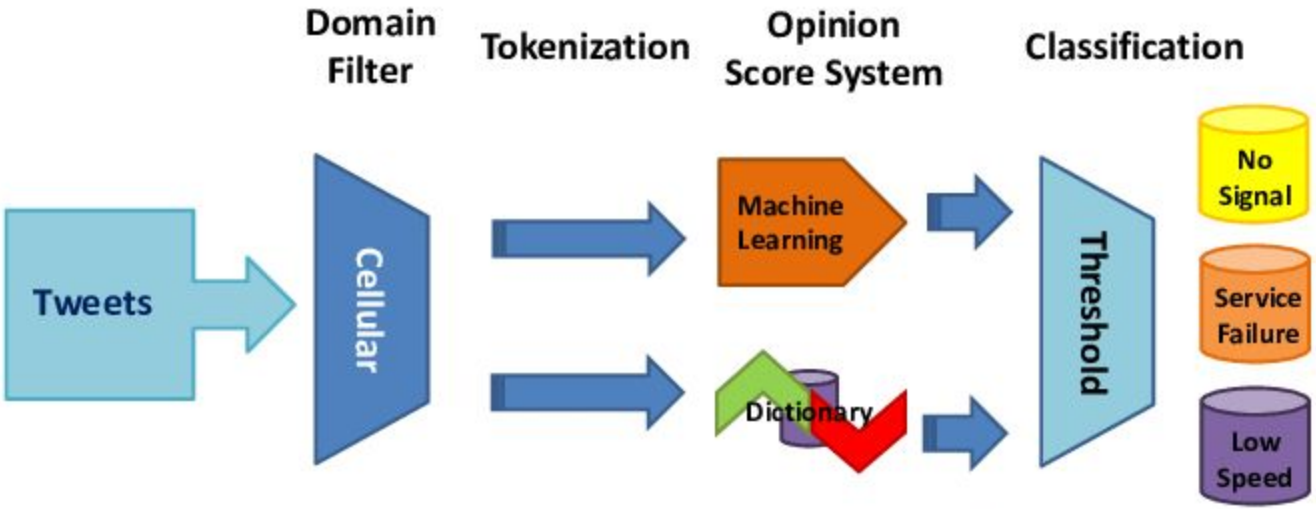
GAN


Real
Image
x
x*
Discriminator
(D)
Is D
correct?
Generator
(G)
z
Generated
Fake Image
Iteratively
Tune

Transfer Learning




Classical NLP
Pre-processing
Modeling
Output
Text File
Language Detection
English
Spanish
Tokenization (English)
Tokenization (Spanish)
Future Extraction (EN)
Sentiment
Classification
Entity Extraction
Translation
Topic Modeling
Deep Learning
Output
Text File
Preprocessing
Dense Embedding
Hidden Layer
Output Units
Sentiment
Classification
Entity Extraction
Translation
Topic Modeling



Lexicon Example:
@AVOID_TalkTalk we've had no service for 4 days. Is this the worst company in the UK?

Domain Keywords:
[TalkTalk] + [no service]

Tokenization:
avoid talktalk we have had no service for 4 days is this the worst company in the UK

Opinion Score (-3):
avoid
no
worst

Classification:
Type: No Service
Opinion Score: -3
GPS: 51.54, -0.11
Location: St Mary Magdalene Academy, N7 8PG, London

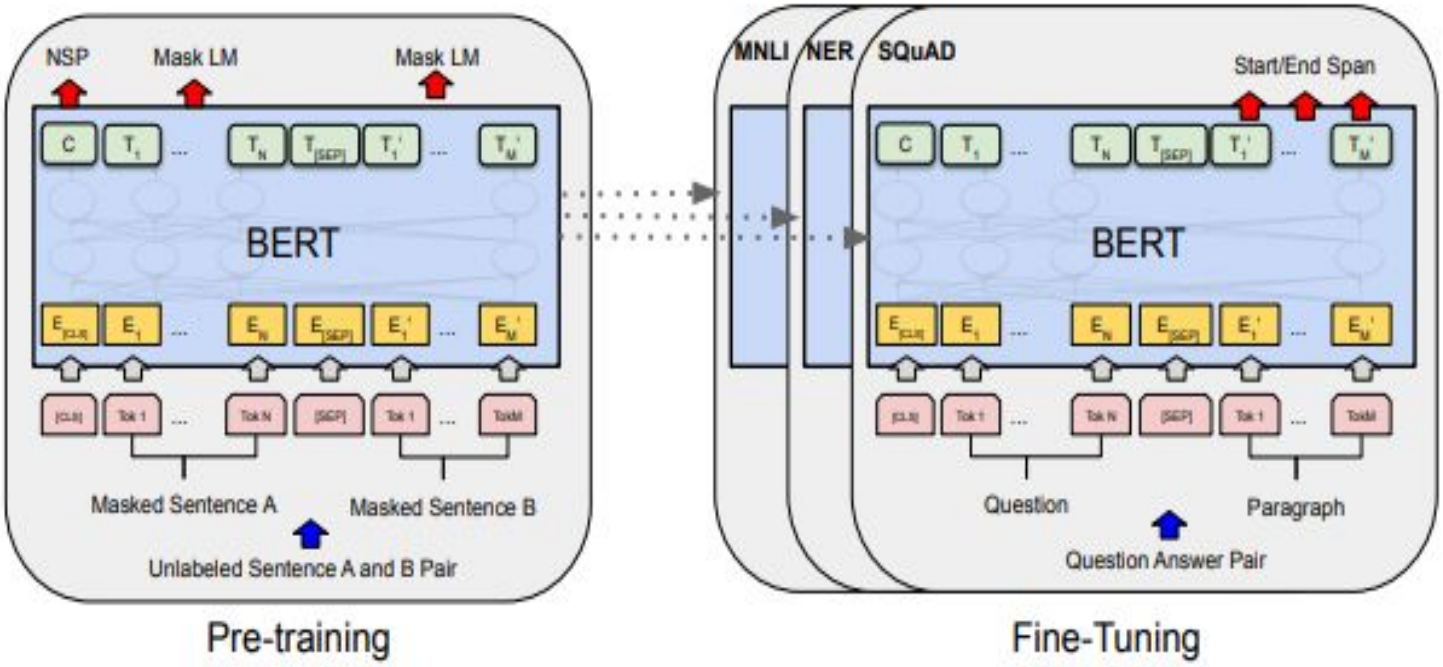
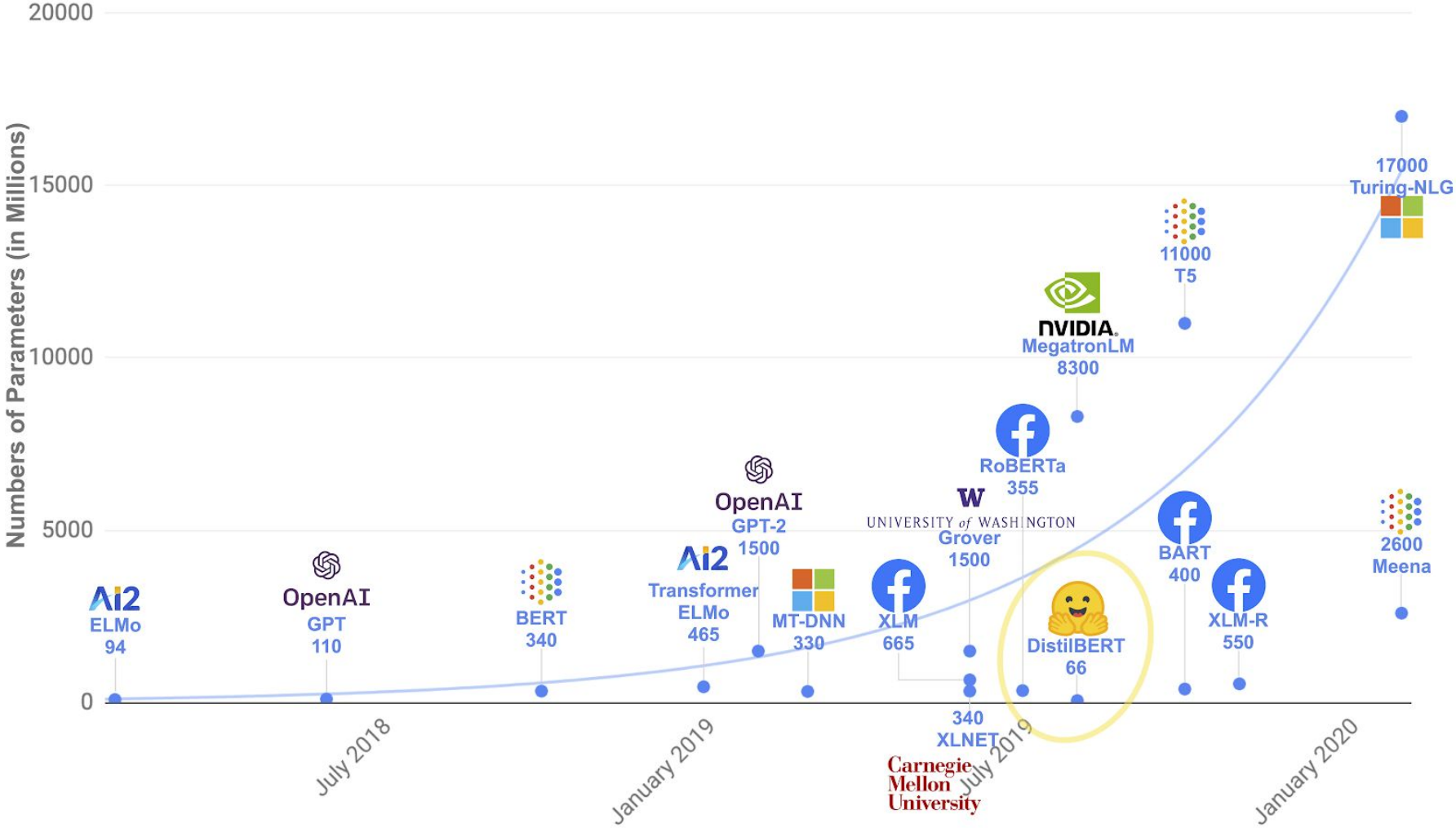
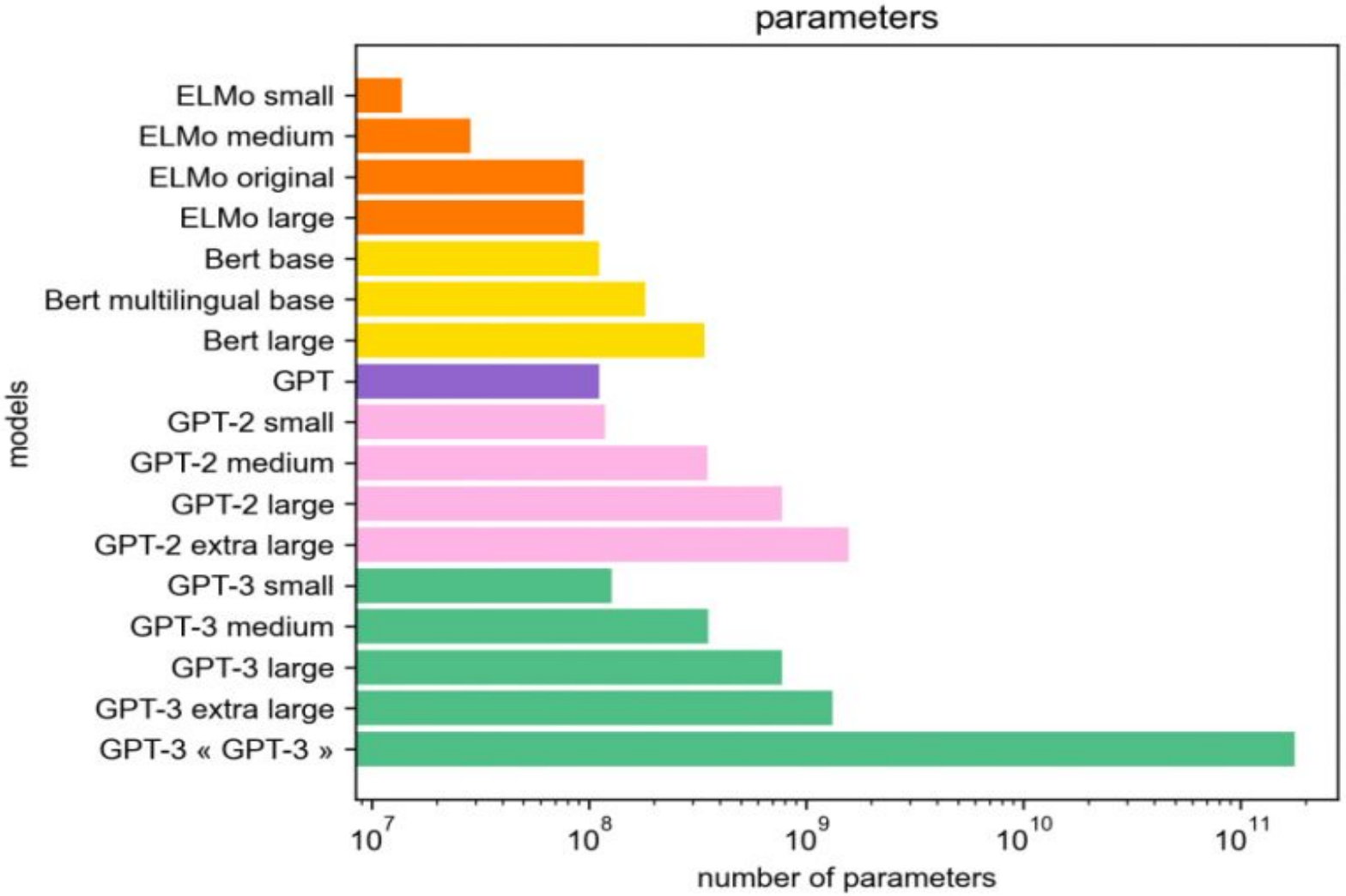


Figure 1: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks. During fine-tuning, all parameters are fine-tuned. `[CLS]` is a special symbol added in front of every input example, and `[SEP]` is a special separator token (e.g. separating questions/answers).


20000
15000
10000
5000
0
Numbers of Parameters (in Millions)
ELMo
94
OpenAI
GPT
110
BERT
340
Transformer
ELMo
465
OpenAI
GPT-2
1500
MT-DNN
330
XLM
665
UNIVERSITY of WASHINGTON
Grover
1500
340
XLNET
Carnegie
Mellon
University
RoBERTa
355
NVIDIA
MegatronLM
8300
DistilBERT
66
11000
T5
BART
400
XLM-R
550
17000
Turing-NLG
2600
Meena
July 2018
January 2019
July 2019
January 2020


parameters
ELMo small
ELMo medium
ELMo original
ELMo large
Bert base
Bert multilingual base
Bert large
GPT
GPT-2 small
GPT-2 medium
GPT-2 large
GPT-2 extra large
GPT-3 small
GPT-3 medium
GPT-3 large
GPT-3 extra large
GPT-3 « GPT-3 »
models
10^7
10^8
10^9
10^{10}
10^{11}
number of parameters