DE VINCI INNOVATION CENTER

DEEP LEARNING

A Modern Approach to Artificial Intelligence

Yliess **HATI**

PHD Student - Computer Science

yliess.hati@**devinci.fr**

LÉONARD DE VINCI PARIS-LA DÉFENSE

ESILV LÉONARD DE VINCI | ÉCOLE D'INGÉNIEURS PARIS-LA DÉFENSE

iiM LÉONARD DE VINCI | INSTITUT DE L'INTERNET ET DU MULTIMEDIA PARIS-LA DÉFENSE

EMLV LÉONARD DE VINCI | ÉCOLE DE MANAGEMENT PARIS-LA DÉFENSE

|00 INTRODUCTION

Perceptron
Rosenblatt
1958

Perceptrons
Minsky & Seymour
1958

Boltzmann Machine
Hinton
1985

CNN
LeCun
1989

Contrastive Divergence
Hinton
2002

GAN
Goodfellow
2014

1959
Hubel & Wiesel
Cat Visual Cortex

1979
Fukushima
NeoCognitron

1986
Smolenski
Harmonium
Hinton
RBM
Rumelhart, Hinton & Williams
MLP
Jordan
RNN

1997
Hochreiter & Schmidhuber
LSTM
Schuster & Paliwal
BRNN

2012
Hinton
Dropout

2017
Sabour, Frosst & Hinton
Capsule Network

DE VINCI INNOVATION CENTER

|00 INTRODUCTION

AlexNet
Krizhevsky, Sutskever & Hinton
2012

ResNet
He, Zhang, Ren & Sun
2015

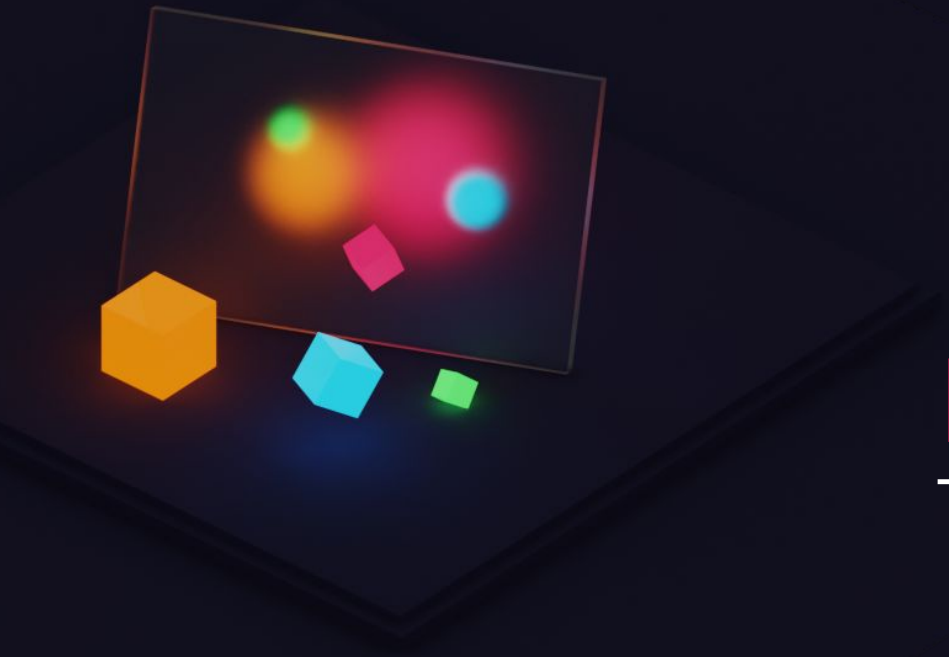
ResNetXt
Xie, Girshick et al.
2019

2014
Simonyan & Zisserman
VGG

Google
Inception Network

2016
Huang et al.
DenseNet

DE VINCI
INNOVATION
CENTER



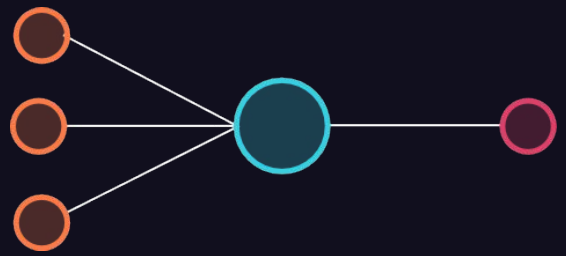
01|

PERCEPTRON

The Beginning and the End

DE VINCI
INNOVATION
CENTER

|01 PERCEPTRON



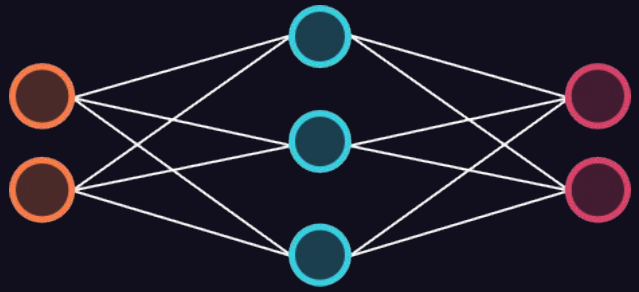
PERCEPTRON

$$\hat{y} = f(wx + b)$$

$$f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases}$$

DE VINCI
INNOVATION
CENTER

|01 PERCEPTRON



MULTILAYER PERCEPTRON

$$\hat{y} = f(w_2 h + b_2)$$

$$h = f(w_1 x + b_1)$$

$$f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases}$$

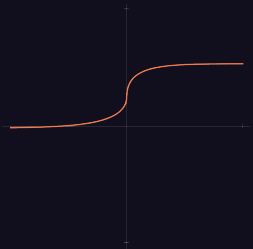
DE VINCI
INNOVATION
CENTER

ACTIVATION FUNCTIONS

Step

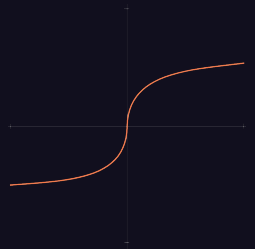
$$f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases}$$

Sigmoid



$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Tanh

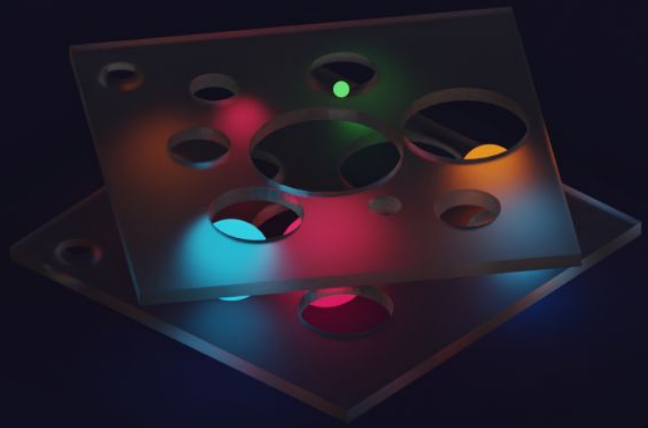


$$\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

|02

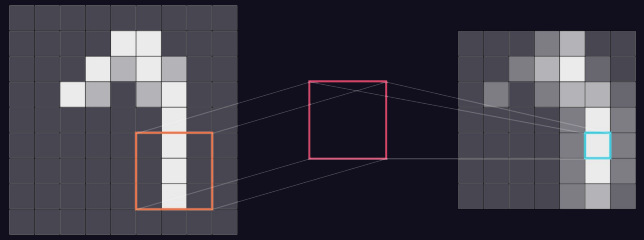
CONVOLUTION

Signal Processing 101



DE VINCI
INNOVATION
CENTER

|02 CONVOLUTION



CONVOLUTION
CROSS CORRELATION

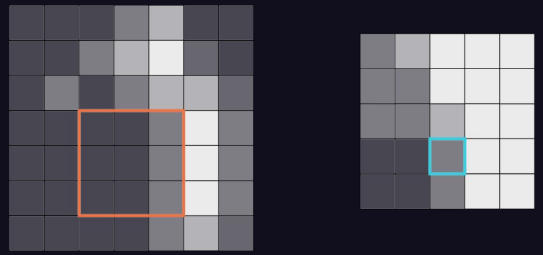
$$(f * g)(x) = \int_{-\infty}^{+\infty} f(x)g(x-t)dt$$

Weight Sharing

DE VINCI
INNOVATION
CENTER

|02 CONVOLUTION

POOLING



Dimensionality Reduction

DE VINCI
INNOVATION
CENTER

|02 CONVOLUTION

CONVOLUTIONAL NEURAL NETWORK

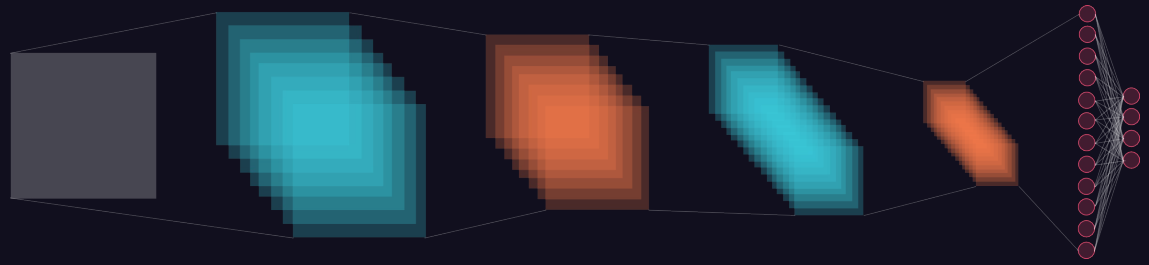


Image Input

Convolution & Pooling

MLP

DE VINCI
INNOVATION
CENTER

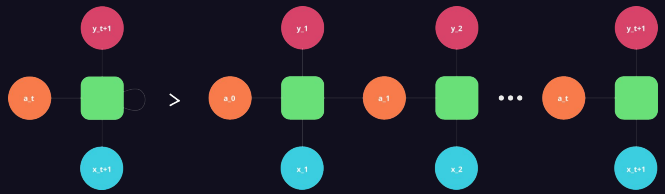
03|

RECURRENT

Backprop Through Time

DE VINCI
INNOVATION
CENTER

|03 RECURRENT



RECURRENT CELLS

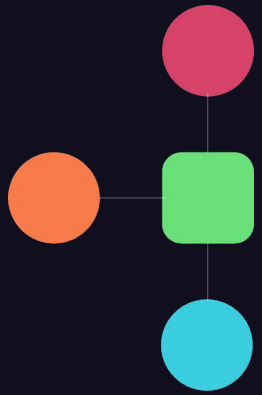
Weight Sharing & Backprop Through Time

$$a_t = g_1(W_{aa}a_{t-1} + W_{ax}x_t + b_a)$$

$$y_t = g_2(W_{ya}a_t + b_y)$$

DE VINCI
INNOVATION
CENTER

|03 RECURRENT



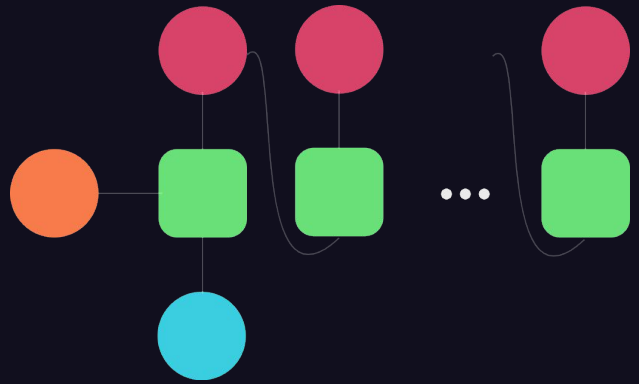
ARCHITECTURES

One to One

Traditional Neural Network

DE VINCI
INNOVATION
CENTER

|03 RECURRENT



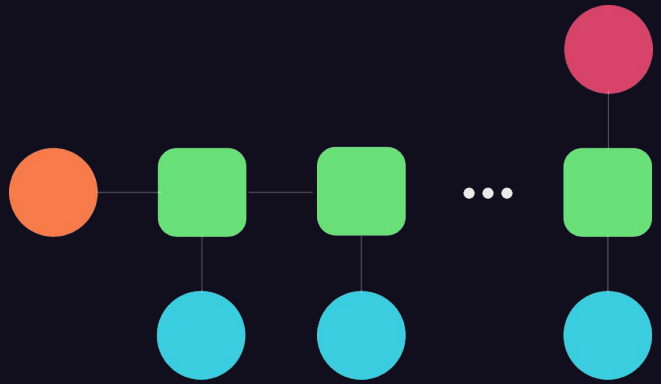
ARCHITECTURES

One to Many

Music Generation

DE VINCI
INNOVATION
CENTER

|03 RECURRENT



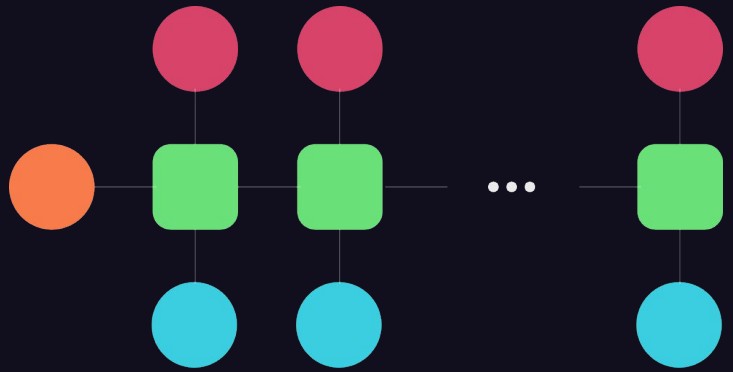
ARCHITECTURES

Many to One

Sentiment Classification

DE VINCI
INNOVATION
CENTER

|03 RECURRENT

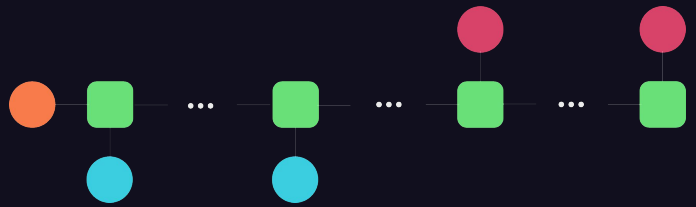


ARCHITECTURES

Many to Many

Name Entity Recognition

DE VINCI
INNOVATION
CENTER



ARCHITECTURES

Many to Many

Machine Translation

|03 RECURRENT

ADVANTAGES

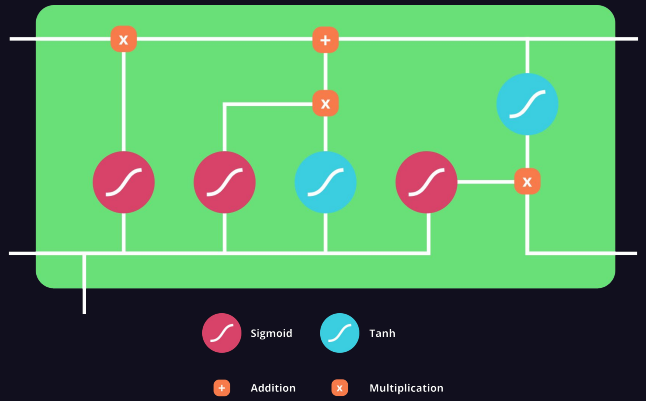
Infinite Input Length
Model **Size Invariant**
Historical Information
Weight Sharing Through Time

DRAWBACKS

Computationally **Slow**
Long Time **Dependency Lost** Over Time
Future Input not Considered
Vanishing/Exploding Gradient

DE VINCI
INNOVATION
CENTER

|03 RECURRENT



LSTM

Gates	I/O
Forget Gate	**Previous** Input
Update Gate	**Cell State**
Output Gate	**Output** State

Still **Suffers** from **Exploding Gradient**

DE VINCI
INNOVATION
CENTER