DE VINCI INNOVATION CENTER

DEEP LEARNING

A Modern Approach to Artificial Intelligence

Ylíess **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

1959
Hubel & Wiesel
Cat Visual Cortex

1979
Fukushima
NeoCognitron

Boltzmann Machine
Hinton
1985

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

CNN
LeCun
1989

1997
Hochreiter & Schmidhuber
LSTM
Schuster & Paliwal
BRNN

Contrastive Divergence
Hinton
2002

2012
Hinton
Dropout

GAN
Goodfellow
2014

2017
Sabour, Frosst & Hinton
Capsule Network

DE VINCI INNOVATION CENTER

|00 INTRODUCTION

AlexNet
Krizhevsky, Sutskever & Hinton
2012

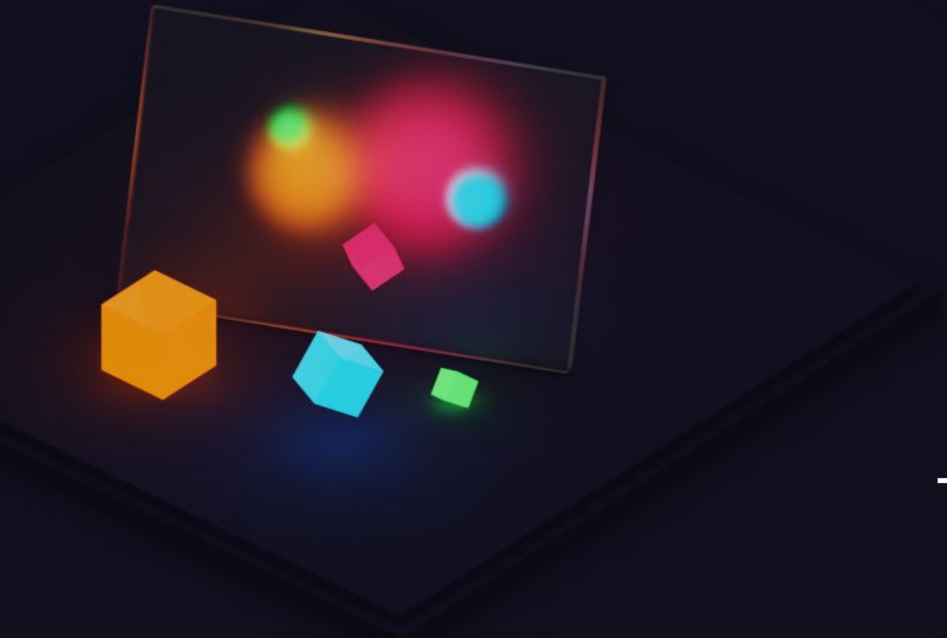
2014
Simonyan & Zisserman
VGG
Google
Inception Network

ResNet
He, Zhang, Ren & Sun
2015

2016
Huang et al.
DenseNet

ResNetXt
Xie, Girshick et al.
2019

DE VINCI INNOVATION CENTER



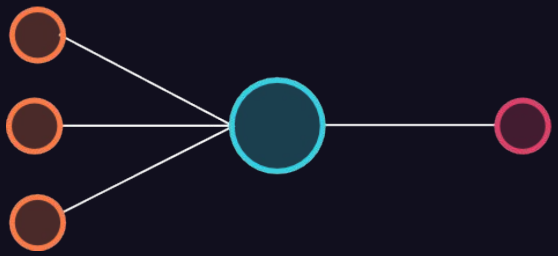
01|

PERCEPTRON

The Beginning and the End

DE VINCI INNOVATION CENTER

|01 PERCEPTRON

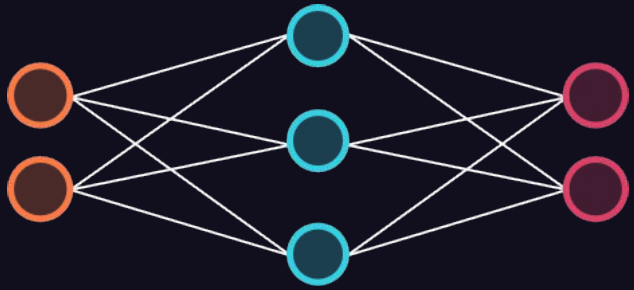


PERCEPTRON

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

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

|01 **PERCEPTRON**



MULTILAYER PERCEPTRON

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

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

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

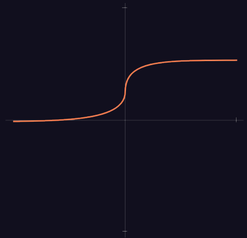
DE VINCI INNOVATION CENTER

ACTIVATION FUNCTIONS

Step

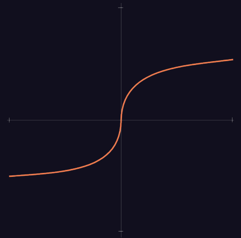
$$f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases}$$

Sigmoid



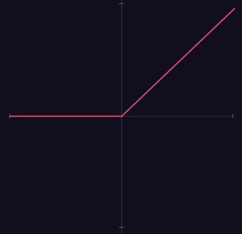
$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Tanh



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

ReLU



$$relu(x) = max(0, x) = x^{+}$$

ACTIVATION FUNCTIONS

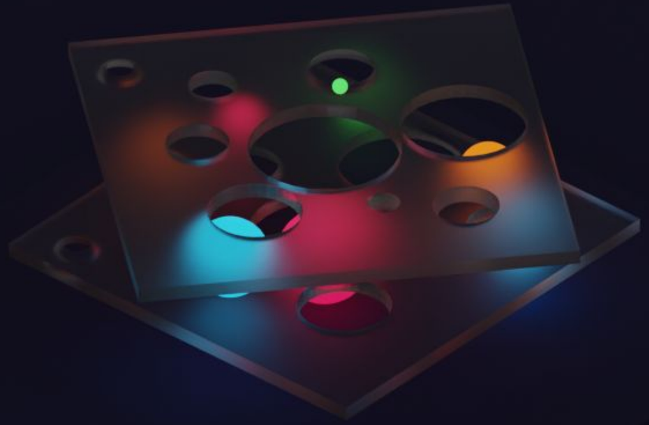
Softmax

$$p_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

|02

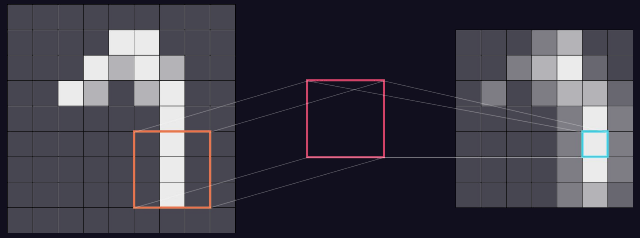
CONVOLUTION

Signal Processing 101



DE VINCI INNOVATION CENTER

|02 CONVOLUTION



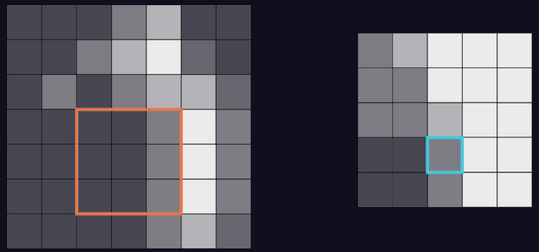
CONVOLUTION
CROSS CORRELATION

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

Weight Sharing

|02 CONVOLUTION

POOLING

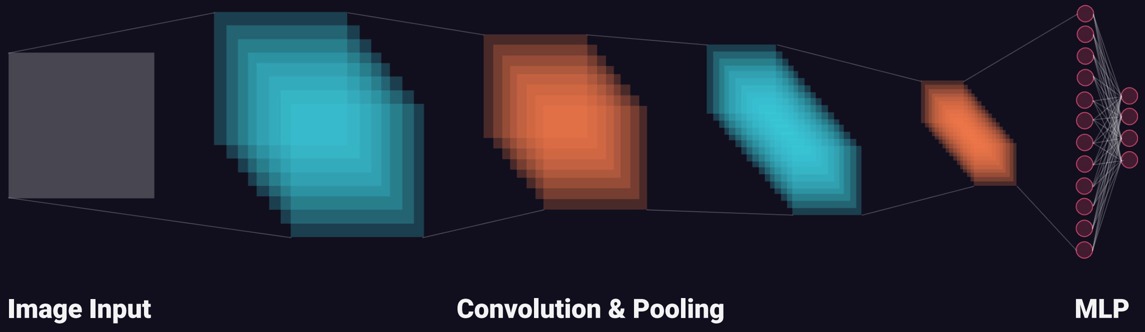


Dimensionality Reduction

DE VINCI INNOVATION CENTER

|02 CONVOLUTION

CONVOLUTIONAL NEURAL NETWORK



DE VINCI INNOVATION CENTER

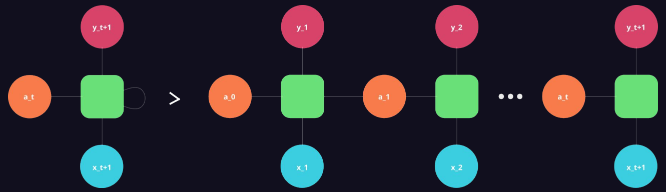
03|

RECURRENT

Backprop Through Time

DE VINCI
INNOVATION
CENTER

|03 RECURRENT



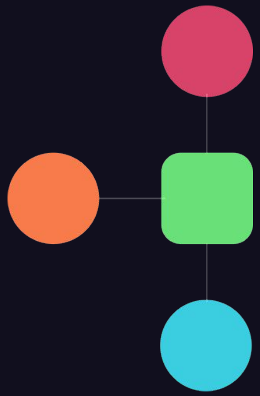
RECURRENT CELLS

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

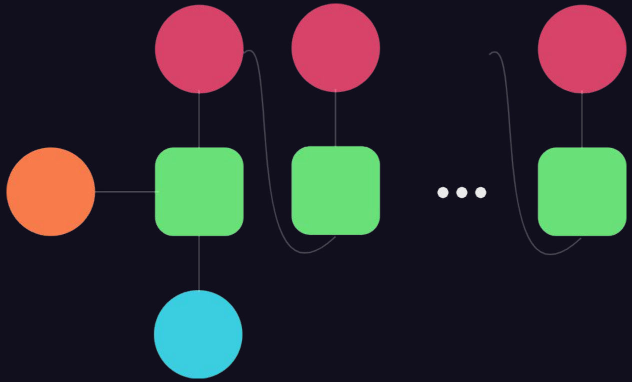


ARCHITECTURES

One to One

Traditional Neural Network

|03 RECURRENT

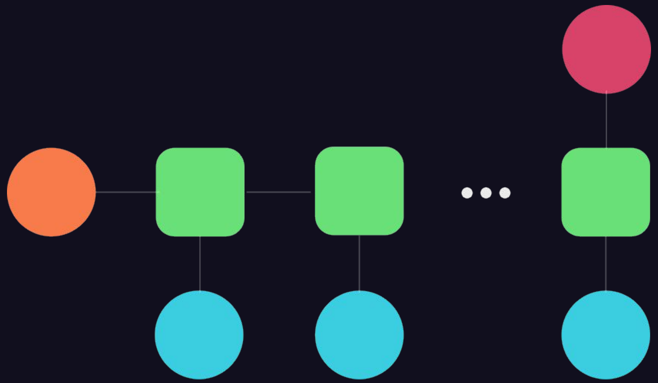


ARCHITECTURES

One to Many

Music Generation

|03 RECURRENT



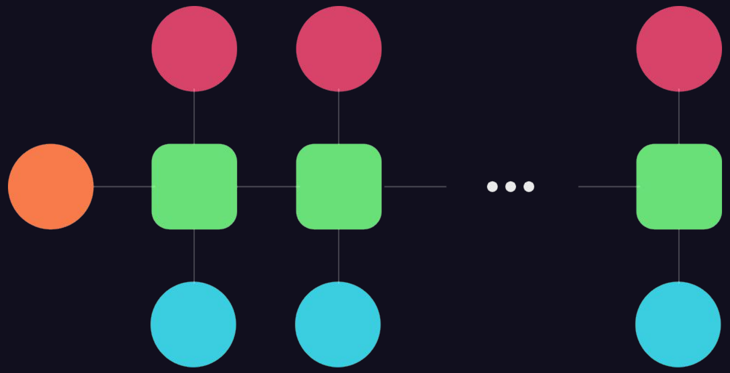
ARCHITECTURES

Many to One

Sentiment Classification

DE VINCI INNOVATION CENTER

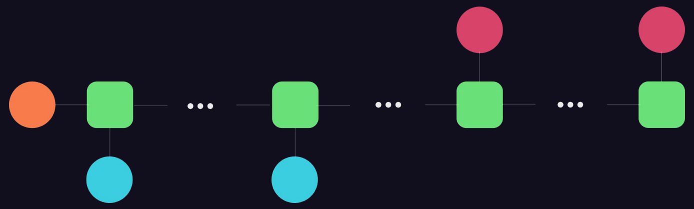
|03 **RECURRENT**



ARCHITECTURES

Many to Many

Name Entity Recognition



ARCHITECTURES

Many to Many

Machine Translation

|03 RECURRENT

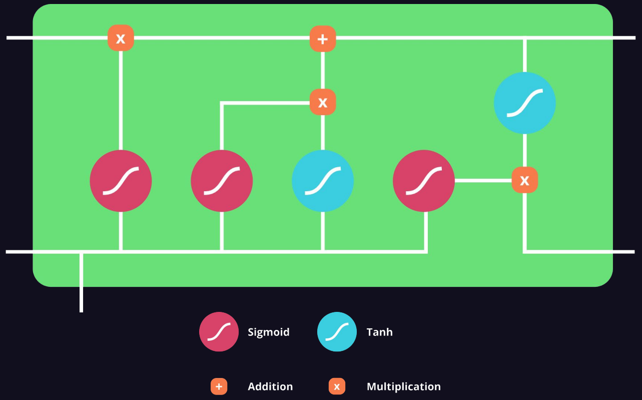
ADVANTAGES

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

|03 RECURRENT



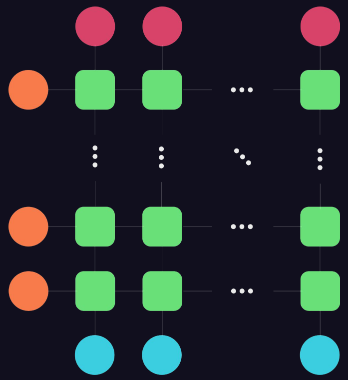
LSTM

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

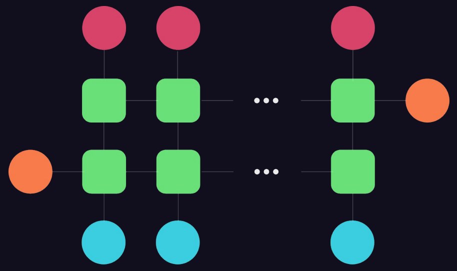
Still **Suffers** from **Exploding Gradient**

|03 RECURRENT

STACKED



BIDIRECTIONAL



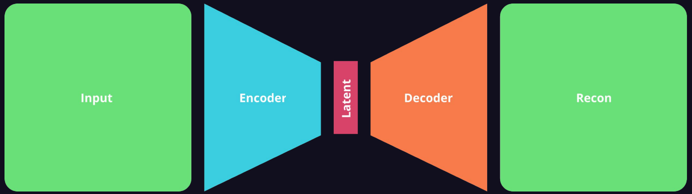
DE VINCI INNOVATION CENTER

|04

AUTO-ENCODER

Hierarchical Compression is Key

DE VINCI
INNOVATION
CENTER



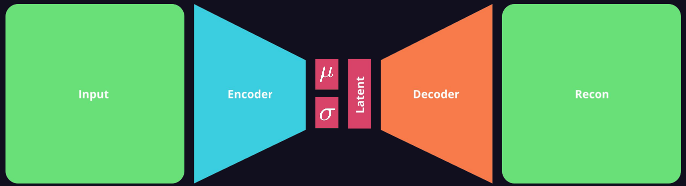
AUTO-ENCODER

$$z = e(x) \qquad \hat{y} = d(z)$$

$$loss = \frac{1}{N} \sum_i^N (\hat{y}_i - y_i)^2$$

DE VINCI INNOVATION CENTER

VARIATIONAL AUTO-ENCODER



$$< \mu, \sigma >= e(x)$$
$$z = \mu \cdot \epsilon + \sigma$$
$$\hat{y} = d(z)$$

$$\epsilon \sim \mathcal{N}(0,1)$$
$$z \sim \mathcal{N}(\mu, \sigma)$$

$$loss = \frac{1}{N}\sum_i^N (\hat{y}_i - y_i)^2 + KL(\mathcal{N}(\mu_i, \sigma_i)||\mathcal{N}(0,1))$$

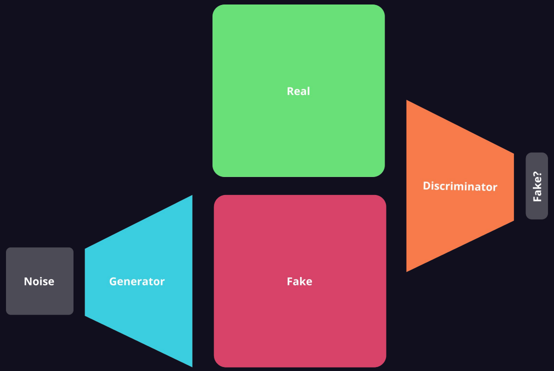
DE VINCI INNOVATION CENTER

05|

GENERATIVE ADVERSARIAL NETWORK

Min Max for the Win

DE VINCI INNOVATION CENTER



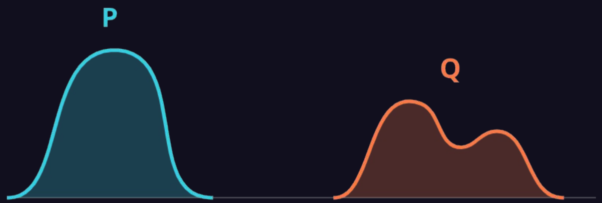
GENERATIVE ADVERSARIAL NETWORK

$$\underset{G}{min}\,\underset{D}{max} = \mathbb{E}_{x\sim p_r}[log(D(x))] + \mathbb{E}_{x\sim p_g}[1 - log(D(x))]$$

DE VINCI INNOVATION CENTER

|05 GENERATIVE ADVERSARIAL NETWORK

WASSERSTEIN



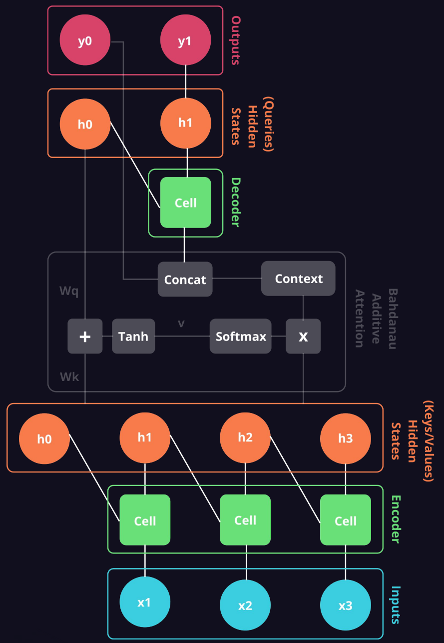
$$W_{(p_r,p_g)} = \inf_{\gamma \sim \pi(p_r,p_g)} \mathbb{E}_{(x,y)\sim\gamma}[||x-y||]$$

|06

ATTENTION

It is All You Need

|06 ATTENTION



BAHDANAU ATTENTION

$$score(k, q) = V^T tanh(W_k k + W_q q)$$

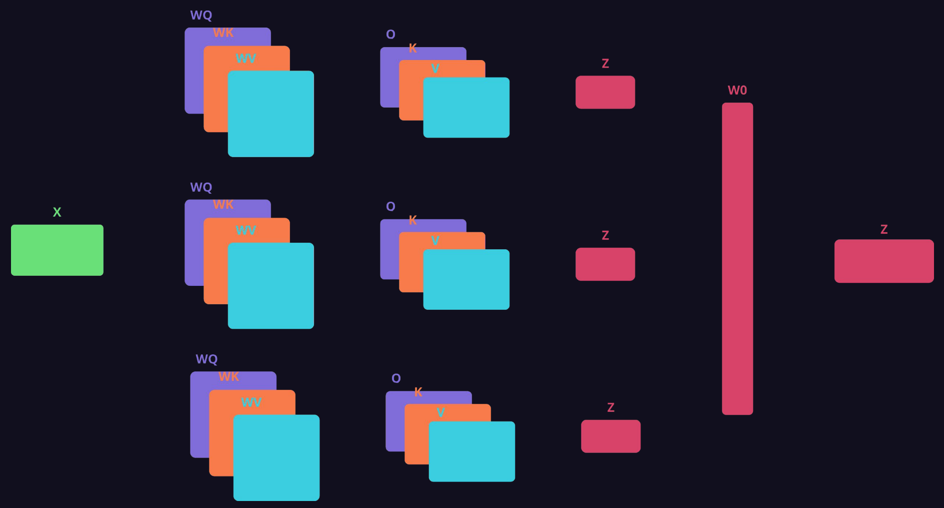
$$attention(k, q) = softmax(score(k, q))$$

$$context = v \cdot attention(k, q)$$

DE VINCI INNOVATION CENTER

|06 ATTENTION

SELF ATTENTION



DE VINCI
INNOVATION
CENTER