MASTER CESMA 2018

deeplearningitalia

Deep Learning
and the «*Deep Learning Italia*» Project

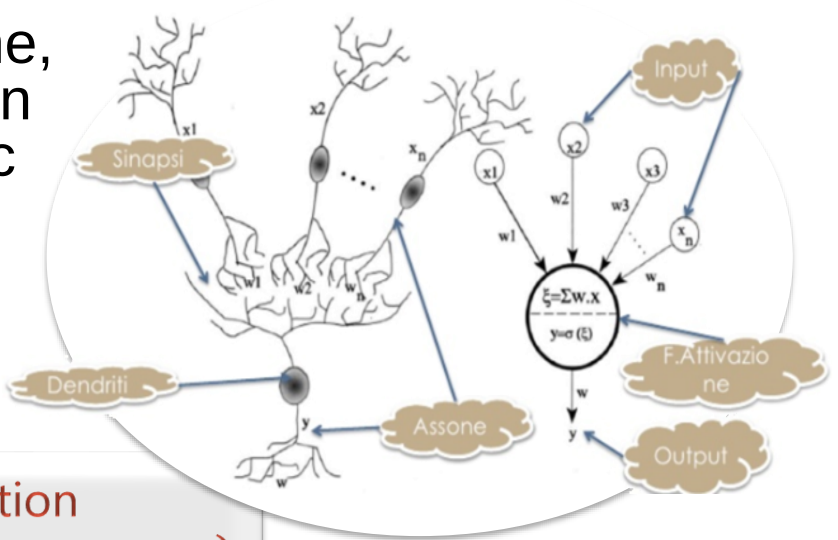
Francesco Pugliese, PhD

f.pugliese@deeplearningitalia.com

http://www.deeplearningitalia.com

ARTIFICIAL NEURAL NETWORKS (ANNs)

ANNs were introduced, for the first time, by 1943, in a work on the formalization of neural activity in propositional logic form (*McCulloch & Pitts, 1943*).
We can define ANNs as a simple model of biological organisms' nervous system.



Neuron Activation
$$A_j = \sum_{i=1}^{N} w_{ij} X_i - \theta_i$$

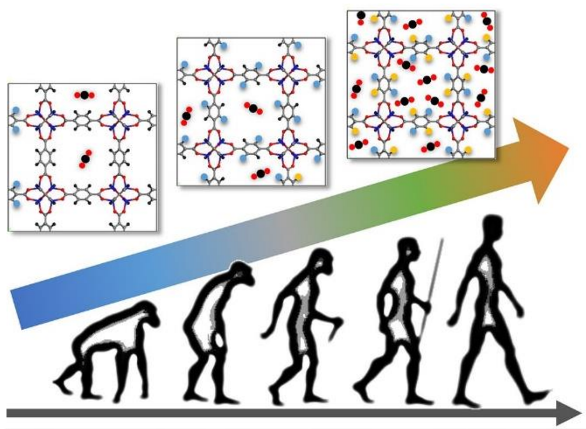
Activation function
$$y_j = \Phi(A_j) = \Phi\left(\sum_{i=1}^{N} w_{ij} X_i - \theta_i\right)$$

In data mining: Methods have been developed to produce comprehensible models and reduce training times:
1)Rule extraction: extraction of symbolic models from pre-trained neural networks.
2)Learn simple, easy-to-understand neural networks.

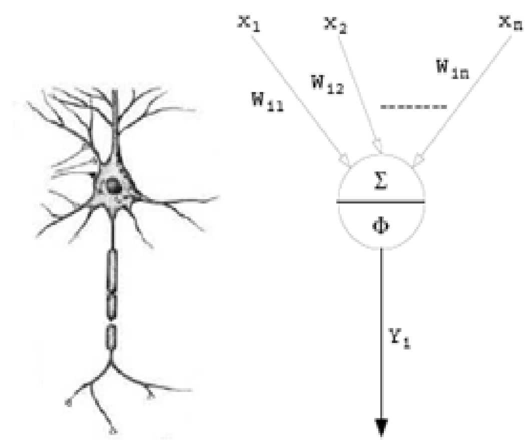
One traditional application ANN: Evolutionary robotics

In order to overcome the problems associated with the robotic system decomposition of traditional approaches (i.e. behavior-based robotics), evolutionary robotics can be used, where the robotic system is able to self-organize

[*Nolfi, S., Floreano, D ., 2000*].

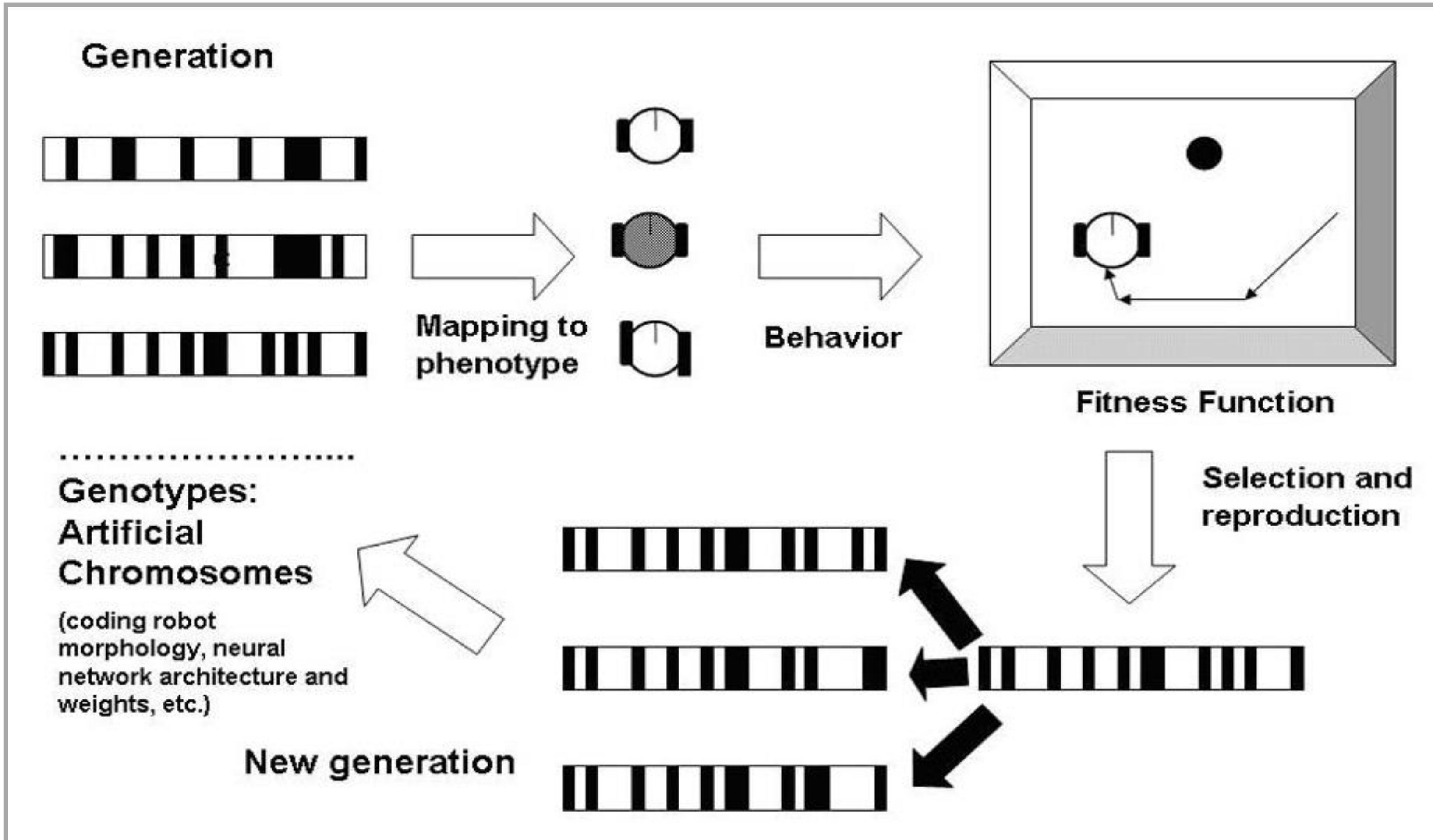


NeuroGenetic Algorithm



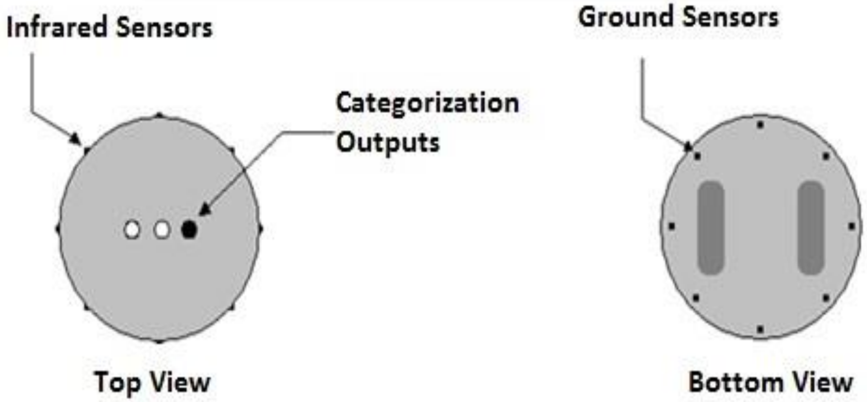
Control Neural Network

Evolutionary Robotics

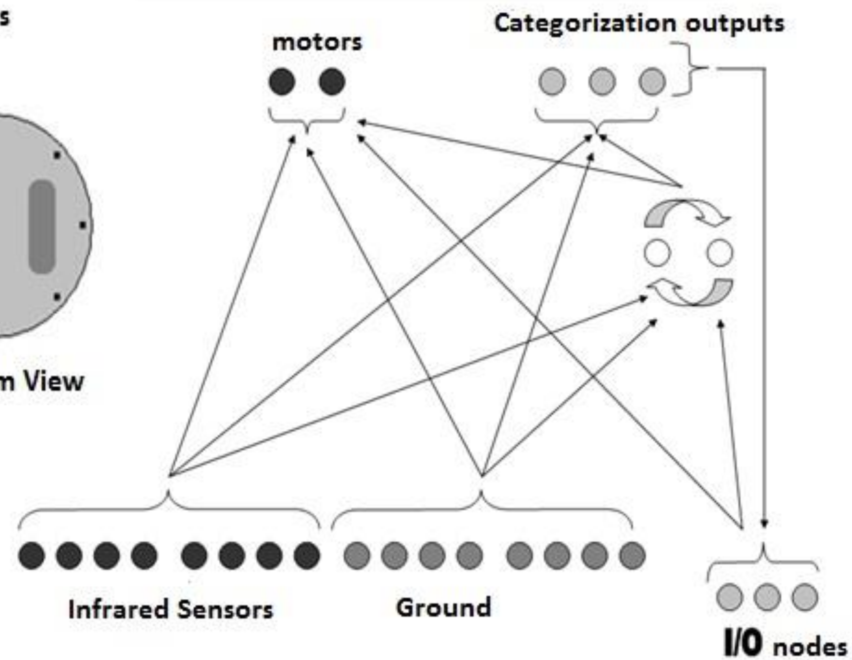


EXPERIMENTAL SETUP N.1

ROBOT'S CHASSIS



NEURAL NETWORK

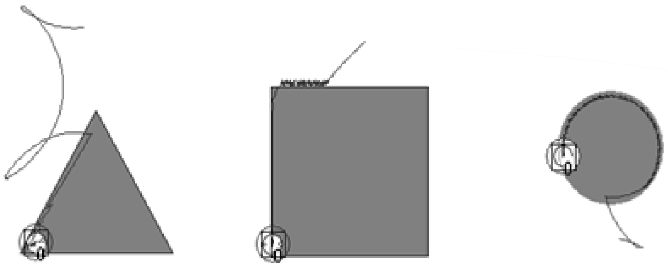


LEAKY Activation

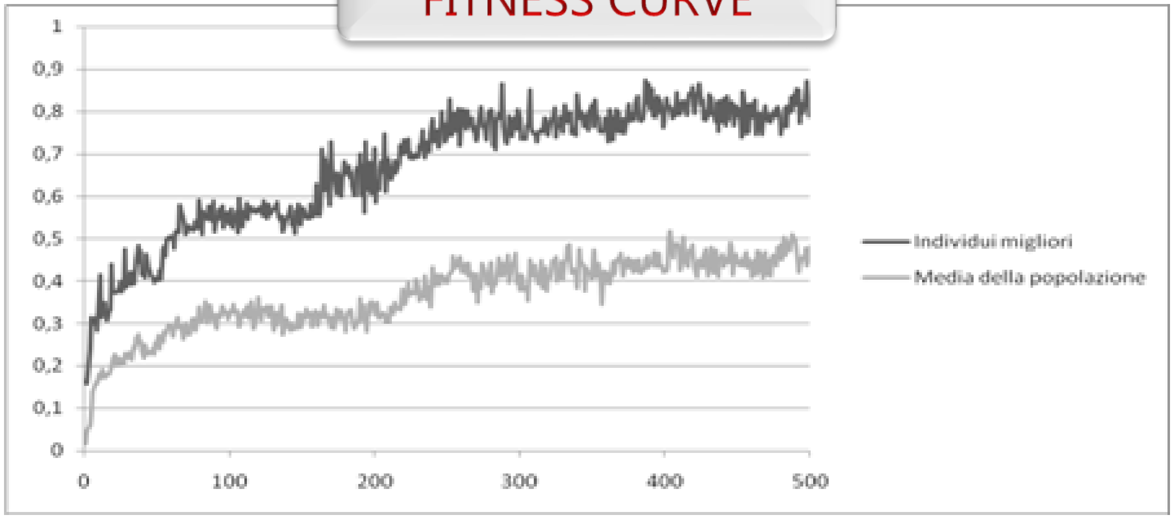
$$A_j = t_j + \sum \mathrm{w_{ij}} \mathrm{O_i} \ , \ O_j = \delta_j O^{t-1} + (1-\delta_j)\left(1+\frac{1}{e^{A_j}}\right), 0 \le \delta_j \le 1$$

EXPERIMENTAL SETUP N.1

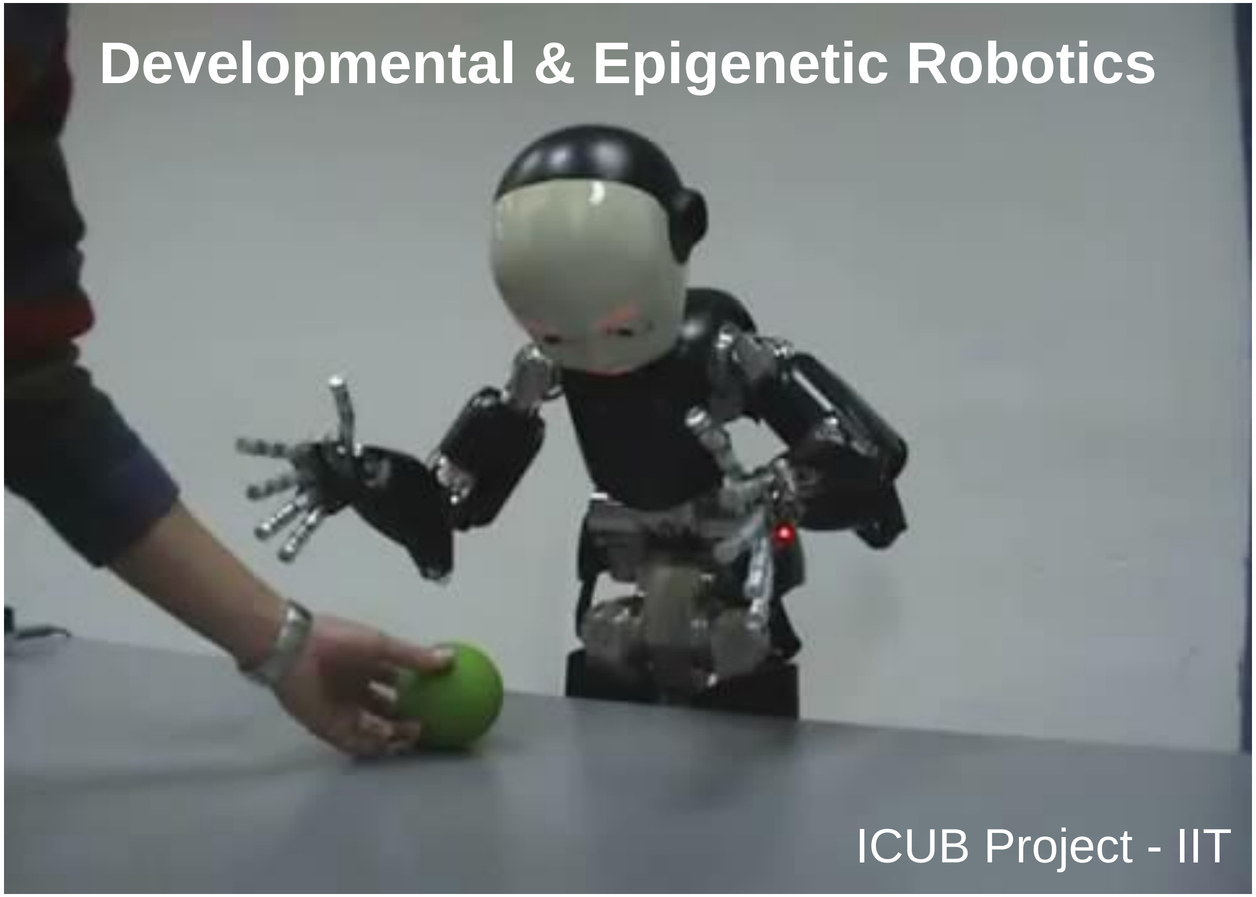
RESULTS



FITNESS CURVE

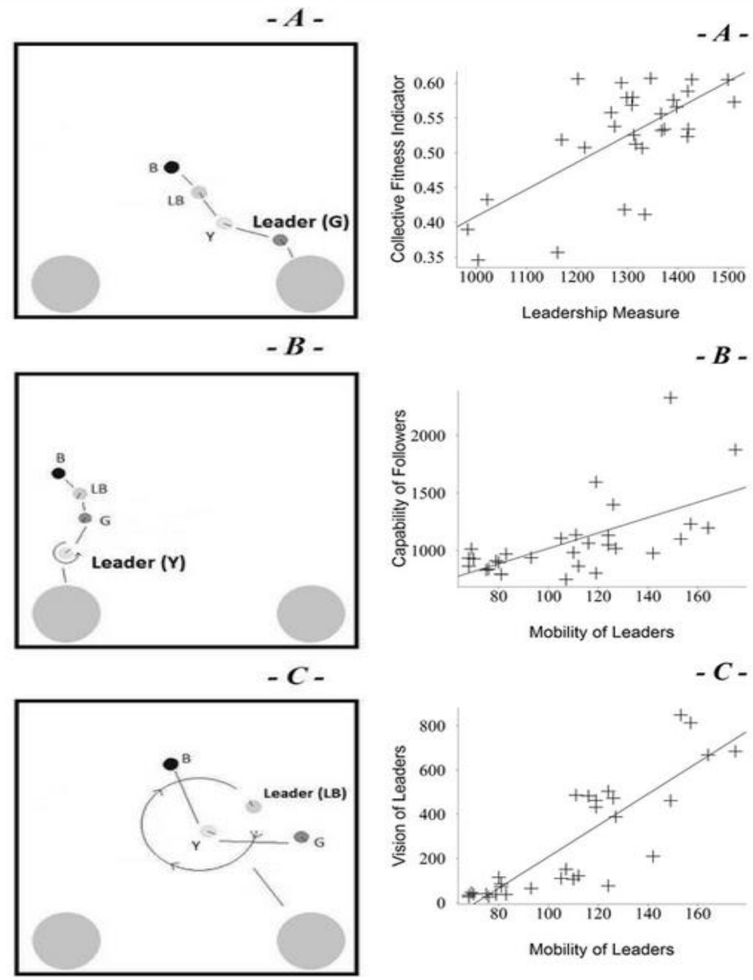


MAY Robotics help to understand social and psychological problems?



Emergence of Leadership in Robots

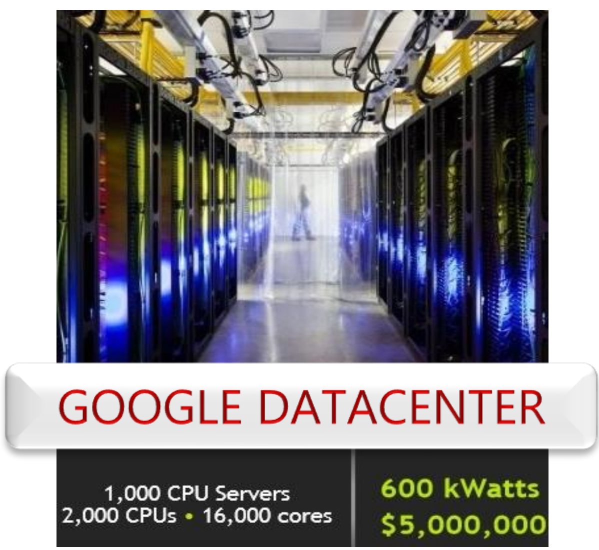
- Behavioural and quantitative analysis indicate that a form of leadership emerges
- Groups with a leader are more effective than groups without.
- The most skilled individuals in a group tend to be the leaders.
- Further analysis reveals the emergence of different “styles” of leadership (active and passive).

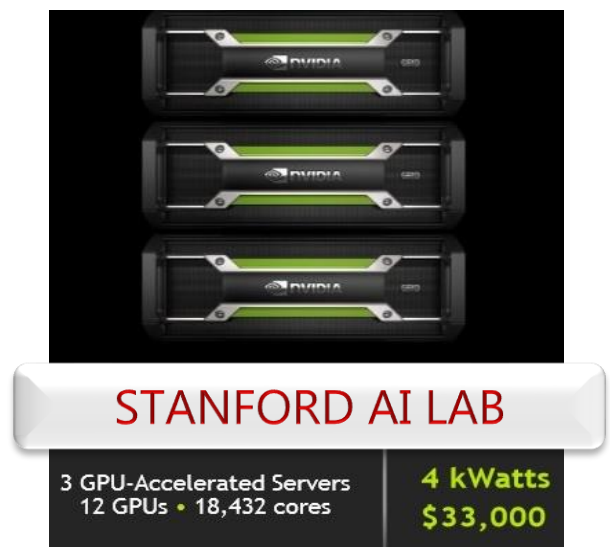


- A - Passive Leadership. - B - Weak Active Leadership.
- C - Strong Active Leadership.

DEEP LEARNING: Neural Networks become more effective

In recent years **Deep Neural Networks** have achieved noticeably breakthroughs in research (*Bengio, 2009*). This new methodology dealing with deep neural networks and their training algorithms is called *“Deep Learning”*. **<u>So far, in all the experiments, the resulting performances were many magnitudes better than other machine learning techniques available.</u>**





DEEP LEARNING: a cutting-edge approach to Computer Vision and NLP

GPU

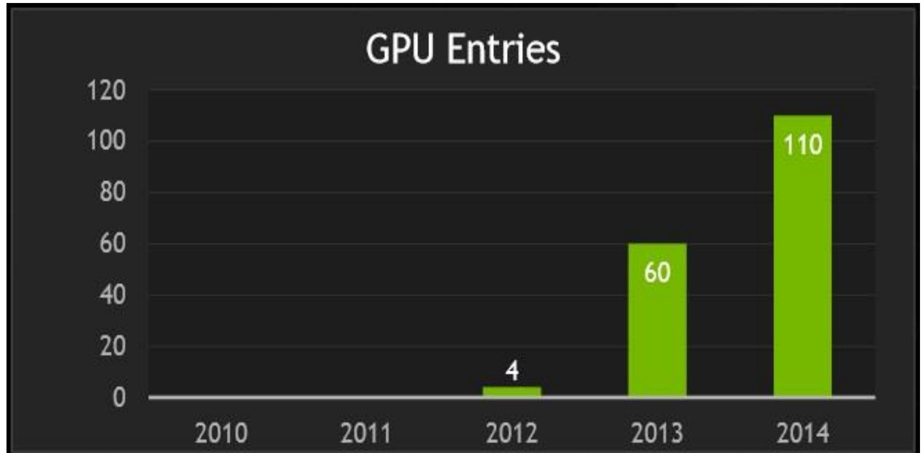
- **The advent of GPUs makes possible the training of very large neural networks with even more than 150 millions of parameters.**

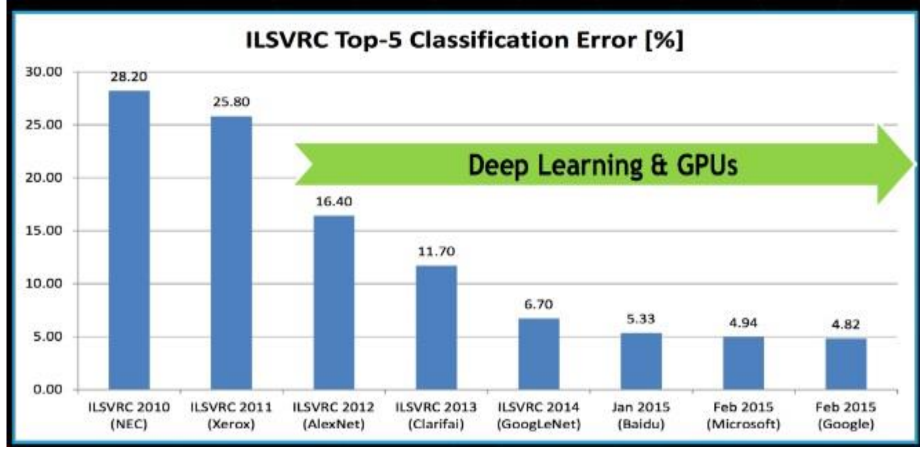
BIG DATA

- **A new generation of larger training and test sets.**

DROPOUT

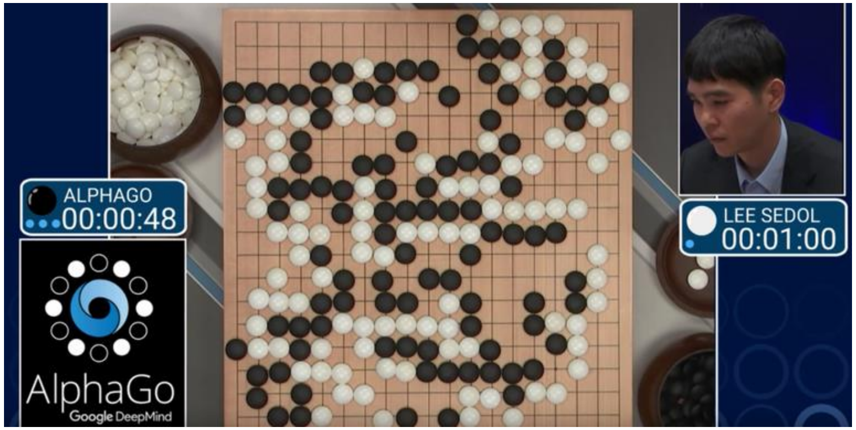
- **Better model regularization techniques have been discovered such as *"Dropout"* or *"Data Augmentation"***





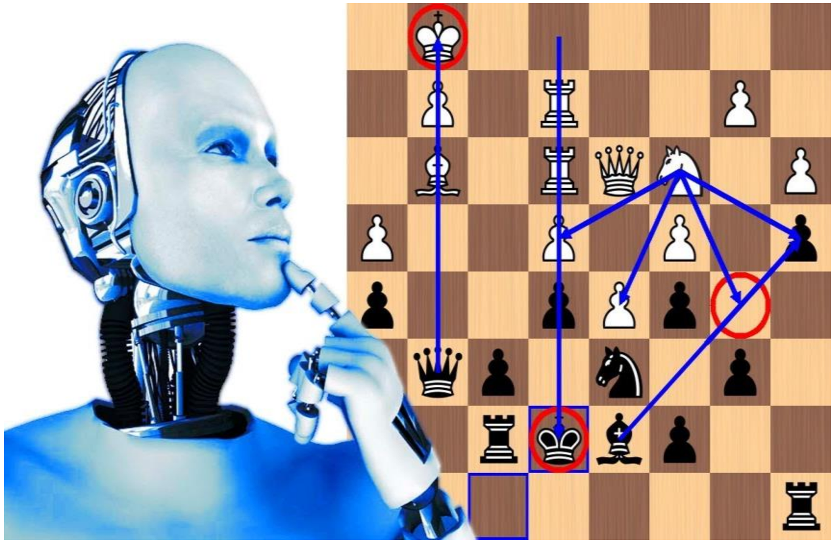
Why Deep Learning over-performed traditional statistics models?

- "Deep Learning" approaches can be **end-to-end trained** without a task-specific feature engineering.
- **These model are scalable**: adding GPUs they can be trained faster.
- **"Deep Learning is killing every problem in AI"** (*Elizabeth Gibney, 2016*)
- **Basically, statistics is not able to deal with very high dimensionalities of data as Deep Learning does**.



Alpha Zero: Mastering the games of Go and Chess without Human Knowledge

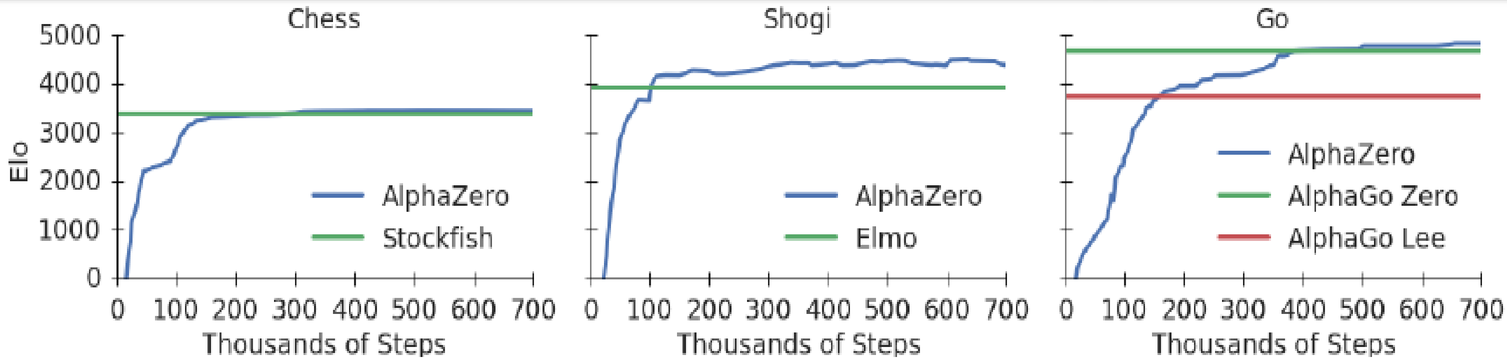
- **In Just 4 Hours, Google's AI Mastered All The Chess Knowledge in History**
- **"I always wondered how it would be if a superior species landed on Earth and showed us how they played chess. Now I know." grandmaster Peter Heine Nielsen.**

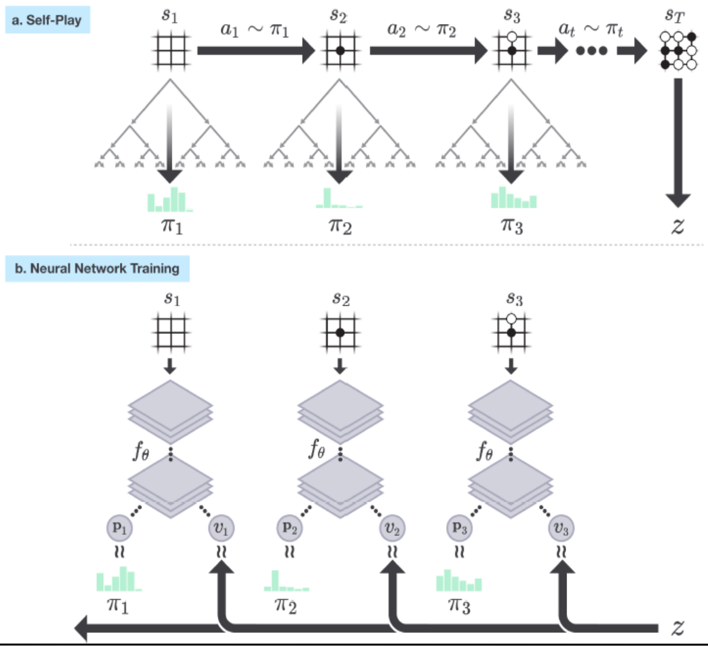


- **Google's AlphaZero Destroys Stockfish In 60 Game Matches**

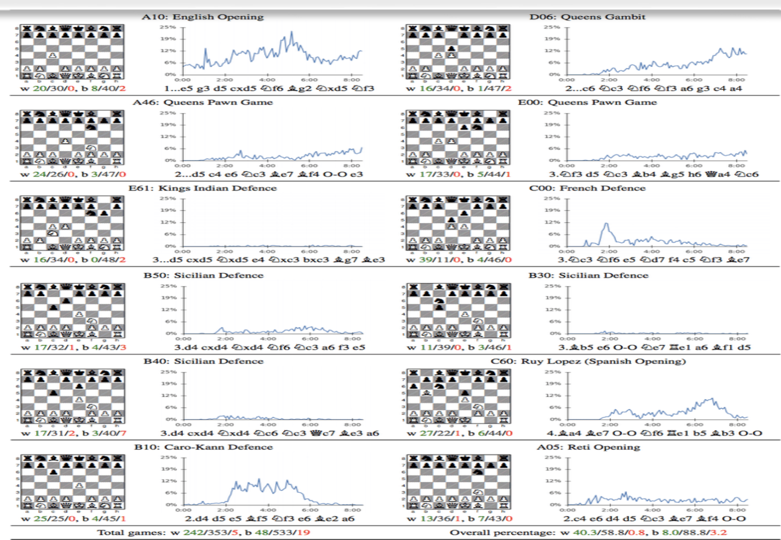
"This algorithm could run cities, continents, universes.“
***PETER DOCKRILL** (**Senior Writer**)*

Alpha Zero IS an Artificial Intelligence, it IS NOT just a Chess Engine..



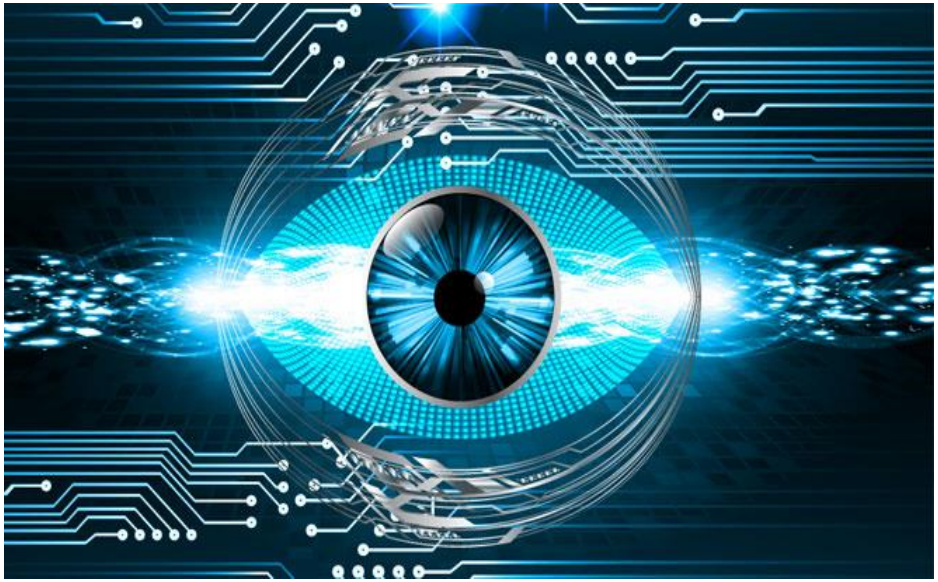


12 Chess Openings Discovered by Alphazero



Computer Vision: Where does Traditional Statistics fail?

- **Computer Vision** is an interdisciplinary field that deals with the way algorithms can be made for gaining high-level understanding from digital images or videos.



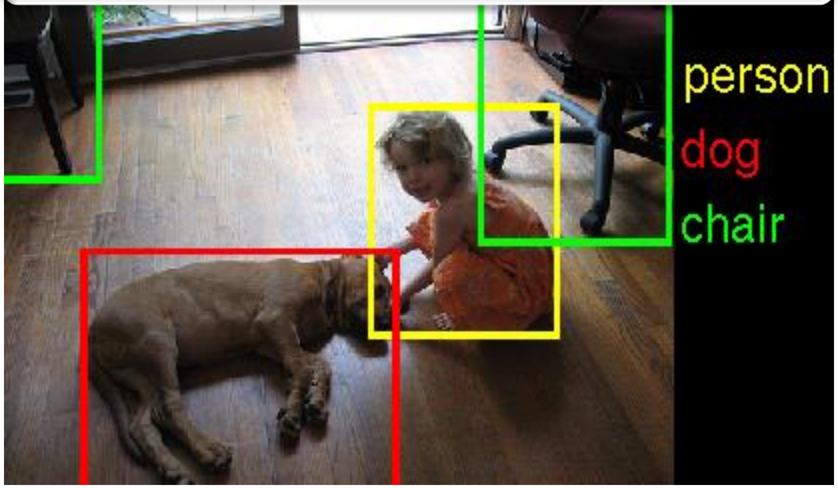
- Statistical methods are not always welcome in computer vision.
- Statistical methods seem not scaling up to the challenges of computer vision problems (*Chellappa, R., 2012*).

Why Does Computer Vision matter so much?

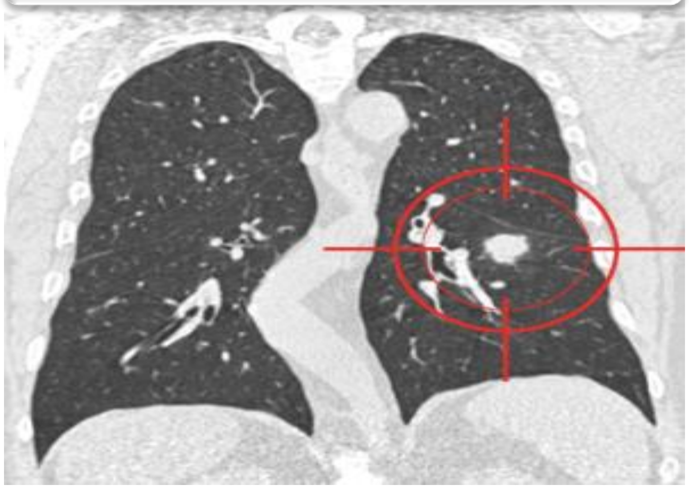
- A new study proves the relationship between Vision capabilities and Intelligence (*Tsukahara et al., 2016*).

Computer Vision needs human-like abilities.

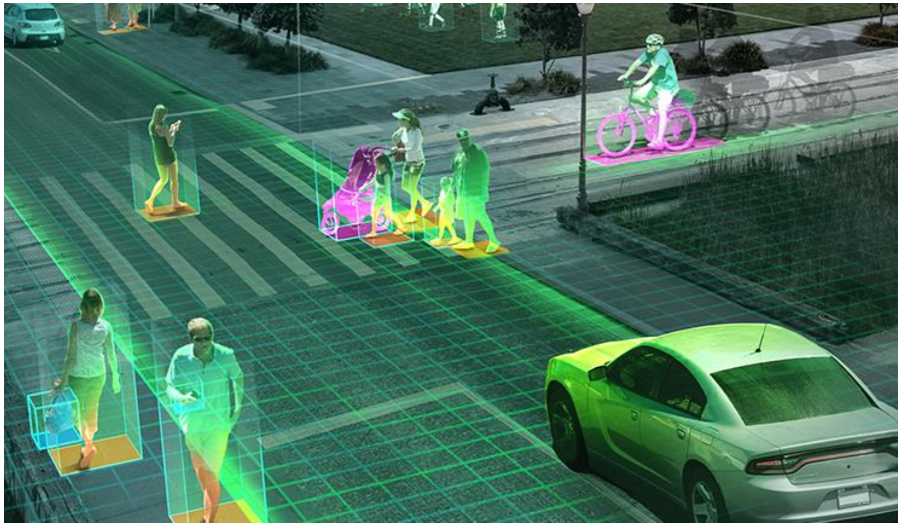
EVERYDAY LIFE



BIOMEDICAL IMAGES



Why Does Computer Vision matter so much?

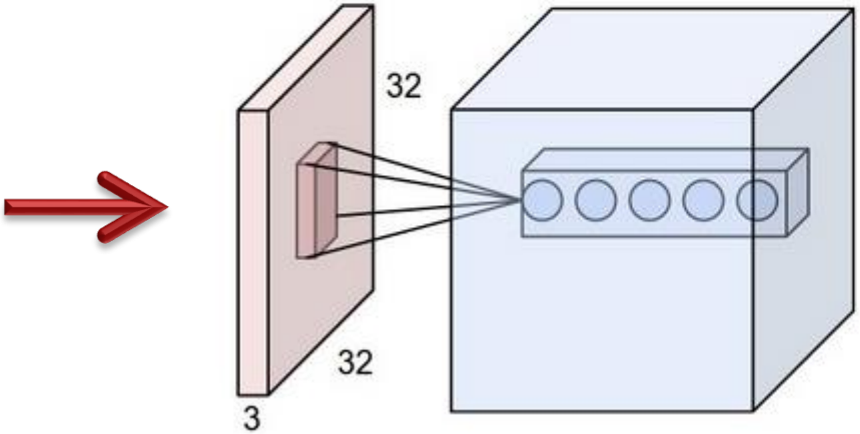
A new generation of machines might accomplish typical human tasks such as recognizing and moving objects, driving cars, cultivating fields, cleaning streets, city garbage collecting, etc.

Convolutional Neural Networks (ConvNets or CNNs)

Convolutional Neural Networks (CNN) are biologically-inspired variants of **MLPs**. We know the visual cortex contains a complex arrangement of cells **(*Hubel, D. and Wiesel, T., 1968*)**. These cells are sensitive to small sub-regions of the visual field, called a *receptive field*. Other layers are: **RELU layer**, **Pool Layer**. Typical CNNs settings are: a) **Number of Kernels (Filters)**, b) **Receptive Field size**, b) **Padding**, c) **Stride.** These parameters are tied by the following equation:

$$(W - F + 2P)/(S + 1)$$

Each neuron in the convolutional layer is connected only to a local region in the input volume spatially. In this case there are 5 neurons along the depth all looking at the same region.



Typical Settings of Convolutional Layers

a) **Number of Kernels (Filters)**

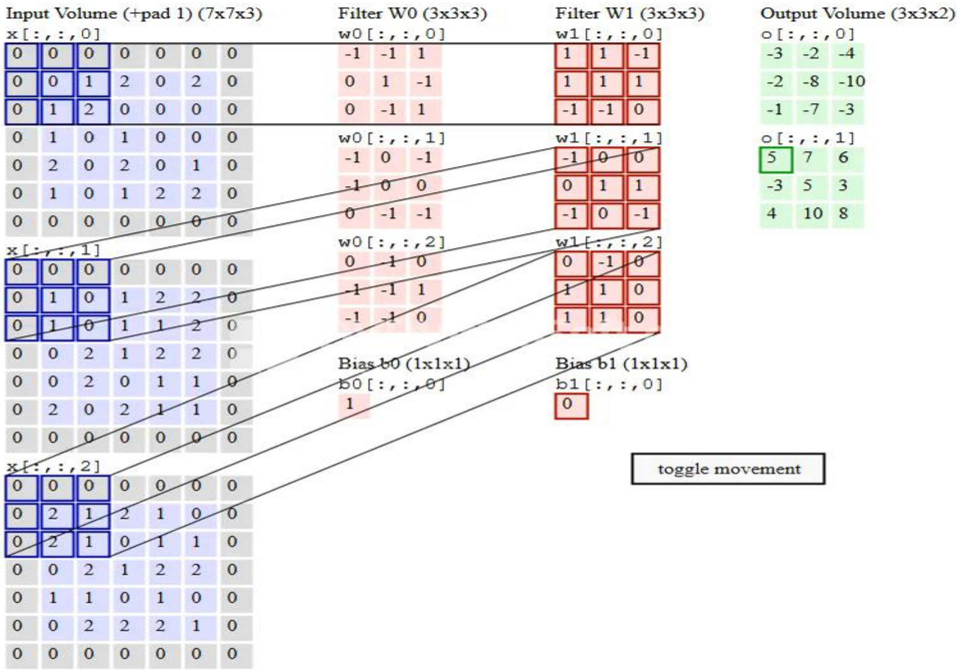
b) **Receptive Field size**

c) **0-Padding**

d) **Stride**

→ **These parameters are tied by the following equation:**

$$(W - F + 2P)/(S + 1)$$

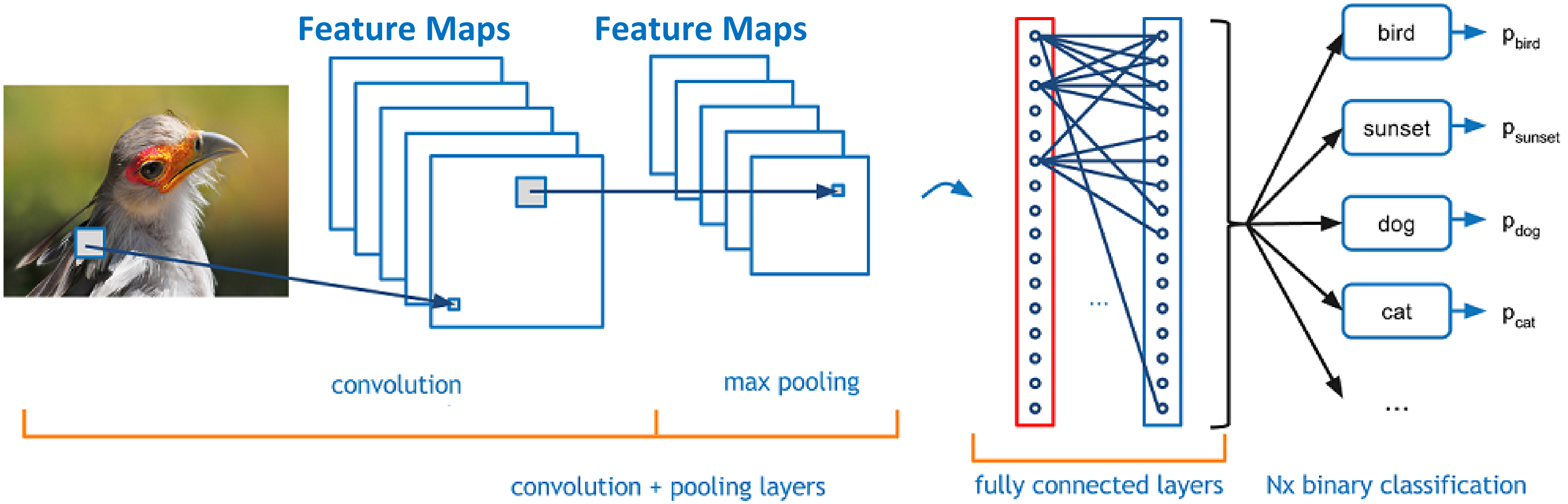


Other layers:

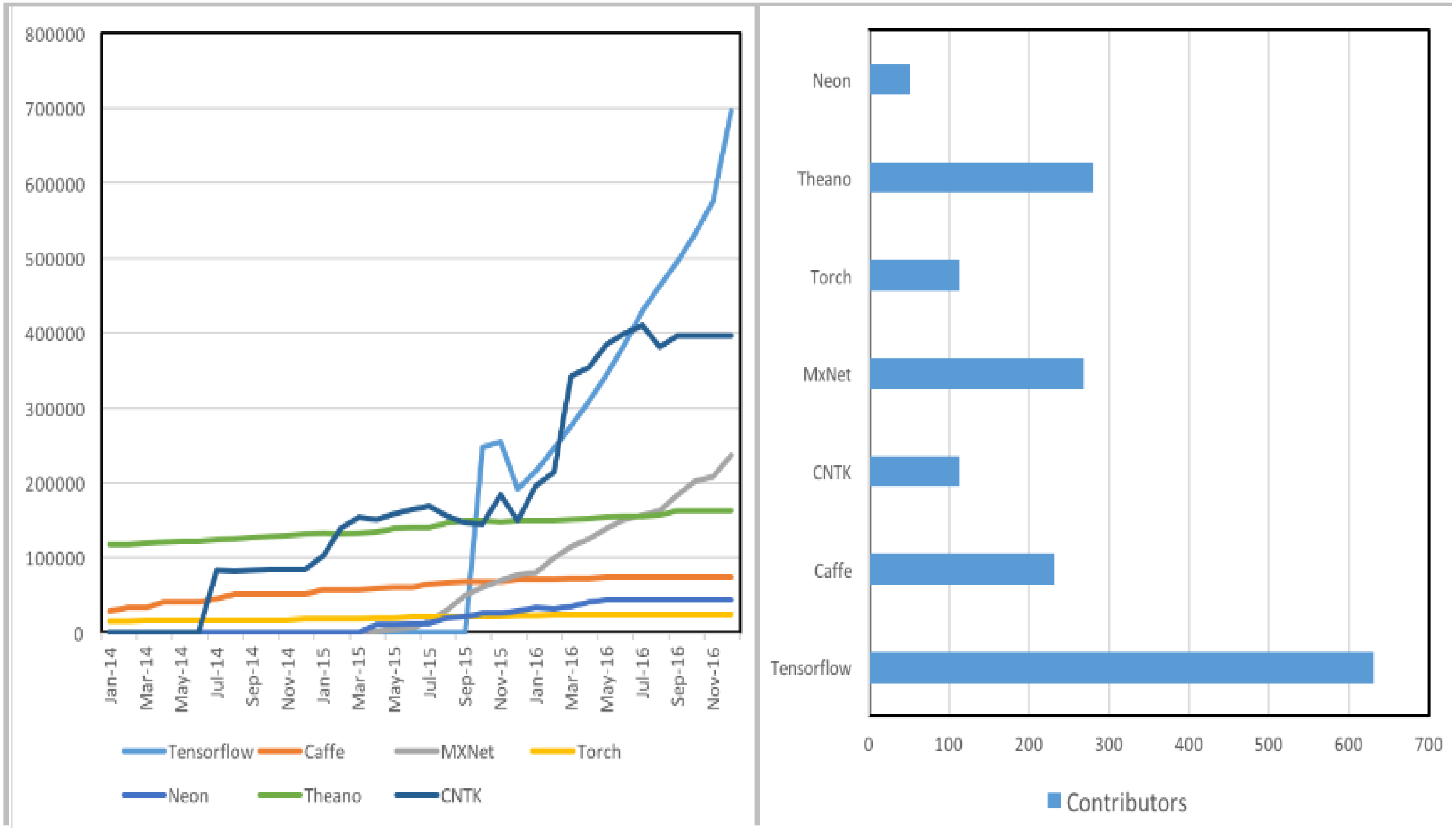
- **Pool Layer**
- **Activation Layer (RELU, TanH, Sigmoid)**
- **Fully Connected Layer**

Convolutional Neural Networks (ConvNets or CNNs)

- **CNNs** were initially devised for **<u>Image Recognition</u>,** nowadays very often reach better-than-human accuracy
- **CNNs** need to be fed with **<u>images</u>,** but since for a machine images are just **<u>numeric matrices</u>…**
- **…**they are increasingly being used in **<u>Natural Language Processing</u>,** e.g. **<u>text classification</u>,** with excellent results



Comparison of GitHub Contributors for Deep Learning Frameworks



Frameworks FOR DEEP LEARNING

	Languages	Tutorials and training materials	CNN modeling capability	RNN modeling capability	Architecture: easy-to-use and modular front end	Speed	Multiple GPU support	Keras compatible
Theano	Python, C++	++	++	++	+	++	+	+
Tensor-Flow	Python	+++	+++	++	+++	++	++	+
Torch	Lua, Python (new)	+	+++	++	++	+++	++	
Caffe	C++	+	++		+	+	+	
MXNet	R, Python, Julia, Scala	++	++	+	++	++	+++	
Neon	Python	+	++	+	+	++	+	
CNTK	C++	+	+	+++	+	++	+	

Frameworks FOR DEEP LEARNING

Keras is an higher-level interface for Theano (which works as backend). Keras displays a more intuitive set of abstractions that make it easy to configure neural networks regardless of the backend scientific computing library.

TensorFlow is an open-source software library for dataflow programming across a range of tasks. It is a symbolic math library, and also used for machine learning applications such as neural networks. It is used for both research and production at Google.

PyTorch is an open-source machine learning library for Python, derived from Torch, used for applications such as natural language processing. It is primarily developed by **Facebook's** artificial-intelligence research group, and **Uber's** "Pyro" software for probabilistic programming is built on it.

Computer Vision Datasets and Competitions

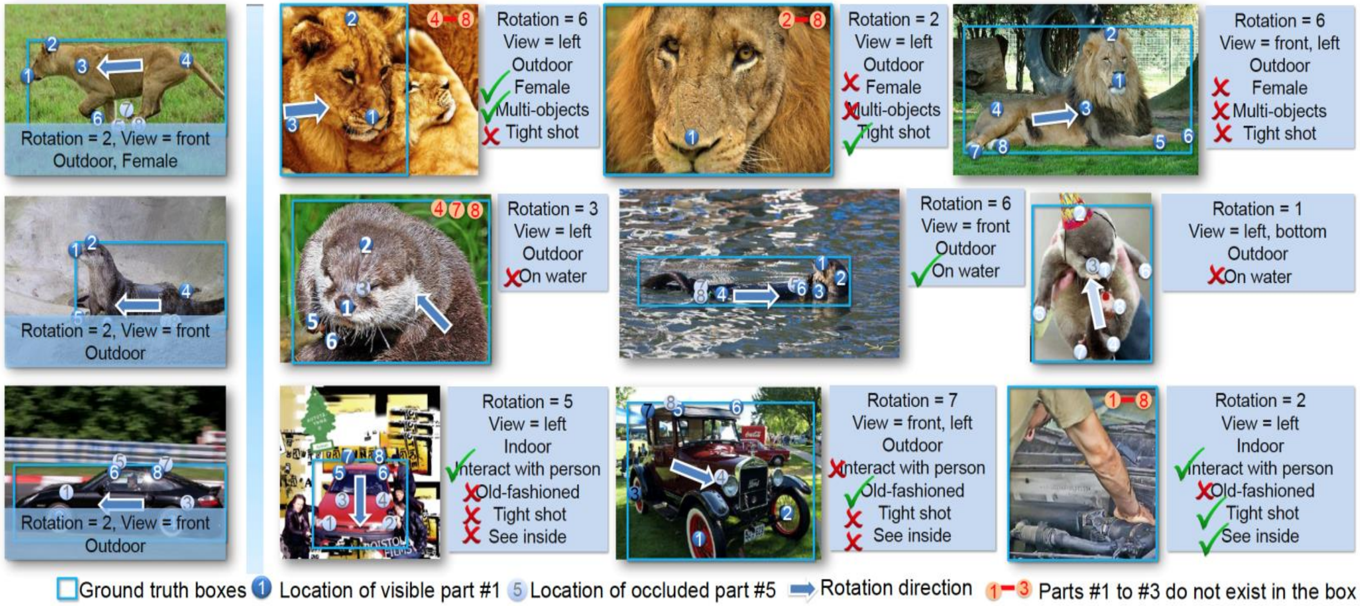
ImageNet: ImageNet is a dataset of over **15** million labeled high-resolution images belonging to roughly **22,000** categories.

- Since **2010** a competition called «ImageNet Large-Scale Visual Recognition Challenge (ILSVRC)» uses a subset of ImageNet with roughly **1000** images in each of **1000** categories.

Train Set: 1.2 million

Validation Set: 50,000

Test set: 150,000



Computer Vision Datasets and Competitions

Kaggle: In 2010, Kaggle was founded as a platform for predictive **modeling** and **analytics competitions** on which companies and researchers post their data.

- Statisticians and data scientists from all over the world compete to produce the best models.
- **Data Science Bowl 2017** was the biggest competition focused on “Lung Cancer Detection”. The competition was founded by **Arnold Foundation** and awarded **$1 million** in prizes (**1st** ranked **$500,000**).

Train Set: around 150 CT labelled scans images per patient from 1200 patients encoded in **DICOM** format.
Stage 1 test set: 190 patients CT scans.
Stage 2 test : 500 patients CT scans.

Grand Challenges in Biomedical Image Analysis: This is a web-site hosting new competitions in the Biomedicine field. Specifically, **LUNA** (**LU**ng **N**odule **A**nalysis) focuses on a large-scale evaluation of automatic nodule detection algorithms.

Train Set: LIDC/IDRI database consisting of 888 CT Scans labelled by 4 expert radiologists.

Computer Vision Datasets and Competitions

Cifar10 Dataset

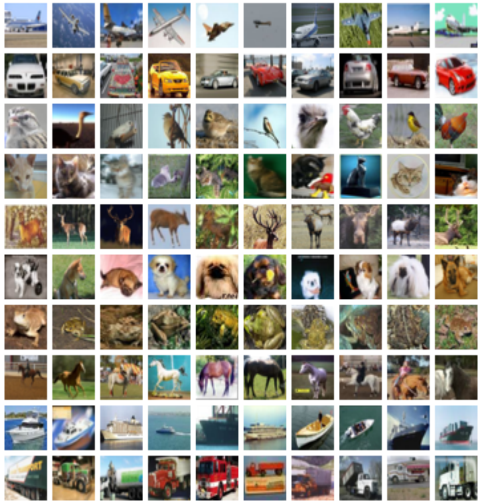
The CIFAR-10 dataset

The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images.

The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class.

Here are the classes in the dataset, as well as 10 random images from each:

airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck



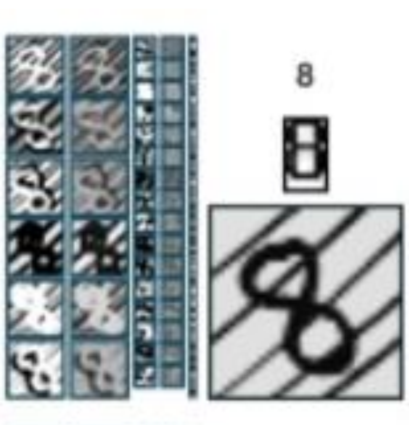
The classes are completely mutually exclusive. There is no overlap between automobiles and trucks. "Automobile" includes sedans, SUVs, things of that sort. "Truck" includes only big trucks. Neither includes pickup trucks.

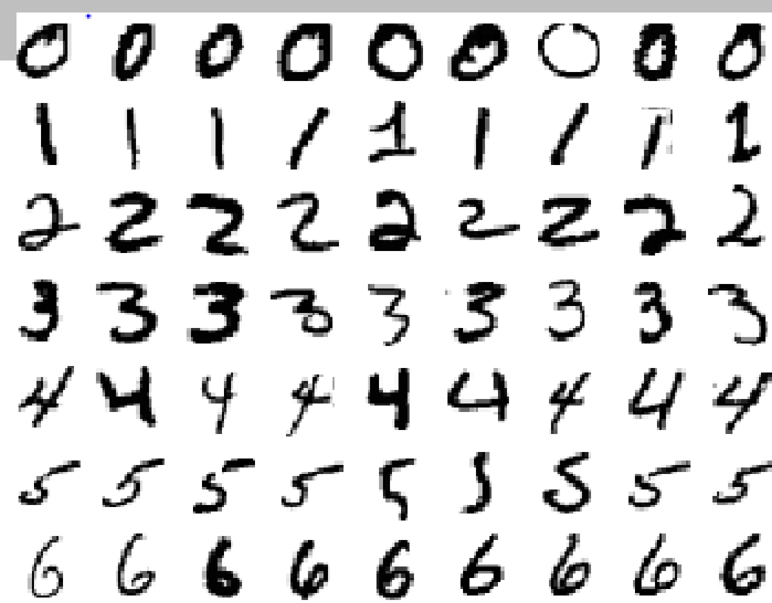
MNIST DATASET

MNIST database is handwritten digits composed of 60.000 pattern.

- 60.000 training set
- 10,000 test set

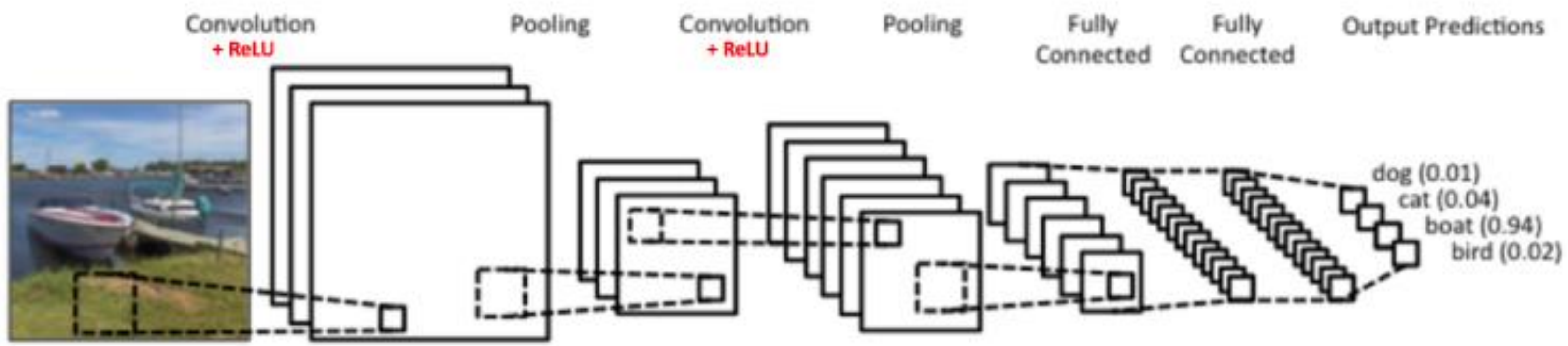
LeNet has been applied to this dataset with accuracy of 0.95%.





LeNet – Convolutional Neural Network

LeNet was one of the very first convolutional neural networks which helped to propel the field of Deep Learning. This pioneering work by Yann LeCun was named LeNet5 after many previous successful iterations since the year 1988.



LeNet in Keras

```python
'''
Created on 22/03/2017

@author: Francesco Pugliese
'''

from keras.models import Sequential
from keras.layers.convolutional import Convolution2D
from keras.layers.convolutional import MaxPooling2D
from keras.layers.core import Activation
from keras.layers.core import Flatten
from keras.layers.core import Dense

class LeNet:
    @staticmethod
    def build(width, height, depth, classes, summary, weightsPath=None):
        # initialize the model
        model = Sequential()
        # first set of CONV => RELU => POOL
        model.add(Convolution2D(20, 5, 5, border_mode="same", input_shape=(depth, height, width)))
        model.add(Activation("relu"))
        model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2)))
```

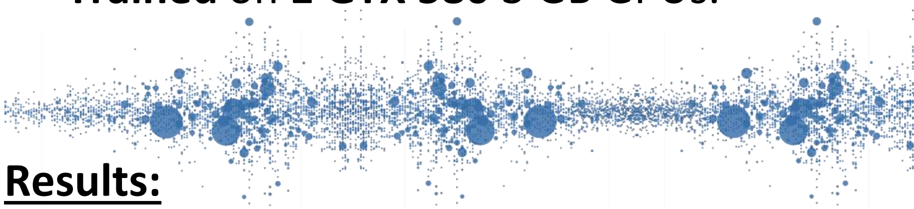
LeNet in Keras

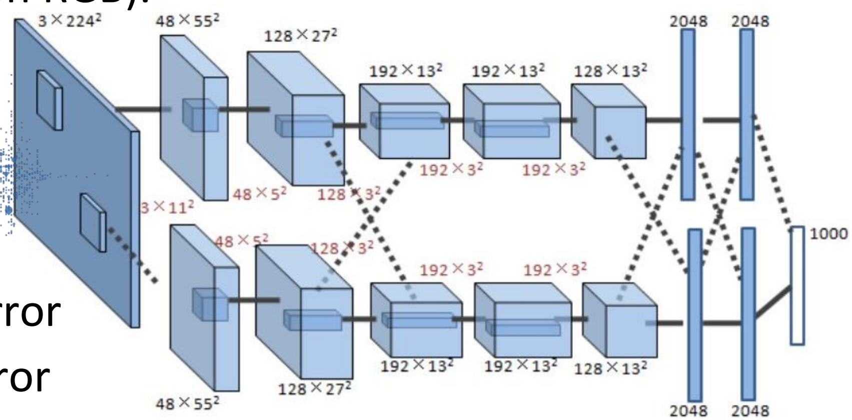
```
# second set of CONV => RELU => POOL
model.add(Convolution2D(50, 5, 5, border_mode="same"))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2)))

#set of FC => RELU Layers
model.add(Flatten())
model.add(Dense(500))
model.add(Activation("relu"))


#softmax classifier
model.add(Dense(classes))
model.add(Activation("softmax"))

if summary==True:
    model.summary()

#if a weights path is supplied (indicating that the model was pre-trained), then load the weights
if weightsPath is not None:
    model.load_wights(weightsPath)

return model
```

AlexNet

<u>Critical Feautures</u> (*Krizhevsky, A. et al, 2012*)

- **8 trainable layers:** 5 convolutional layers and 3 fully connected layers.
- **Max pooling layers** after 1st, 2nd and 5th layer.
- **Rectified Linear Units (ReLUs) (***Nair, V., & Hinton, G. E. 2010*).
- **Local Response Normalization.**
- **60 millions parameters, 650 thousands neurons**.
- **Regularizations**: Dropout (prob 0.5 in the first 2 fc layers, Data Augmentation (translactions, horizontal reflections, PCA on RGB).
- **Trained** on 2 **GTX 580** 3 GB GPUs.





<u>**Results:**</u>

- **1 CNNs: 40.7%** Top-1 Error, **18.2%** Top-5 Error
- **5 CNNs: 38.1%** Top-1 Error,**16.4%** Top-5 Error
- **SIFT+FVs: 26.2%** Top-5 Error (*Sánchez, J., et al., 2013*).

AlexNet in Keras

```
def get_alexnet(input_shape,nb_classes,mean_flag):
        # code adapted from https://github.com/heuritech/convnets-keras

        inputs = Input(shape=input_shape)

        if mean_flag:
                mean_subtraction = Lambda(mean_subtract, name='mean_subtraction')(inputs)
                conv_1 = Convolution2D(96, 11, 11,subsample=(4,4),activation='relu',
                                   name='conv_1', init='he_normal')(mean_subtraction)
        else:
                conv_1 = Convolution2D(96, 11, 11,subsample=(4,4),activation='relu',
                                   name='conv_1', init='he_normal')(inputs)

        conv_2 = MaxPooling2D((3, 3), strides=(2,2))(conv_1)
        conv_2 = crosschannelnormalization(name="convpool_1")(conv_2)
        conv_2 = ZeroPadding2D((2,2))(conv_2)
        conv_2 = merge([
            Convolution2D(128,5,5,activation="relu",init='he_normal', name='conv_2_'+str(i+1))(
                splittensor(ratio_split=2,id_split=i)(conv_2)
            ) for i in range(2)], mode='concat',concat_axis=1,name="conv_2")

        conv_3 = MaxPooling2D((3, 3), strides=(2, 2))(conv_2)
        conv_3 = crosschannelnormalization()(conv_3)
        conv_3 = ZeroPadding2D((1,1))(conv_3)
        conv_3 = Convolution2D(384,3,3,activation='relu',name='conv_3',init='he_normal')(conv_3)
```

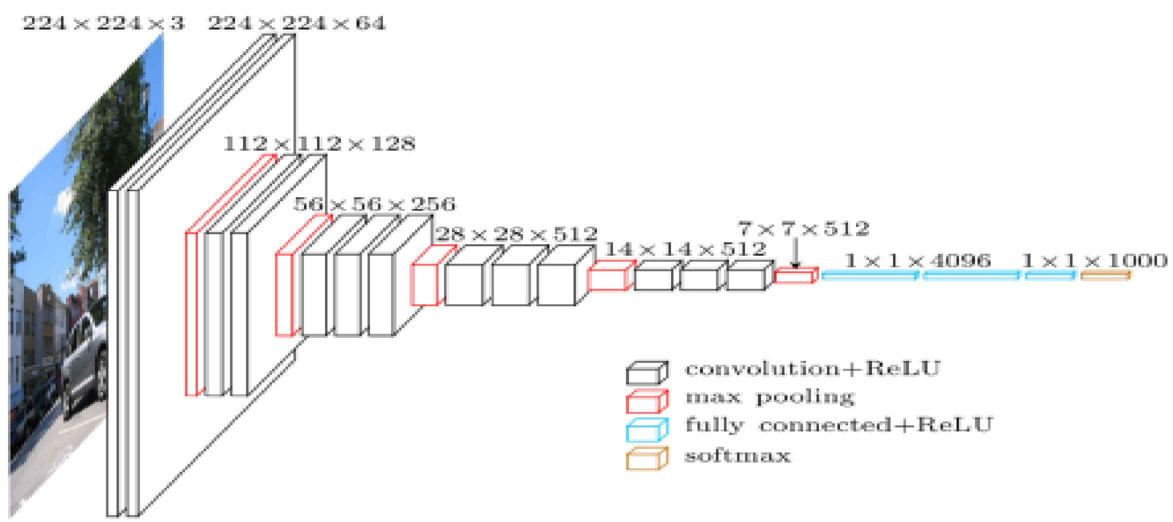
AlexNet in Keras

```
conv_4 = ZeroPadding2D((1,1))(conv_3)
conv_4 = merge([
    Convolution2D(192,3,3,activation="relu", init='he_normal', name='conv_4_'+str(i+1))(
        splittensor(ratio_split=2,id_split=i)(conv_4)
    ) for i in range(2)], mode='concat',concat_axis=1,name="conv_4")

conv_5 = ZeroPadding2D((1,1))(conv_4)
conv_5 = merge([
    Convolution2D(128,3,3,activation="relu",init='he_normal', name='conv_5_'+str(i+1))(
        splittensor(ratio_split=2,id_split=i)(conv_5)
    ) for i in range(2)], mode='concat',concat_axis=1,name="conv_5")

dense_1 = MaxPooling2D((3, 3), strides=(2,2),name="convpool_5")(conv_5)

dense_1 = Flatten(name="flatten")(dense_1)
dense_1 = Dense(4096, activation='relu',name='dense_1',init='he_normal')(dense_1)
dense_2 = Dropout(0.5)(dense_1)
dense_2 = Dense(4096, activation='relu',name='dense_2',init='he_normal')(dense_2)
dense_3 = Dropout(0.5)(dense_2)
dense_3 = Dense(nb_classes,name='dense_3_new',init='he_normal')(dense_3)

prediction = Activation("softmax",name="softmax")(dense_3)

alexnet = Model(input=inputs, output=prediction)

return alexnet
```

VGG-Net – University of Oxford

__Critical Feautures (__**Simonyan, K., & Zisserman, A., 2014):**

- **Kernels with small receptive fields:** 3x3 which is the smallest size to capture the notion of left/right up/down, center. It is easy to see that a stack of two 3×3 conv. layers (without spatial pooling in between) has an effective receptive field of 5×5, and so on.
- Small size **Receptive Field** is a way to increase the nonlinearity of the decision function fields of the conv. layers.
- **Increasing depth architectures: VGG-16** (2xConv3-64, 2xConv3-128, 3xConv3-256, 6xConv3-512, 3xFC), **VGG-19** (same as VGG-16 but with 8xConv3-512).
- **Upside:** less complex topology, outperforms GoogleNet on single-network classification accuracy
- **Downside: 138 millions parameters for VGG-16 !**



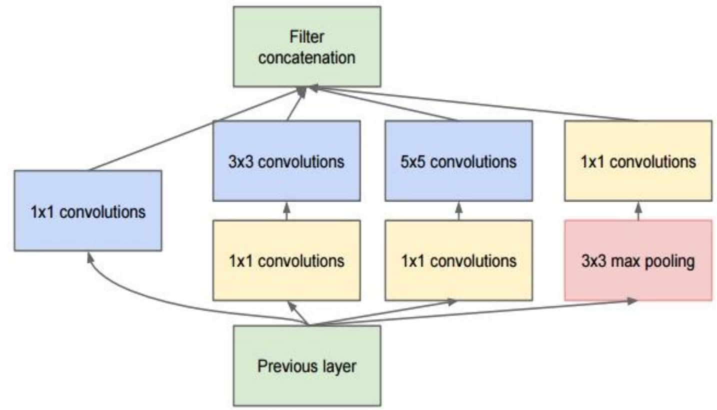
__Results:__

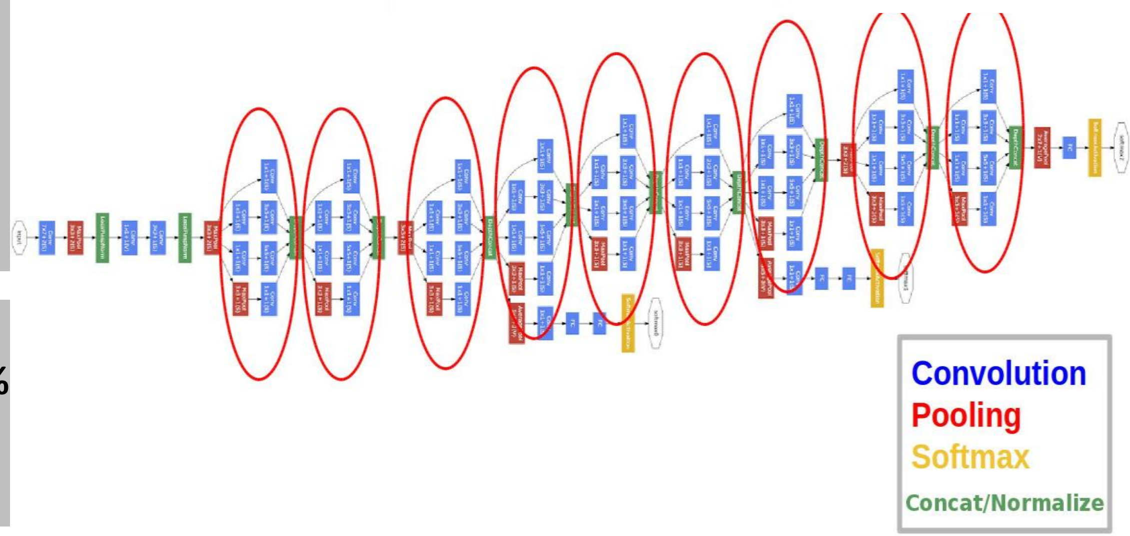
- **Multi ConvNet model :** (D/[256;512]/256,384,512), (E/[256;512]/256,384,512), multi-crop & dense eval: **23.7%** Top-1 Error, **6.8%** Top-5 Error.

GoogleNet – Google

Critical Feautures (Szegedy, C., et al., 2015):

- **Computationally Effective Deep architecture:** 22 layer
- **Why the name inception, you ask?** Because the module represents a *network within a network*. If you don't get the reference, go watch Christopher Nolan's **“INCEPTION”**, computer scientists are hilarious.
- Inception: it isbasically the parallel combination of 1×1 3×3, and 5×5 convolutional filters.
- **Bottleneck layer:** The great insight of the inception module is the use of 1×1 conv-olutional blocks (NiN) to reduce the number of features before the expensive parallel blocks.
- **Upside: 4 millions parameters!**
- **Downside: Not scalable**!





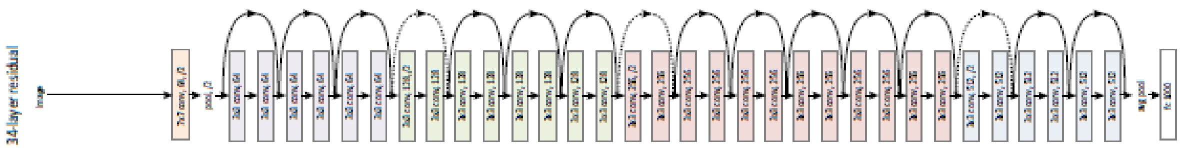
Results:

- **7 Models Ensemble : 6.67%** Top-5 Error.

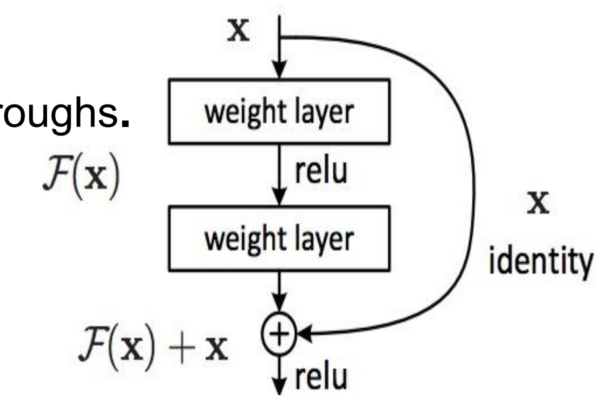
Wide Res Net – Microsoft

<u>Critical Feautures</u> (*He, K., et al., 2016*) :

- **Degradation Problem:** Stacking more and more layers **IS NOT** better. With the <u>network</u> depth increasing, accuracy gets saturated and then degrades rapidly! It's an issue of "solvers".



- **Solves the "Degradation problem":** by fitting a residual mapping which is easier to optimize.
- **Shortcut connections**
- **Very deep architecture:** up to 1202 layers with WideResnet with only **19.4 million parameters**!
- **Upside:** Increasing accuracy with more depth
- **Downside:** They don't consider other architectures breakthroughs**.**

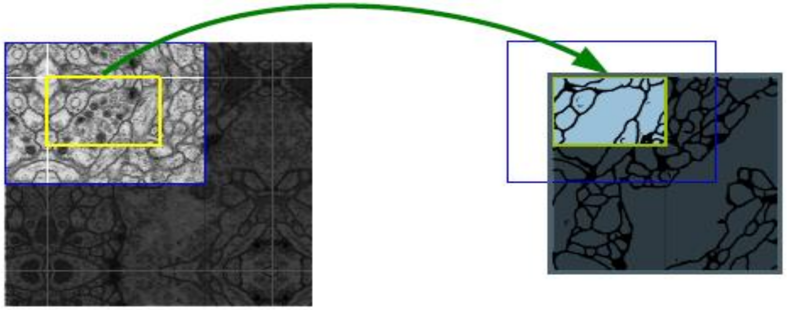


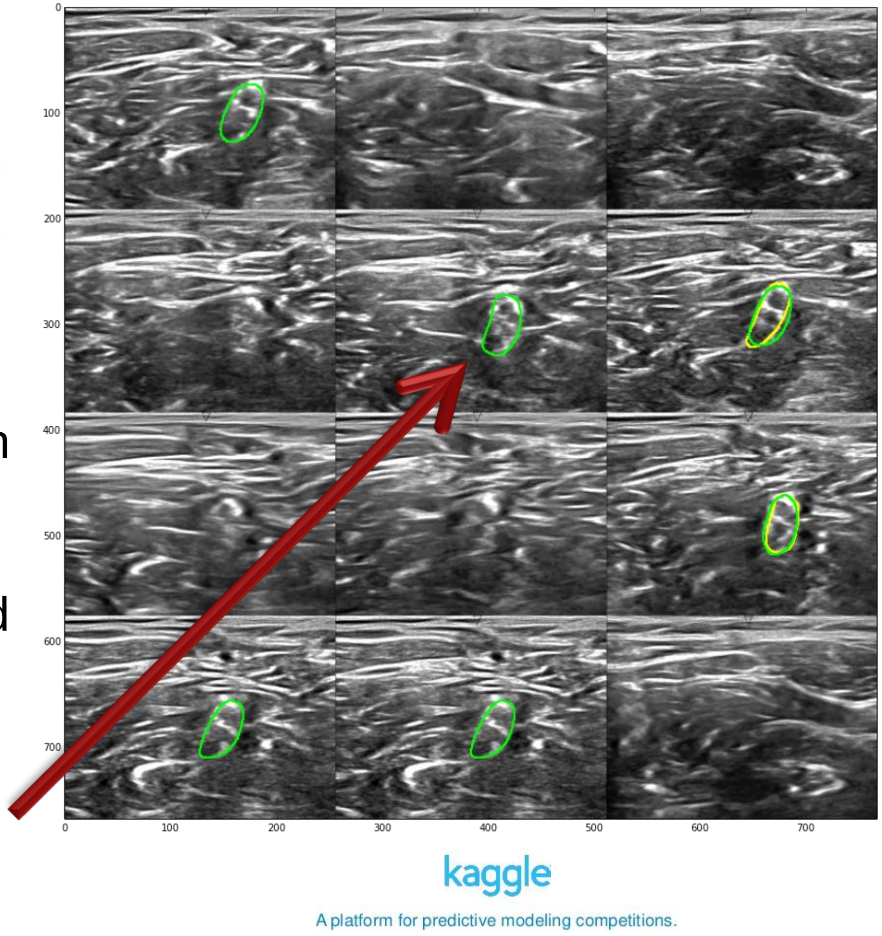
<u>Results:</u>

ResNet : 3.57% Top-5 Error.

CNNs show superhuman abilities at Image Recognition!
5% Human estimated Top-5 error. (*Johnson, R. C., 2015*)

Segmentation problem: alias with biomedical images CNNs fail

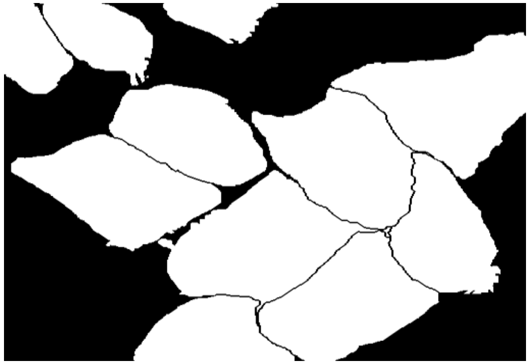




Problems:

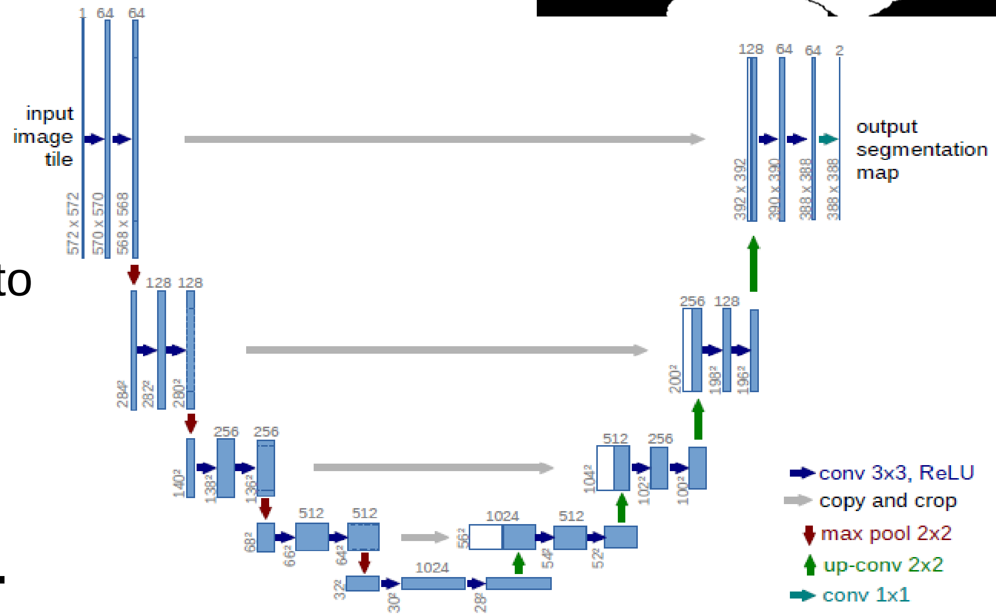
- **Feature extraction:** In biomedicine feature extraction is not as easy as in an Imagenet competition with general images. A previous **Image Preprocessing** is needed. This is called **Segmentation**.
- On Kaggle website there are whole competitions just regarding Segmentation. One of these was called **«Ultrasound Nerve Segmentation».**

U-NET (Fully connected CNN)

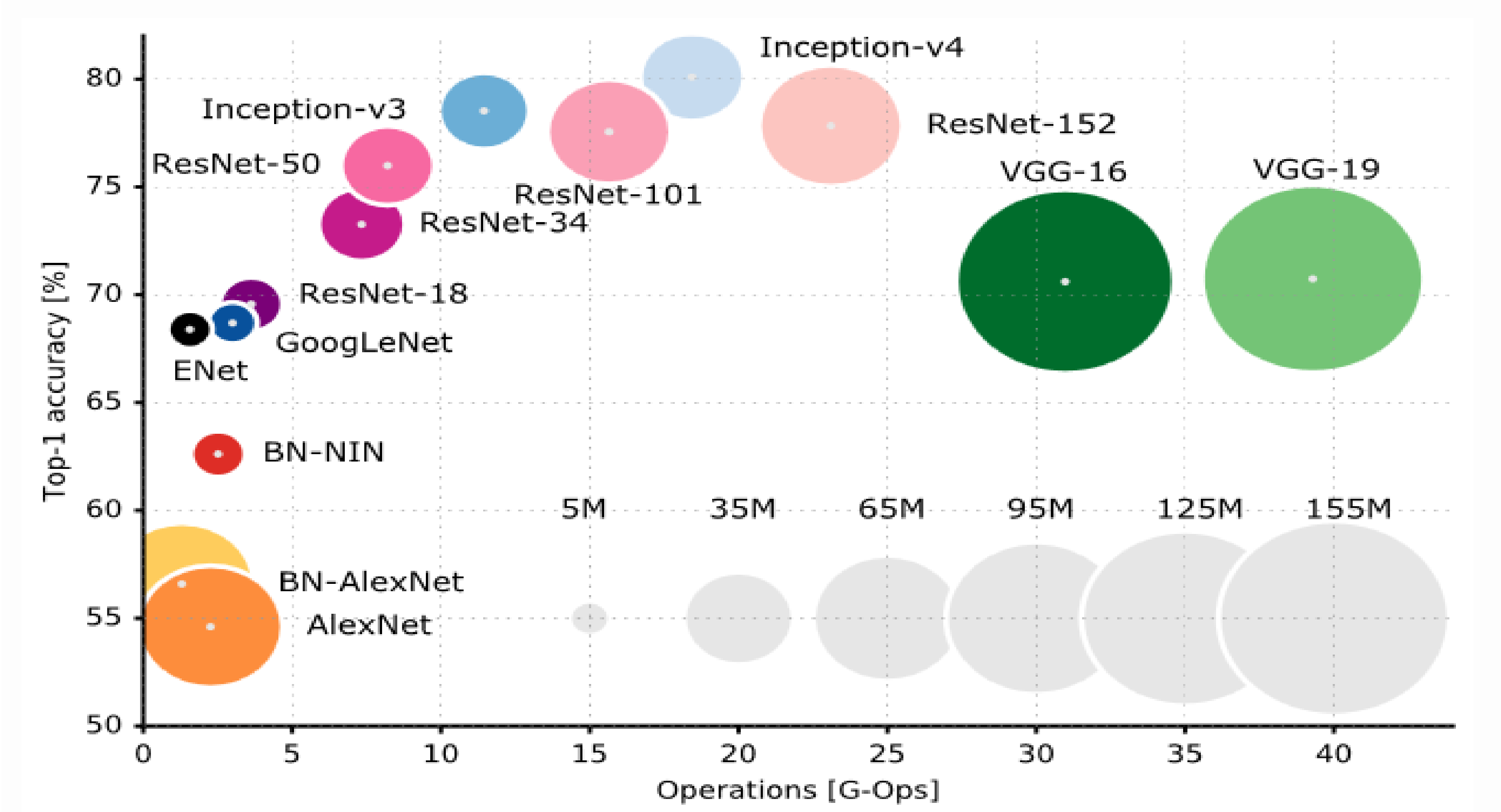


Critical Feautures (*Ronneberger, O., et al., 2015*):

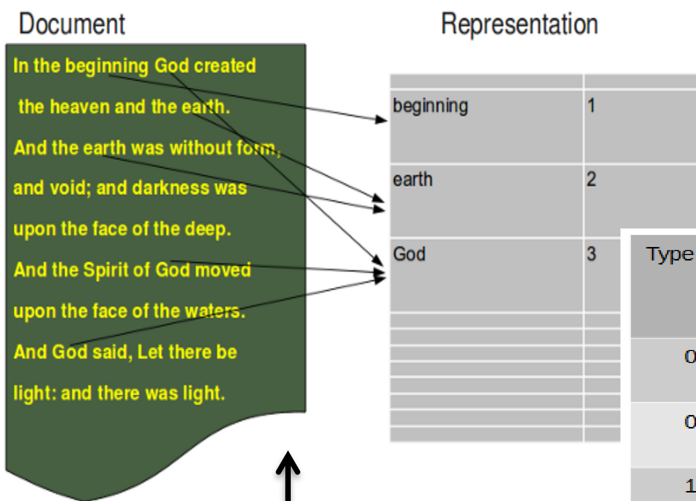
- **U-NET** can be trained end-to-end from very **few images** and outperforms the prior best methods.
- It consists of a **contracting path (left side)** to capture context and an symmetric **expansive path (right side**) enabling precise localization.
- **Upsampling part (repeating rows and cols)** has a large number of feature channels which allow the network to propagate context information to higher resolution layers.
- **Spatial Dropout:** feature maps dropout.
- **Upside: Small training set.**
- **Downside: Risk of overfitting.**



Neural Network Architectures

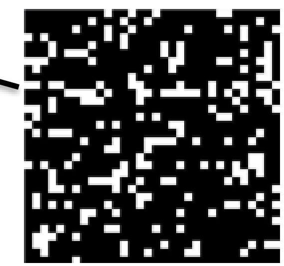


Bag of Words and Term Document Matrix



From each web-site: Bag-Of-Words

Type	relax	carrello	prenotare	shop	wishlist
0	0	0	0	0	0
0	0	0	0	0	0
1	0	0	0	0	0
1	0	0	1	0	0
0	0	0	0	0	0
0	0	1	0	1	1
1	1	1	1	0	0

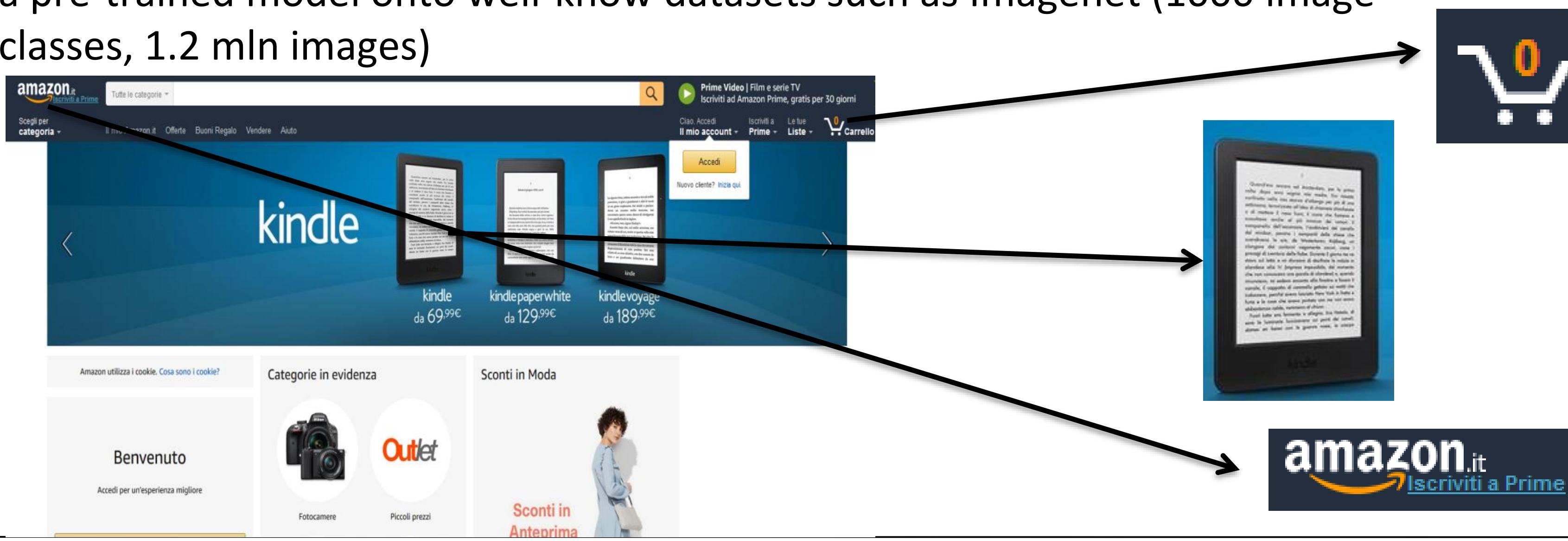


From all web-sites' bag-of-words: TDM

From each TDM row: 32x32 squared image encoding

Web-Site classification by Images Approach

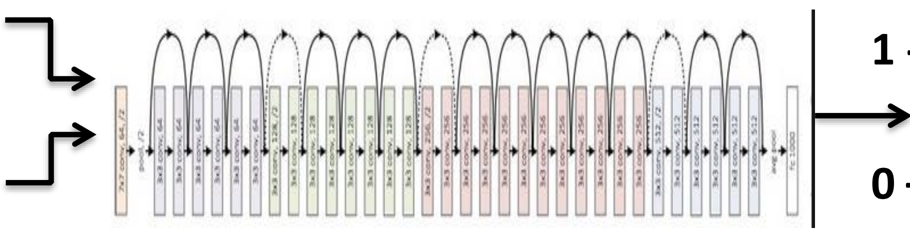
- According to the **False Positive Reduction** technique we exploit the inner images segmentation of a Web-site in order to train an evolved ConvNet (ResNet) model onto the single websites images segments.
- **ConvNet** is trained in **“Transfer Learning”** mode, which means taking advantage of a pre-trained model onto well-know datasets such as Imagenet (1000 image classes, 1.2 mln images)



Web-Site classification by Images Approach

Positive Set (E-commerce)

Residual Neural Network



1 - E-commerce

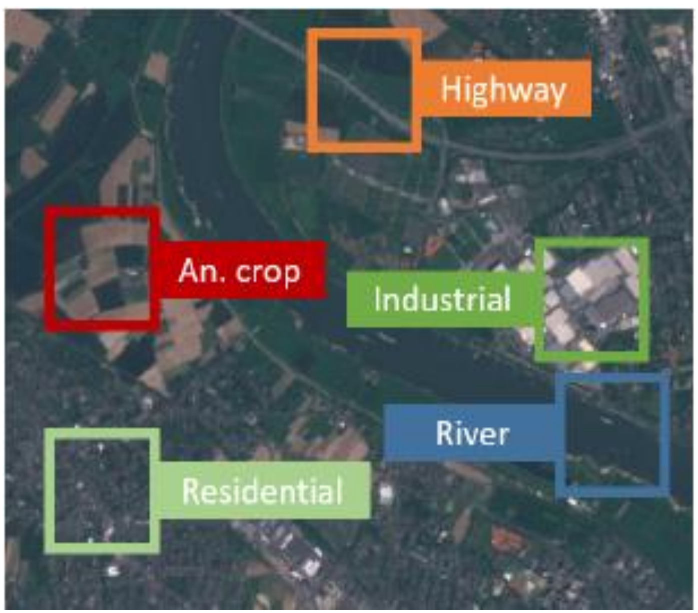
0 – Non E-commerce

Negative Set (Non E-commerce)

- In test stage, on a **Test Set,** Web-site images are still segmented and the label of the image with higher probability is assigned to the web-site itself.

Automatic Extraction of Statistics from Satellite Imagery: Land Use and Land Cover Classification

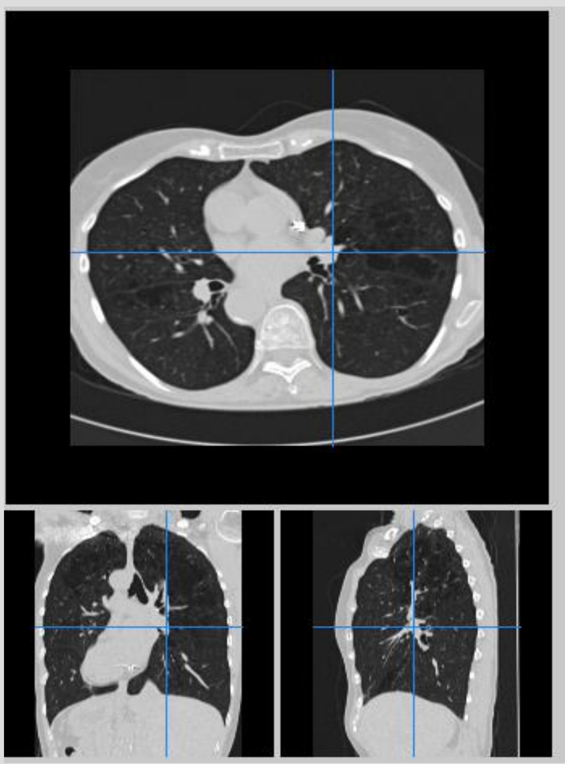
Nowadays, more and more public and up-to-dated **satellite image** data for Earth observation are available.



However, to fully utilize this data, to automatically extract statistics, satellite images must be processed and transformed into structured semantics.

Lung Cancer Detection

Training Set (DICOM)



Preprocessing

DICOM to PNG Conversion

.png Images

Denoising Filters

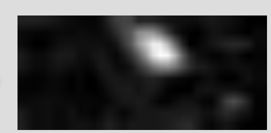
.png Images

SEGMENTATION (U-NET)

(x,y) coordinates

Nodules Distance Merging & Z Merging

(Connected Graphs by Nodules Centre of Mass)



Nodules Images

Classification Model

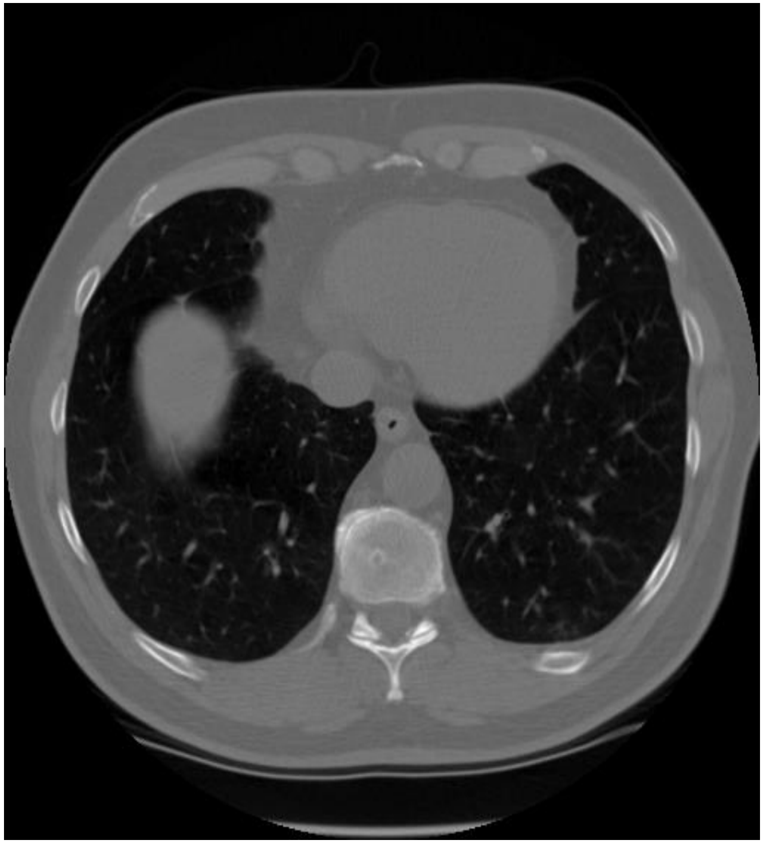
FALSE POSITIVE REDUCTION

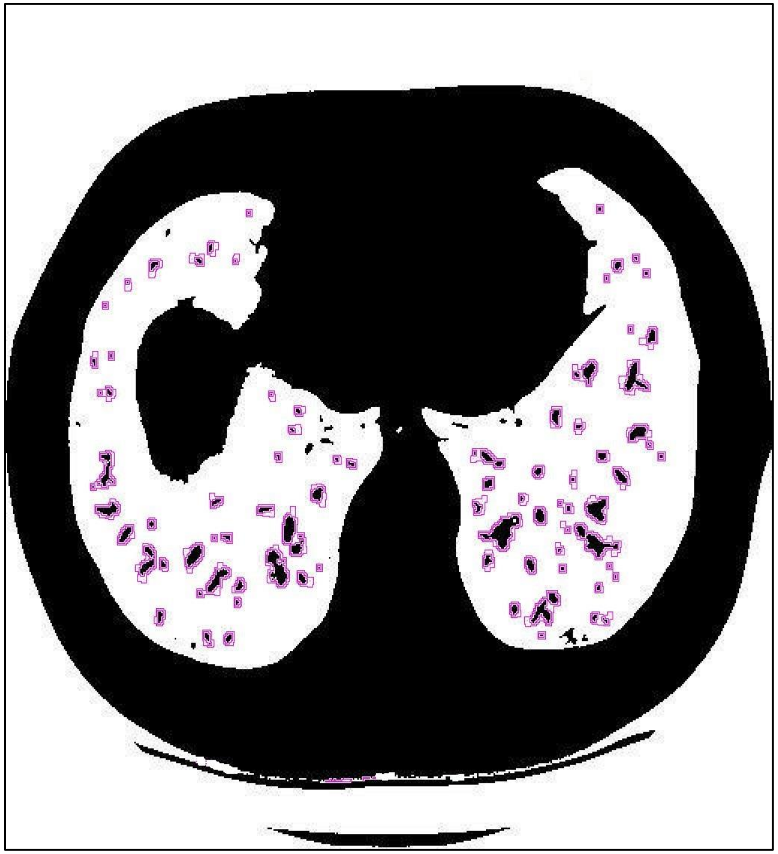
3 Wide Res Net XY, XZ, YZ planes Models

Cancer **Non-Cancer**

Segmentation

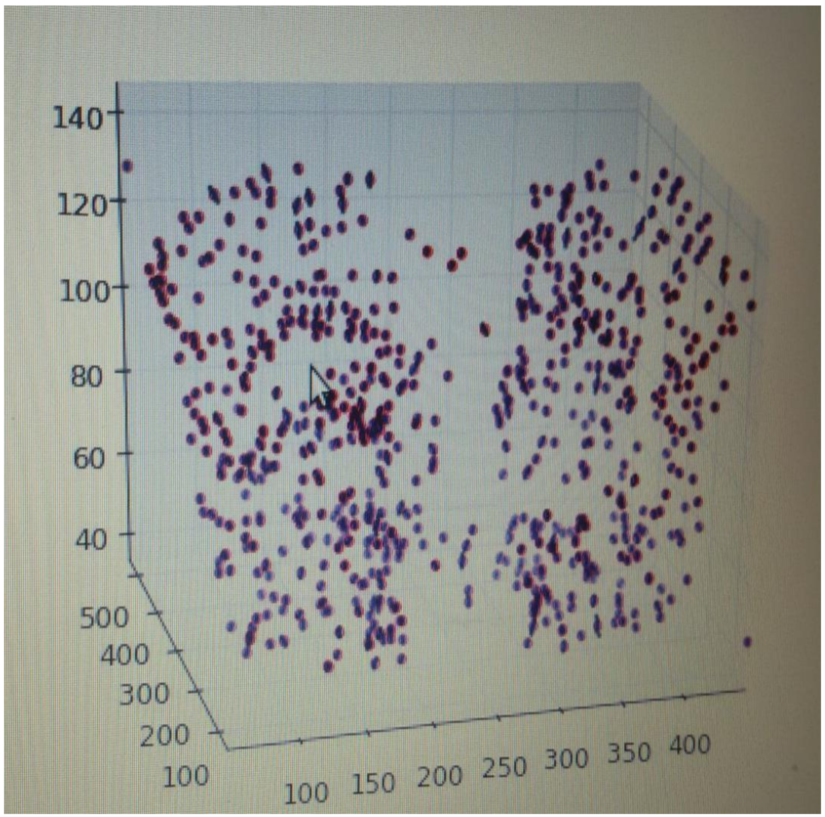
- Segmentation algorithm yields the coordinates (X,Y) of the nodules centers which enable the distance merging algorithm to extract nodules from directly from input CT-Scans.



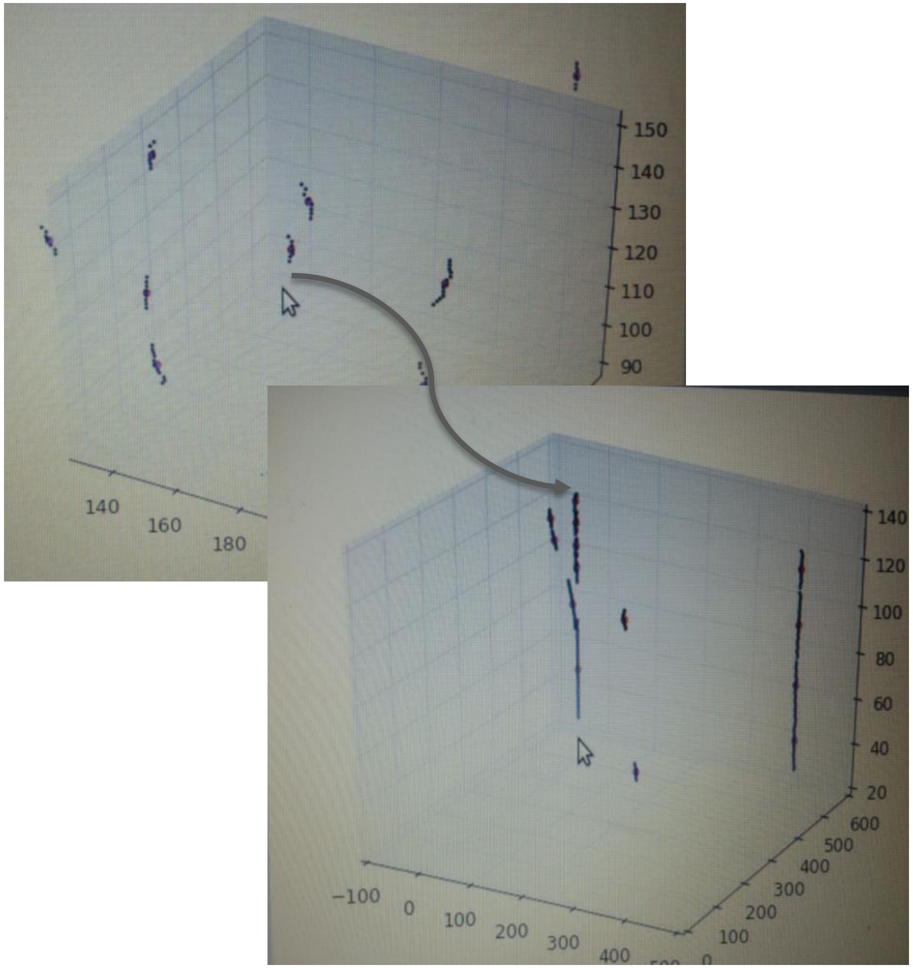


Distance Merging and Z-Merging

Distance Merging

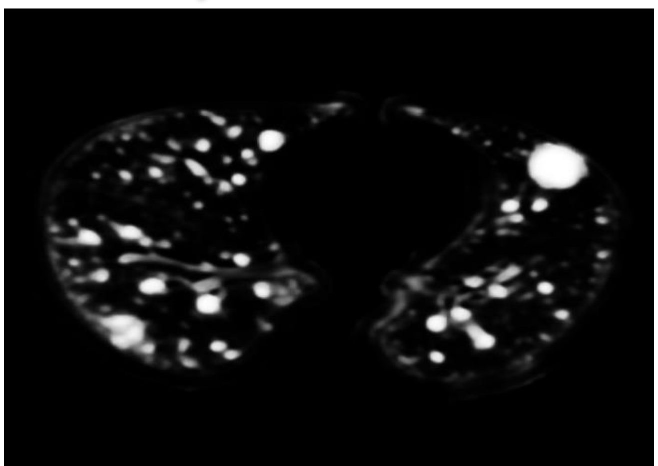


Z-Merging

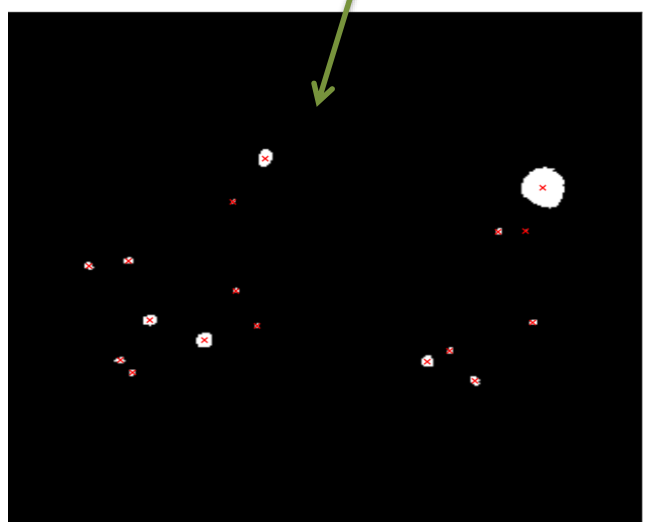


Lung Cancer Classification

Candidate Nodule Selection via UNET



Dilation, Erosion, Nodules Distance Merging



False Positive Reduction via WideResNet

Cancer / Non cancer

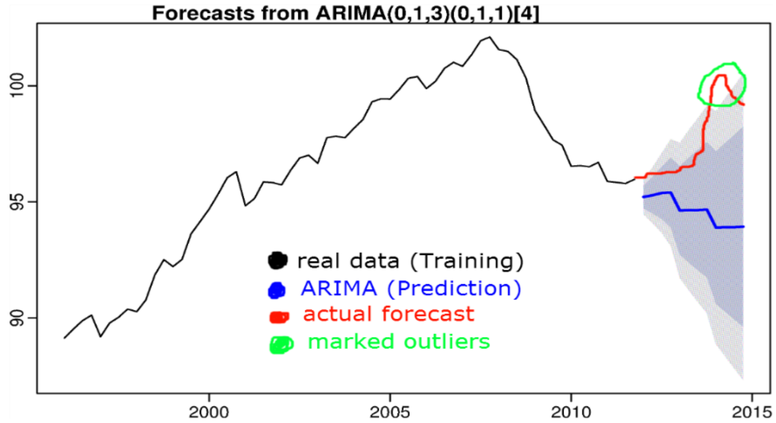
Deep Learning for Time-Series Prediction

- The application of **Deep Learning approaches** to time-series prediction has received a great deal of attention from both **entrepreneurs** and **researchers**. Results show that deep learning models outperform other statistical models in predictive accuracy **(Bao, et al., 2017).**

The application of classic time series models, such as **Auto Regressive Integrated Moving Average (ARIMA)**, usually requires strict assumptions regarding the distributions and stationarity of time series. For complex, non-stationary and noisy time-series it is necessary for one to know the properties of the time series before the application of classic time series models **(Bodyanskiy and Popov, 2006).** Otherwise, the forecasting effort would be ineffective.

$$X_t - \alpha_1 X_{t-1} - \cdots - \alpha_{p'} X_{t-p'} = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q},$$

$$\left(1 - \sum_{i=1}^{p'} \alpha_i L^i\right) X_t = \left(1 + \sum_{i=1}^{q} \theta_i L^i\right) \varepsilon_t$$

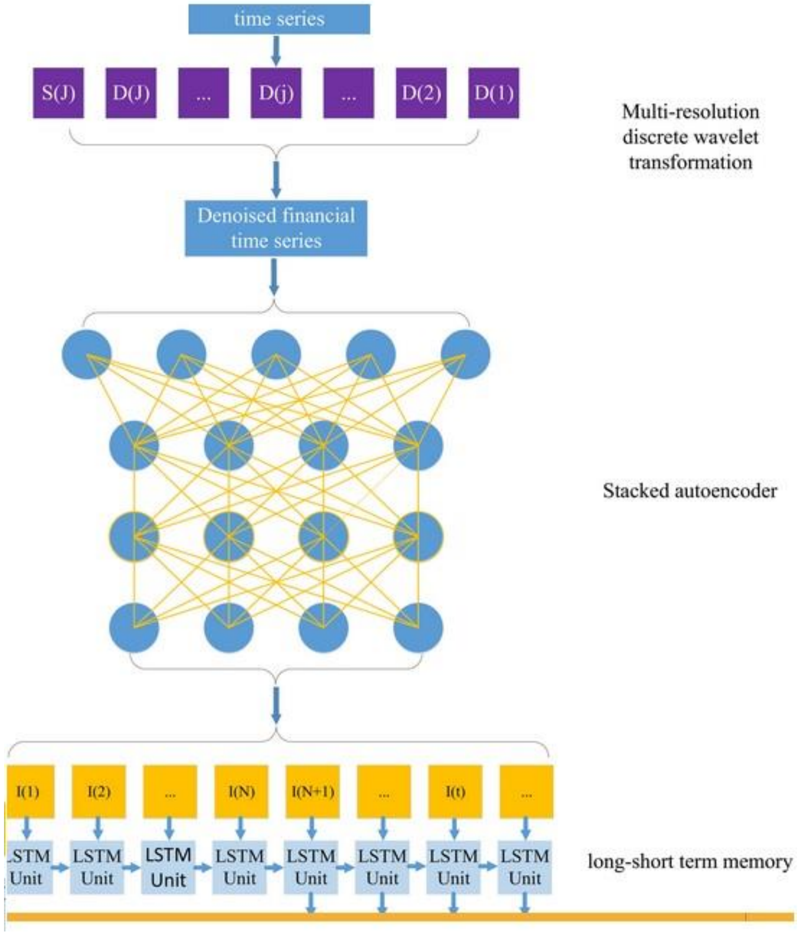


Advantages of Artificial Neural Networks (ANNs) in Time-Series Prediction

- However, by using ANNs, a priori analysis as ANNs do not require prior knowledge of the time series structure because of their black-box properties **(Nourani, et al., 2009)**.

- Also, the impact of the stationarity of time series on the prediction power of ANNs is quite small. It is feasible to relax the stationarity condition to non-stationary time series when applying ANNs to predictions **(Kim, et al., 2004)**.

- ANNs allow **multivariate time-series forecasting** whereas classical linear methods can be difficult to adapt to multivariate or multiple input forecasting problems.

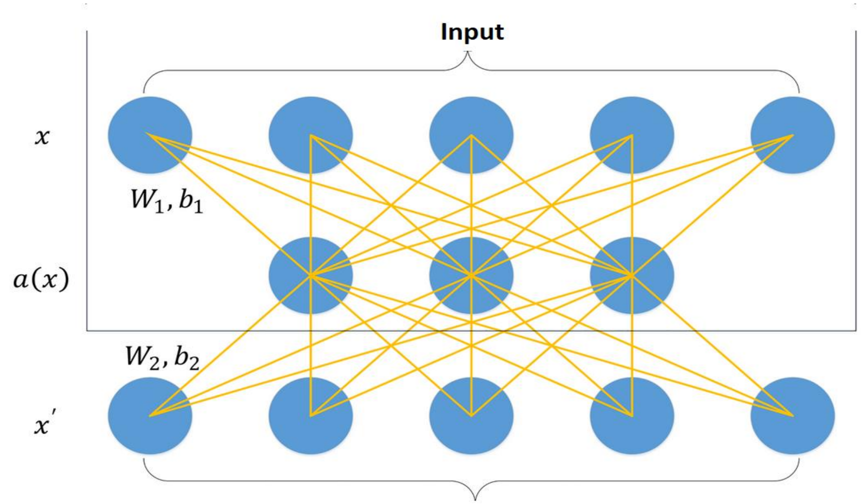


Stacked Auto-encoders (SAEs)

- According to recent studies, better approximation to nonlinear functions can be generated by stacked deep learning models than those models with a more shallow structure.

- A Single layer **Auto-Encoder (AE)** is a three-layer neural network. The first layer and the third layer are the input layer and the reconstruction layer with *k* units, respectively. The second layer is the hidden layer with *n* units, which is designed to generate the deep feature for this single layer AE.

- The aim of training the single layer AEE is to minimize the error between thee input vector and the reconstruction vector by **gradient descent**.



$$a(x) = f(\mathbf{W}_1 x + b_1)$$

$$x' = f(\mathbf{W}_2 a(x) + b_2)$$

Stacked Auto-encoders (SAEs)

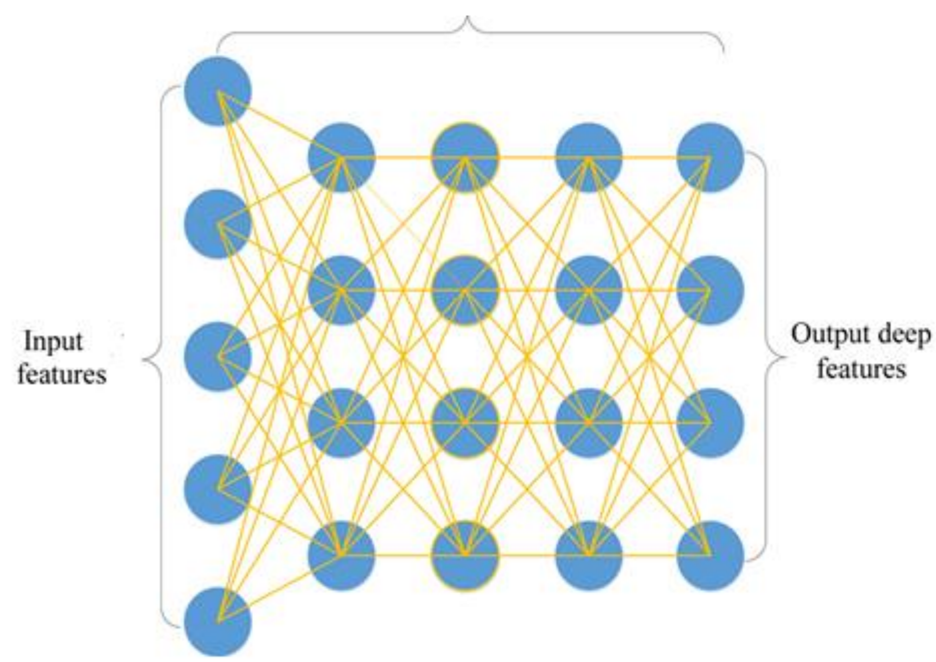
- Stacked auto-encoders (SAEs) are constructed by stacking a sequence of single-layer AEs layer by layer **(Bengio Y, et. Al. 2007)**.

- After training the first single-layer auto-encoder, the reconstruction layer of the first single layer auto-encoder is removed (included weights and biases), and the **hidden layer** is reserved as the input layer of the second single-layer auto-encoder.

- **Depth** plays an important role in **SAE** because it determines qualities like invariance and abstraction of the extracted feature.

- **Wavelet Transform (WT)** can be applied as input to SAEs to handle data particularly non-stationary (**Ramsey, (1999)** .

4 Auto-Encoders



Recurrent Neural Networks (RNNs) : Elman's Architecture

- There exist seveal indicators to measure the predictive accuracy of each model (**Hsieh, et. al.,** 2011; **Theil**, 1973)

- **RMSE (Root Mean Square Error): R**epresents the sample standard deviation of the differences between predicted values and observed values.
- **MAPE (Mean Absolute Percentage Error):** Measures the size of the error in percentage terms. Most people are comfortable thinking in percentage terms, making the MAPE easy to interpret.
- Thanks to its recursive formulation, RNNs are not limited by the **Markov assumption** for sequence modeling:

$$p\{ x(t) \mid x(t-1), \ldots, x(1) \} = p\{ x(t) \mid x(t-1) \}$$

Simple Feed Forward Artificial Neural Network (MLP)

```
o-----o-----o
x(t) h(t) y(t)
```

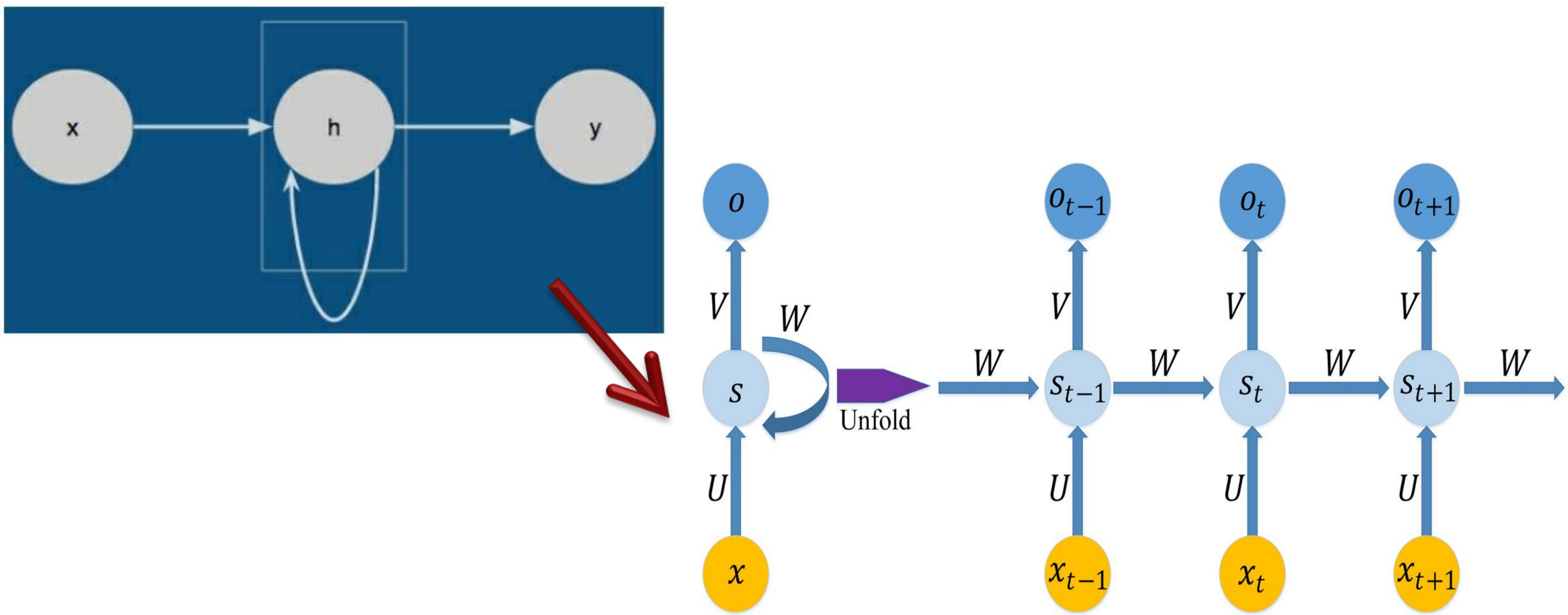
Recurrent Neural Network (Elman's Architecture)

```
 _
/ \ Wh
\ /
o-----o-----o
  Wi    Wo
```

$$h(t) = f(x(t)W_i + h(t-1)W_h + b_h)$$

Unfolding RNNs

- Although RNN models the time series well, it is hard to learn long-term dependencies because of the vanishing gradient problem in Back-Propagation Through Time (BPTT) **(Palangi H, et al., 2016)**



Back Propagation Through Time (BPTT)

- In BPTT updating weights is going to look exactly the same.
- **We can prove that the derivative of the loss function "Cross-Entropy" passes through the derivative of and the Softmax.** $p_j = \frac{e^{a_i}}{\sum_{k=1}^{N} e_k^a}$
- **Things are going to be multiplied together over and over again, due to the chain rule of calculus:**

$$d[W_h^T h(t-1)] / dW_h$$

- **The result is that gradients go down through the time (vanishing gradient problem) or they get very large very quickly (exploding gradient problem)**
- **RRNNs, GRUs, LSTMs solve the gradient problems with BPTT**

$$y(t)=\mathrm{softmax}(W_o^T h(t))$$

$$y(t)=\mathrm{softmax}(W_o^T f(W_h^T h(t-1)+W_x^T x(t)))$$

$$y(t)=\mathrm{softmax}(W_o^T f(W_h^T f(W_h^T h(t-2)+W_x^T x(t-1)) +W_x^T x(t)))$$

$$y(t)=\mathrm{softmax}(W_o^T f(W_h^T f(W_h^T f(W_h^T h(t-3)+W_x^T x(t-2))+W_x^T x(t-1))+W_x^T x(t)))$$

- We drop the bias in ordcer to display things simply

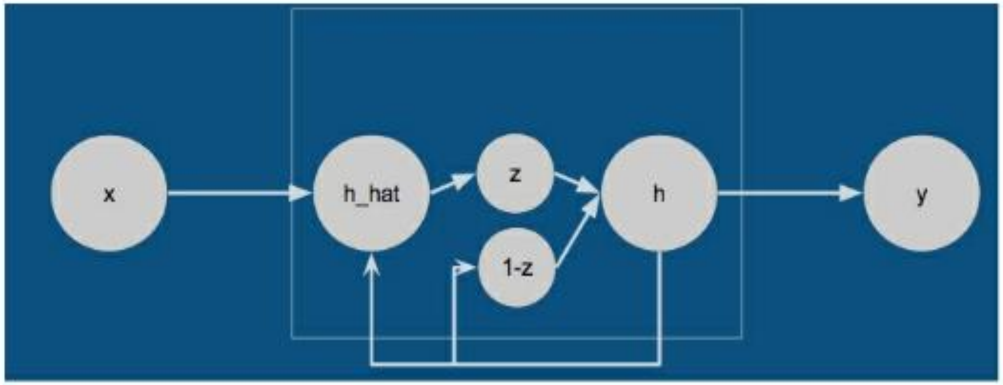
Rated Recurrent Neural Networks (RRNNs)

- The idea is to weight *f(x, h(t-1)),* which is the output of a simple RNN and *h(t-1)* which is the previous state **(Amari, et al., 1995)**.

- **We add a rating operation between what would have been the output of a simple RNN and the previous output value.**

- **This new operation can be seen as a gate since it takes a value between 0 an 1, and the other gate has to take 1 minus that value**

- **This is a gate that is choosing between 2 things: a) taking on the old value or taking the new value. As result we get a mixture of both.**



$$h\_hat(t) = f(x(t)W_x + h(t-1)W_h + b_h)$$

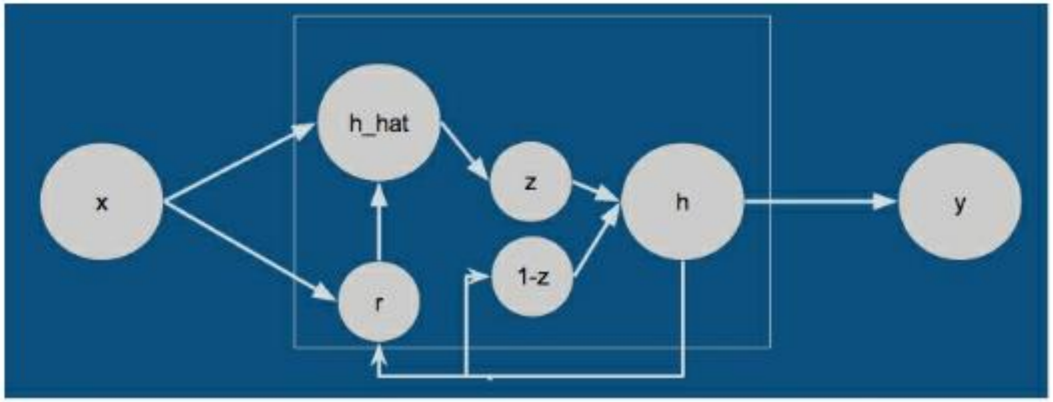
$$z(t) = sigmoid(x(t)W_{xz} + h(t-1)W_{hz} + b_z)$$

$$h(t) = (1 - z(t)) * h(t-1) + z(t) * h\_hat(t)$$

- **Z(t) is called the “rate”**

Gated Recurrent Neural Networks (GRUs)

- Gated Reccurrent Units were introduced in 2014 and are a simpler versione of LSTM. They have less parameters but same concepts **(Chung, et al., 2014).**
- **Recent research has also show that the accuracy between LSTM and GRU is comparable and even better with the GRUs in some cases.**
- **In GRUs we add one more gate with regard to RRNNs: the "reset gate r(t)" controlling how much of the previous hidden we will consider when we create a new candidate hidden value. In other words, it can "reset" the hidden value.**
- **The old gate of RRNNs is now called "update gate z(t)" balancing previous hidden values and new candidate hidden value for the new hidden value.**

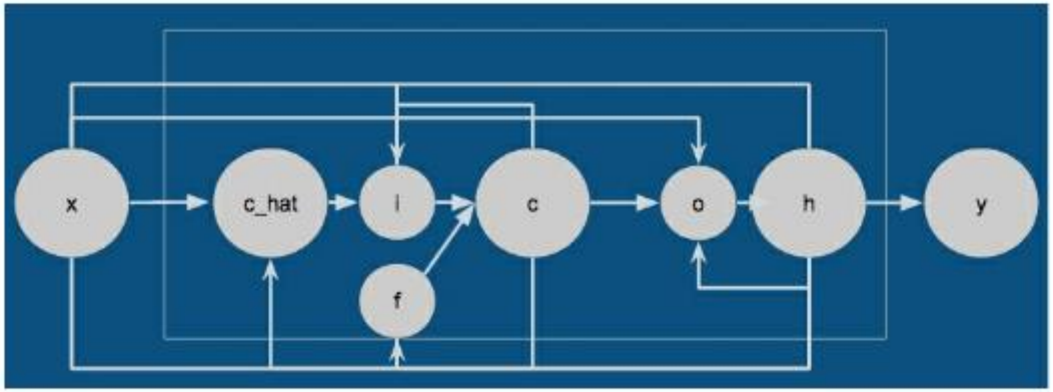


$$r_t = \sigma(x_t W_{xr} + h_{t-1} W_{hr} + b_r)$$
$$z_t = \sigma(x_t W_{xz} + h_{t-1} W_{hz} + b_z)$$
$$\hat{h}_t = g\,(x_t W_{xh} + (r_t \odot h_{t-1}) W_{hh} + b_h)$$
$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \hat{h}_t.$$

Long-Short Term Memories (LSTMs)

- LSTM is an effective solution for combating vanishing gradients by using memory cells **(Hochreiter, et al., 1997).**

- **A memory cell is composed of four units: an input gate, an output gate, a forget gate and a self-recurrent neuron**
- **The gates control the interactions between neighboring memory cells and the memory cell itself. Whether the input signal can alter the state of the memory cell is controlled by the input gate. On the other hand, the output gate can control the state of the memory cell on whether it can alter the state of other memory cell. In addition, the forget gate can choose to remember or forget its previous state.**



$$
\begin{aligned}
i_t &= \sigma(x_t W_{xi} + h_{t-1} W_{hi} + c_{t-1} W_{ci} + b_i) \\
f_t &= \sigma(x_t W_{xf} + h_{t-1} W_{hf} + c_{t-1} W_{cf} + b_f) \\
c_t &= f_t c_{t-1} + i_t \tanh(x_t W_{xc} + h_{t-1} W_{hc} + b_c) \\
o_t &= \sigma(x_t W_{xo} + h_{t-1} W_{ho} + c_t W_{co} + b_o) \\
h_t &= o_t \tanh(c_t)
\end{aligned}
$$

Deep LSTM/GRU Architectures

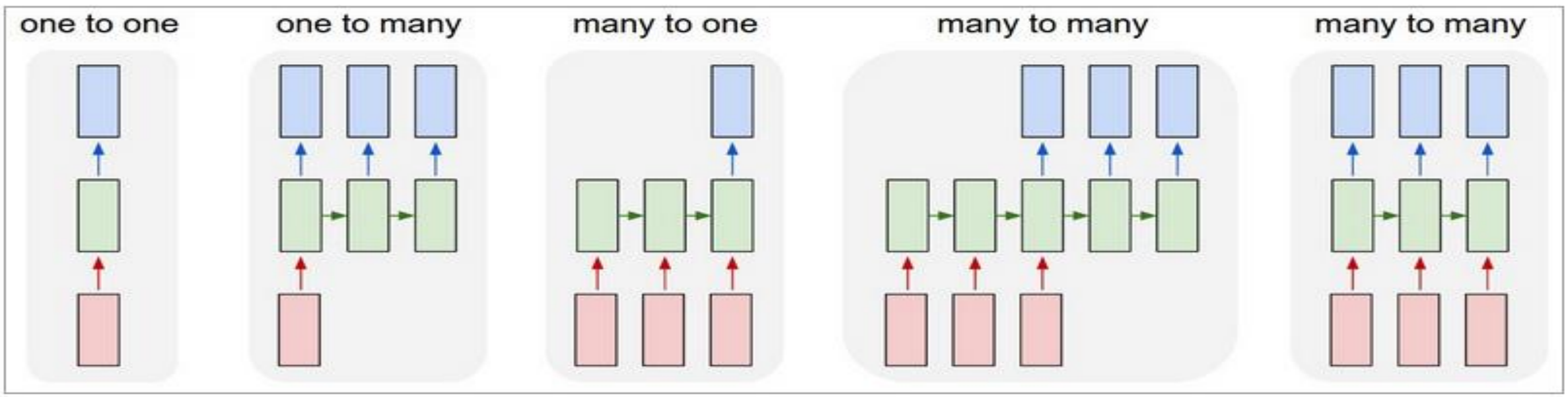
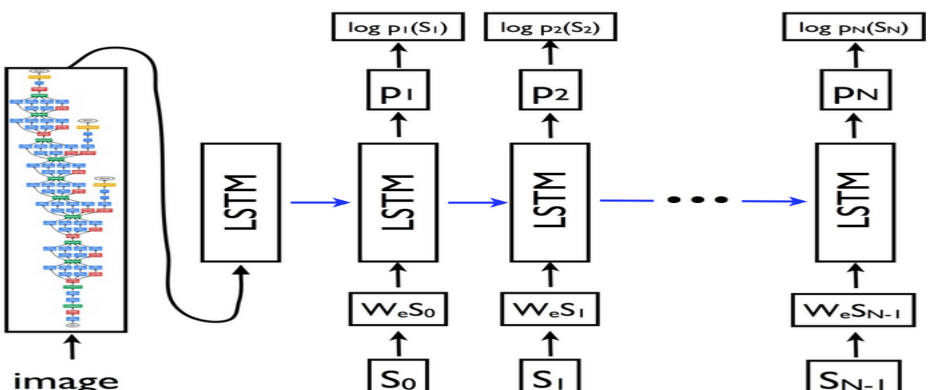
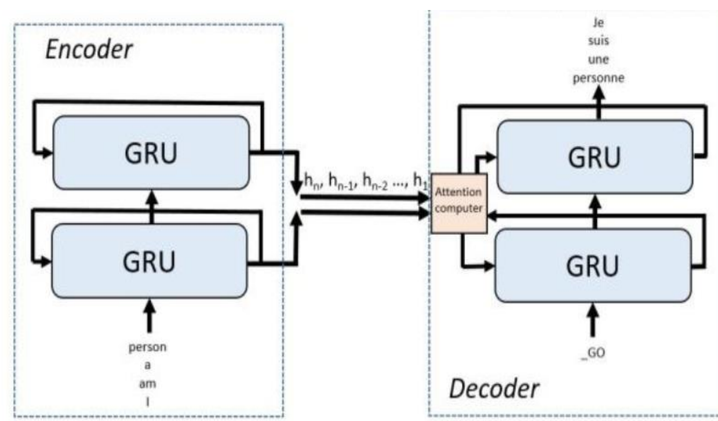


Image Caption Generator



Seq2seq model



Metrics

- There exist several indicators to measure the predictive accuracy of each model (**Hsieh, et. al.,** 2011; **Theil**, 1973)

- **RMSE (Root Mean Square Error): R**epresents the sample standard deviation of the differences between predicted values and observed values.
- **MAPE (Mean Absolute Percentage Error):** Measures the size of the error in percentage terms. Most people are comfortable thinking in percentage terms, making the MAPE easy t interpret.
- **Theil U:** Theil U is a relative measure c the difference between two variable: It squares the deviations to give mor weight to large errors and t exaggerate errors.

RMSE$= \sqrt{\frac{\sum_{t=1}^{T}(\hat{y}_t - y_t)^2}{n}}$

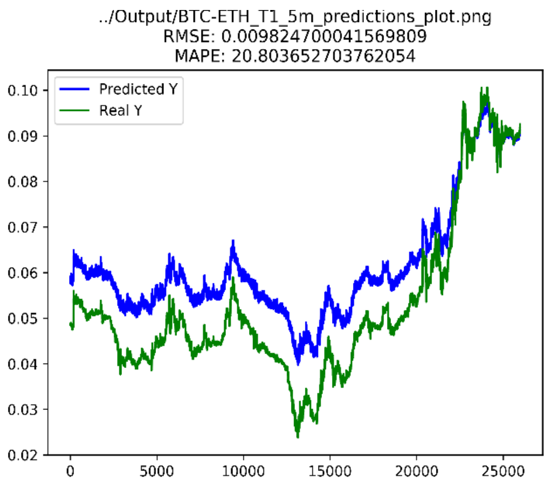
$$MAPE = \frac{\sum_{t=1}^{N}\left|\frac{y_t - y_t^*}{y_t}\right|}{N}$$

$$Theil\ U = \frac{\sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - y_t^*)^2}}{\sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t)^2} + \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t^*)^2}}$$

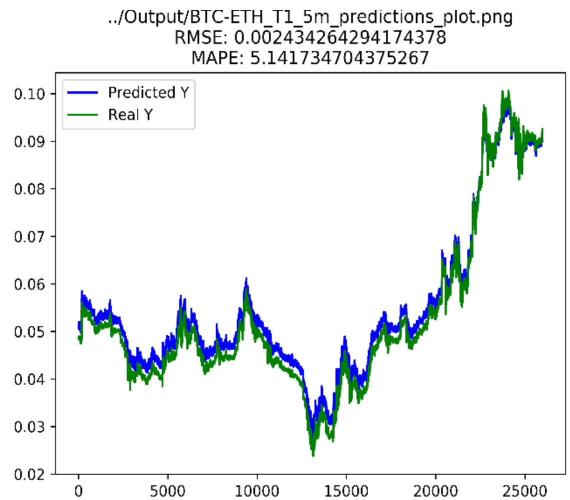
In these equations, yt is the actual value and is the predicted value.

Results: Bitcoin BTC-ETH exchange Time Series Prediction – 5 mins (Poloniex)

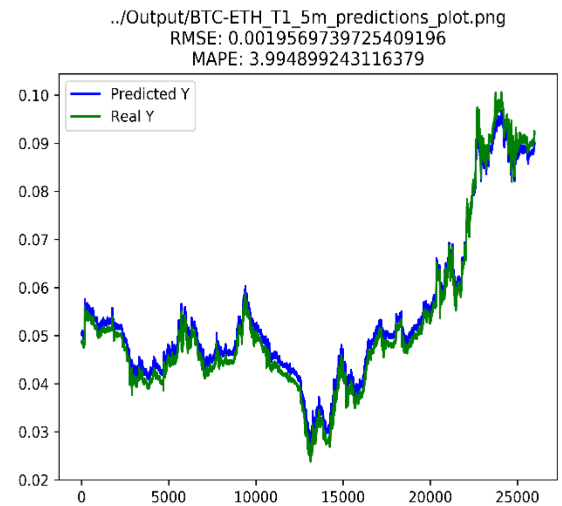
Test Set : 10%



1 Epoch



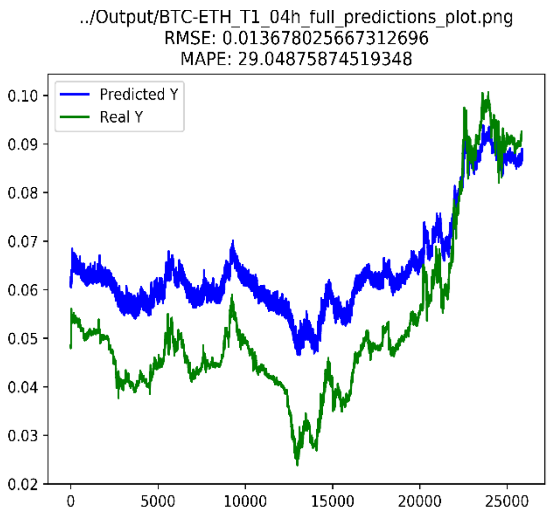
10 Epochs



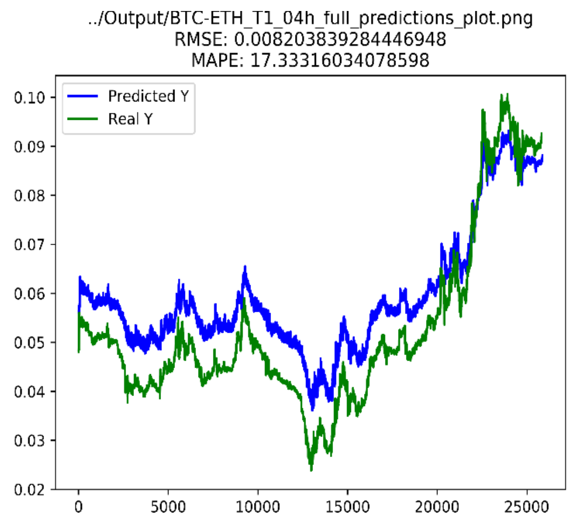
100 Epochs

Results: Bitcoin BTC-ETH exchange Time Series Prediction – 4 hours (Poloniex)

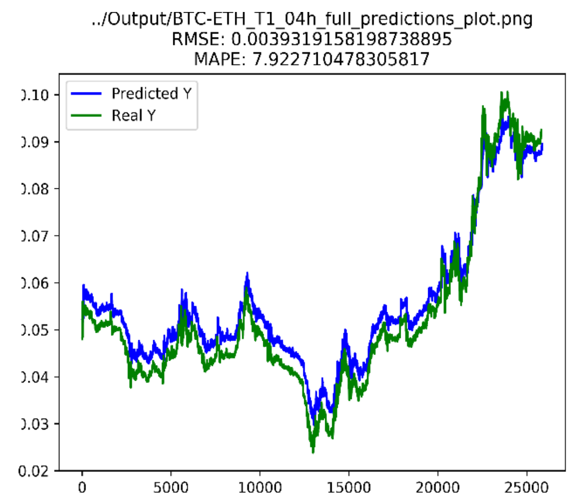
Test Set : 10%



1 Epoch



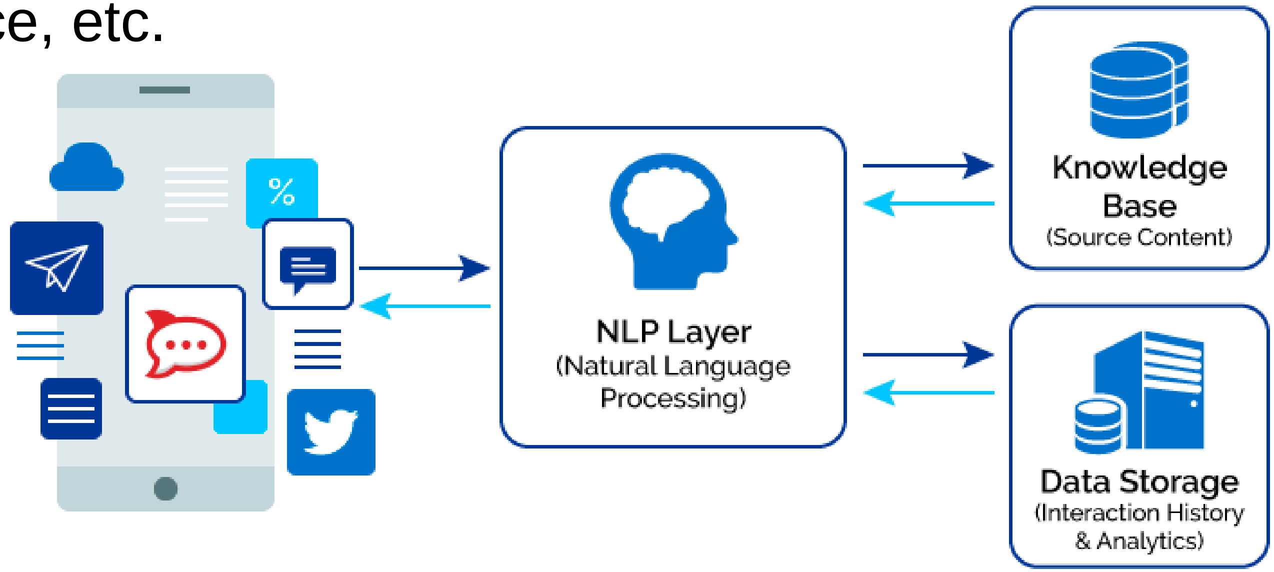
10 Epochs



100 Epochs

Textual Big Data alias The problem of the Natural Languale Processing - NLP

- Understanding **complex language utterances is one of the hardest challenge** for Artificial Intelligence (AI) and Machine Learning (ML).
- **NLP** is everywhere because people communicate most everything: web search, advertisement, emails, customer service, etc.



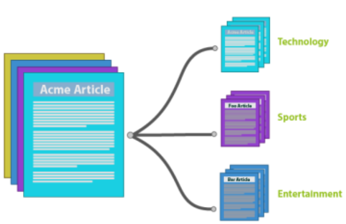
Deep Learning and NLP

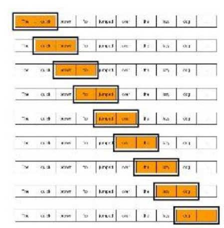
- "Deep Learning" approaches have obtained very high performance across many different **NLP** tasks. These models can often be trained with a **single end-to-end model** and do not require traditional, task-specific feature engineering.
  (*Stanford University School Of Engineering – CS224D*)

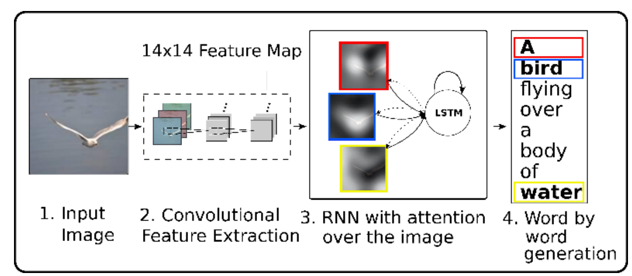
- **Natural language processing** is shifting from statistical methods to **Neural Networks**.

7 NLP applications where Deep Learning achieved «state-of-art» performance

- **1 Text Classification**: Classifying the topic or theme of a document (i.e. **Sentiment Analysis**).
- **2 Language Modeling**: Predict the **next word given the previous words**. It is fundamental for other tasks.
- **3 Speech Recognition**: Mapping an **acoustic signal** containing a spoken natural language utterance into the corresponding sequence of words intended by the speaker.
- **4 Caption Generation**: Given a **digital image**, such as a photo, generate a **textual description** of the contents of the image.





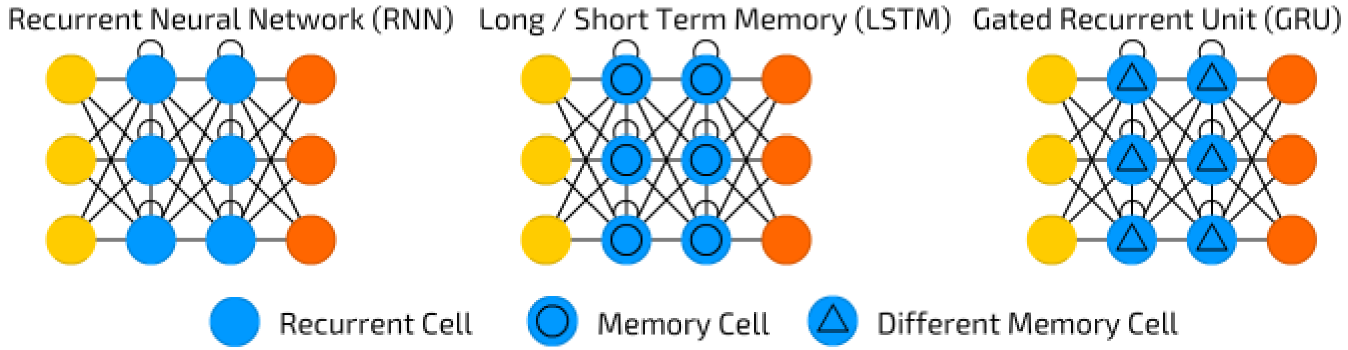
7 NLP applications where Deep Learning achieved «state-of-art» performance

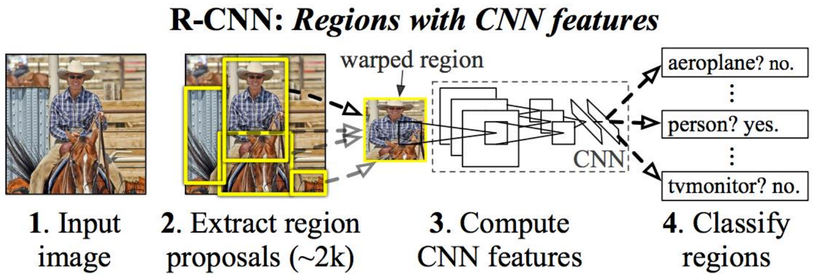
- **5 <u>Machine Translation</u>**: Automatic translation of text or speech from one language to another, is one [of] the most important applications of NLP.
- **6 <u>Document Summarization</u>**: It is the task where a short description of a text document is created.
- **7 <u>Question Answering</u>**: It is the task where the system tries to answer a user query that is formulated in the form of a question by returning the appropriate noun phrase such as a location, a person, or a date. (i.e. Who killed President Kennedy? Oswald)



Text Classification Models

- **RNN, LSTM, GRU, ConvLstm, RecursiveNN, RNTN, RCNN**
- The modus operandi for text classification involves the use of a pre-trained **word embedding** for **representing words** and a **deep neural networks** for **learning how to discriminate documents** on classification problems.



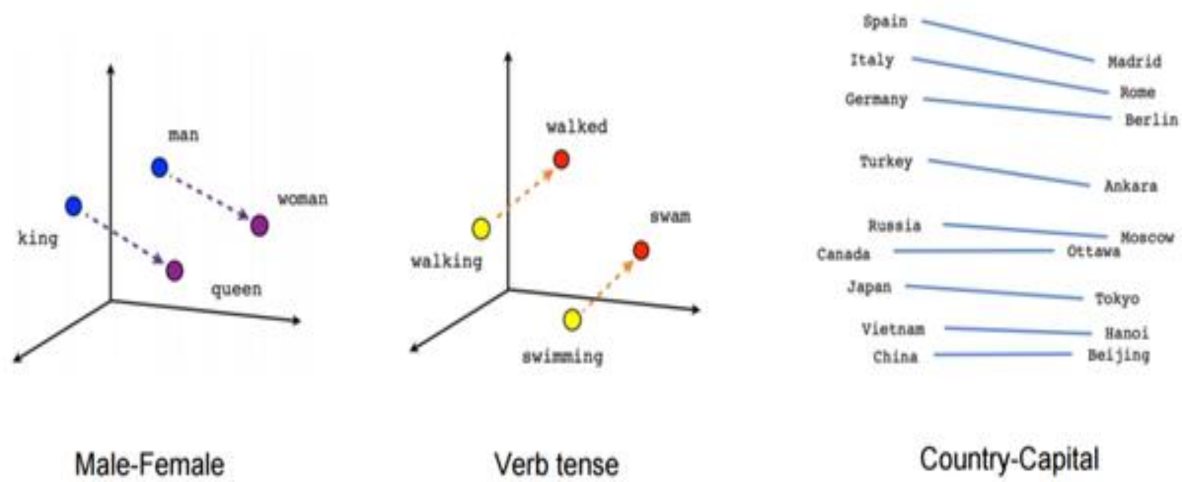


- The **non-linearity of the NN** leads to superior classification accuracy.

Word Embedding & Language Modeling

- Word embedding is the collective name for a set of language modeling and feature learning techniques for natural language processing (NLP) where words or sentences from the vocabulary are mapped to vectors of real numbers.

- These vectors are semantically cor-related by metrics like cosine distan-ce

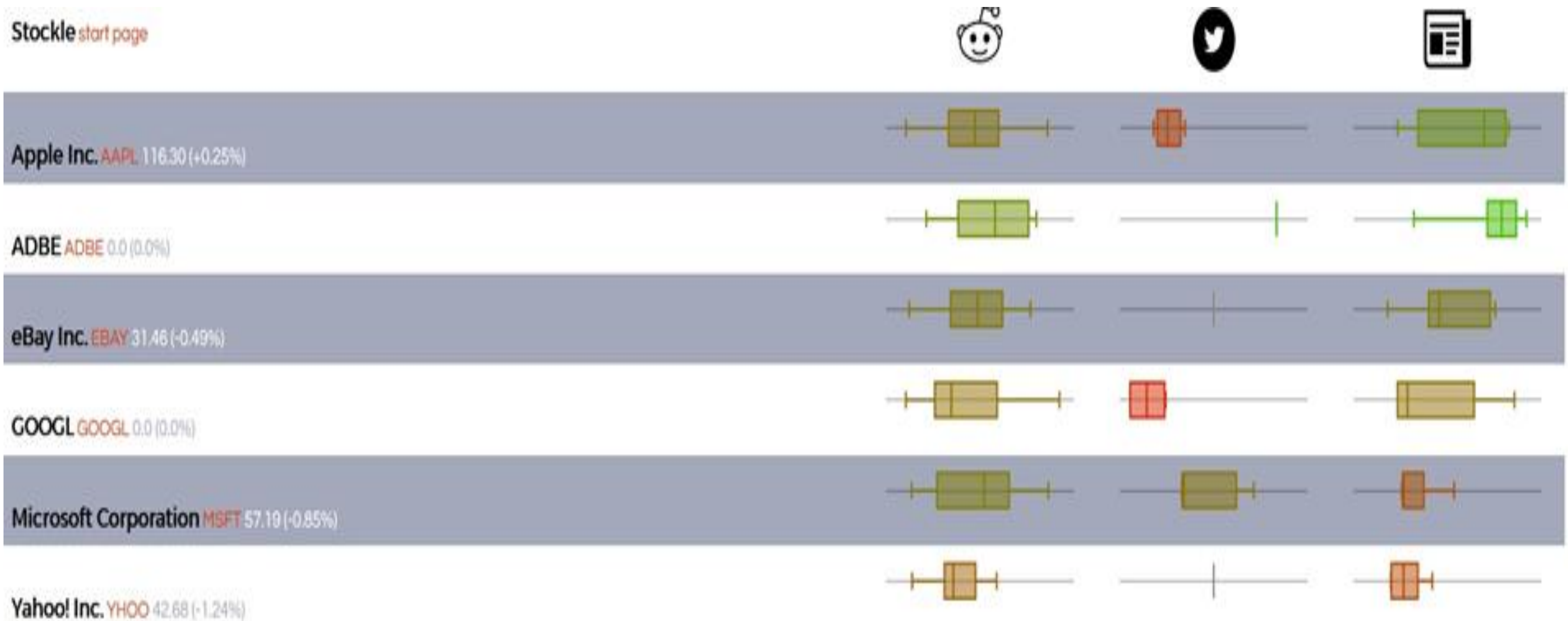


Sentiment Analysis (*Ain, et al. 2017*)

- **<u>Sentiments</u>** of users that are expressed on the web has great influence on the readers, product vendors and politicians.
- **<u>Sentiment Analysis</u>** refers to text organization for the classification of mind-set or feelings in different manners such as negative, positive, favorable, unfavorable, thumbs up, thumbs down, etc. Thanks to DL, the SA can be visual as well.

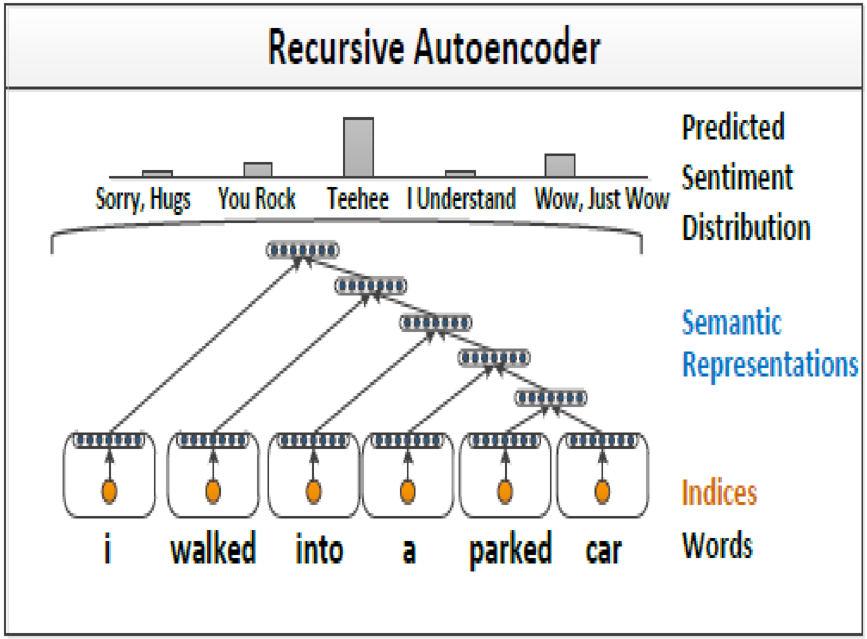
Discovering people opinions, emotions and feelings about a product or service

Sentiment Analysis with Feedback



Recursive Neural Tensor Networks (RecursiveNN) (Socher, R., et al., 2011b)

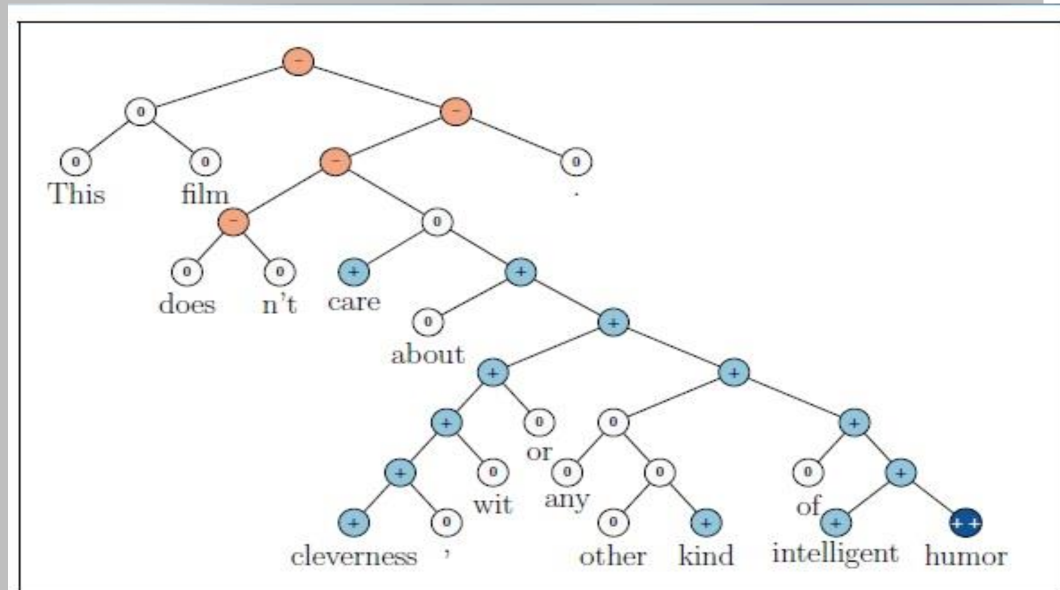
- This models are recursive auto-encoders which learn semantic vector representations of phrases. Word indices (orange) are first mapped into a semantic vector space (blue).
- Then they are recursively merged by the same auto-encoder network into a fixed length sentence representation. The vectors at each node are used as features to predict a distribution over text labels.



Recursive Neural Tensor Networks (RNTN) (Socher, R., et al. 2013)

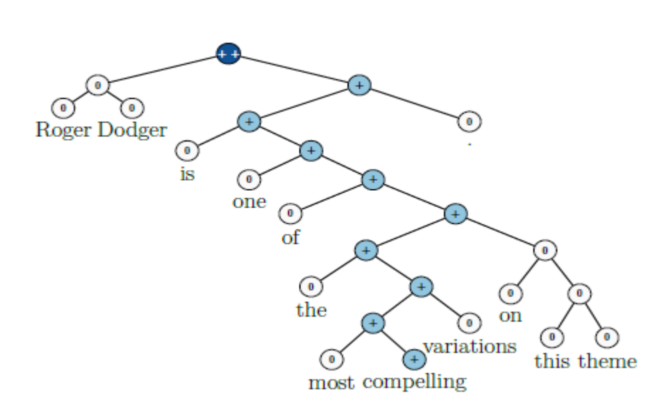
- **The Stanford Sentiment Treebank is the first corpus with fully labeled parse trees that allows for a complete analysis of the compositional effects of sentiment in language.**

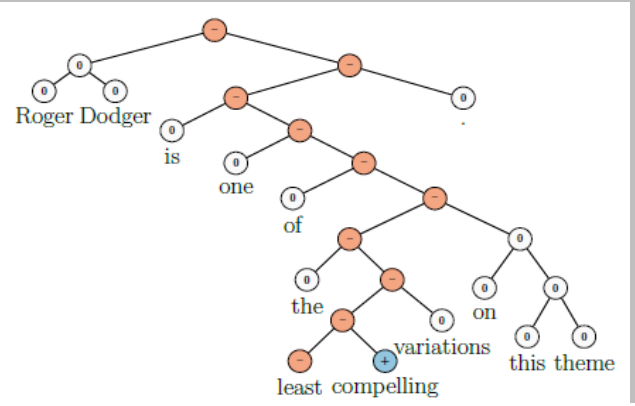
- **<u>RNTNs</u> compute parent vectors in a bottom up fashion using a compo-sitionality function and use node vectors as features for a classifier at that node.**



RNTN – Upside and Downside

- RNTNS are very efficient in terms of constructing sentence representations.
- RNTNs capture the semantics of a sentence via a tree structure. Its performance heavily depends on the performance of the textual tree construction.
- Constructing such a textual tree exhibits a time complexity of at least O(n2), where n is the length of the text.
- RNTNs are unsuitable for modeling long sentences or documents.



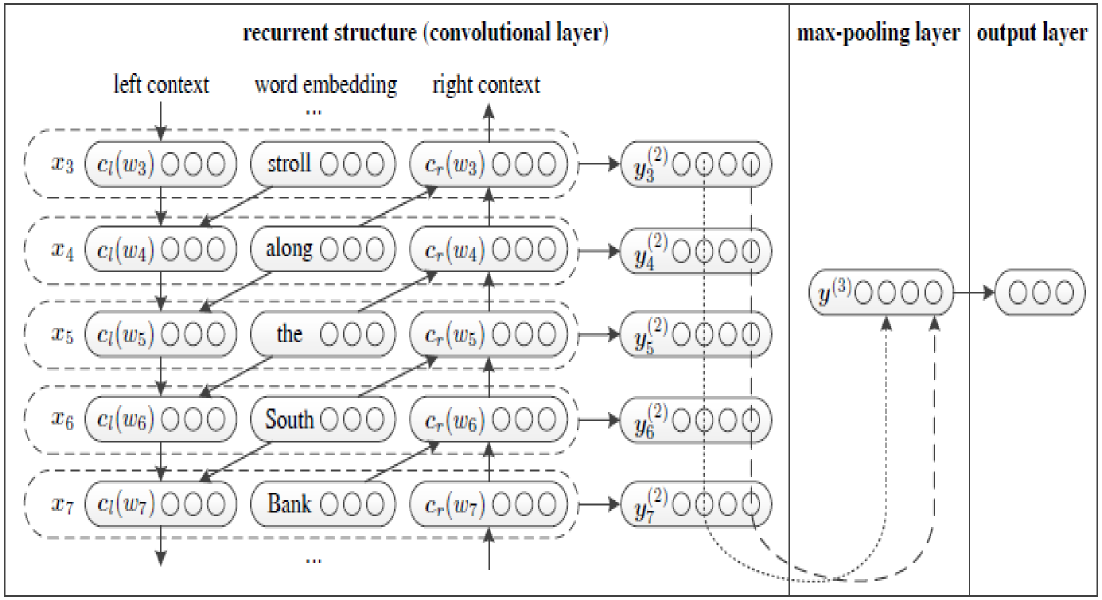


Recurrent Convolutional Neural Networks (RCNN) (*Lai, S., et al. 2015*)

- They adopt a a recurrent structure to **capture contextual information** as far as possible when learning word representations, which may introduce considerably **less noise compared** to traditional window-based neural networks.

- The **bi-directional recurrent structure** of RCNNs.

- **RCNNs** exhibit a time complexity of O(n)



RCNN Equations

- RCNNs exhibit a **time complexity of O(n)** , which is linearly correlated with the length of the text length.

- **7 equations** defining all the Neural Network topology

- **Input length** can be variable

$$c_l(w_i) = f(W^{(l)} c_l(w_{i-1}) + W^{(sl)} e(w_{i-1})) \quad (1)$$

$$c_r(w_i) = f(W^{(r)} c_r(w_{i+1}) + W^{(sr)} e(w_{i+1})) \quad (2)$$

$$x_i = [c_l(w_i); e(w_i); c_r(w_i)] \quad (3)$$

$$y_i^{(2)} = \tanh\left(W^{(2)} x_i + b^{(2)}\right) \quad (4)$$

$$y^{(3)} = \max_{i=1}^{n} y_i^{(2)} \quad (5)$$

$$y^{(4)} = W^{(4)} y^{(3)} + b^{(4)} \quad (6)$$

$$p_i = \frac{\exp\left(y_i^{(4)}\right)}{\sum_{k=1}^{n} \exp\left(y_k^{(4)}\right)} \quad (7)$$

RCNN in Keras

```
class SentimentModelRecConvNet:
    @staticmethod

    def build(input_length, vector_dim):
        hidden_dim_RNN = 200
        hidden_dim_Dense = 100

        embedding = Input(shape=(input_length, vector_dim))

        left_context = LSTM(hidden_dim_RNN, return_sequences = True)(embedding)                       # Equation 1
        # left_contex: batch_size x tweet_length x hidden_state_dim
        right_context = LSTM(hidden_dim_RNN, return_sequences = True, go_backwards = True)(embedding)  # Equation 2
        # right_cntext: come left_contex
        together = concatenate([left_context, embedding, right_context], axis = 2)                    # Equation 3
        semantic = TimeDistributed(Dense(hidden_dim_Dense, activation = "tanh"))(together)            # Equation 4
        pool_rnn = Lambda(lambda x: backend.max(x, axis = 1), output_shape = (hidden_dim_Dense, ))(semantic)  # Equation 5
        pool_rnn_args = Lambda(lambda x: backend.argmax(x, axis=1), output_shape = (hidden_dim_Dense, ))(semantic)

        output = Dense(1, input_dim = hidden_dim_Dense, activation = "sigmoid")(pool_rnn)            # Equations 6, 7

        deepnetwork = Model(inputs=embedding, outputs=output)
        deepnetwork_keywords = Model(inputs=embedding, outputs=pool_rnn_args)

        return [deepnetwork, deepnetwork_keywords]
```

RCNN: Feature Extraction

- RCNNs employ a max-pooling layer that automatically judges which words play key roles in text classification to capture the key components in texts.

- The most important words are the information most frequently selected in the max-pooling layer.

- Contrary to the most positive and most negative phrases in RNTN, RCNN does not rely on a syntactic parser, therefore, the presented n-grams are not typically "phrases".

	RCNN
P	well *worth* the; a *wonderful* movie; even *stinging* at; and *invigorating* film; and *ingenious* entertainment; and *enjoy* .; 's *sweetest* movie
N	A *dreadful* live-action; Extremely *boring* .; is *n't* a; 's *painful* .; Extremely *dumb* .; an *awfully* derivative; 's *weaker* than; incredibly *dull* .; very *bad* sign;
	RNTN
P	an amazing performance; most visually stunning; wonderful all-ages triumph; a wonderful movie
N	for worst movie; A lousy movie; a complete failure; most painfully marginal; very bad sign

RCNN applied to Extractive Text Summarization

- **Best keywords** lead to **best contextes** ---> **Summarization**

```
Tweet 29: "Giã  avete letto 136 pagine del piano scuola? #Fenomeni #labuonascuola"

Sentiment: -0.95 - -1
Keywords: pagine, avete, fenomeni, piano

Tweet 30: "\"Per l'#aternanza #scuola #lavoro bisogna passare da 11a 100milioni di euro\" #labuonascuola http://t.co/zGAzkni8rv"

Sentiment: -0.81 - -1
Keywords: euro, t, scuola, lavoro



Most significant keywords driving the sentiment decision:

Eccolo

Siamo

Scuola

Giuste

Escluso


Most significant sentences driving the sentiment decision:

...cambierã  solo se noi metteremo al centro...

...solo se noi metteremo al centro la...

...piã¹ grande spettacolo mai visto passodopopasso scuola...

...mai visto passodopopasso scuola labuonascuola...

...nessuno si senta escluso la buona scuola...
```

Recurrent Neural Networks are able to understand negations and other things

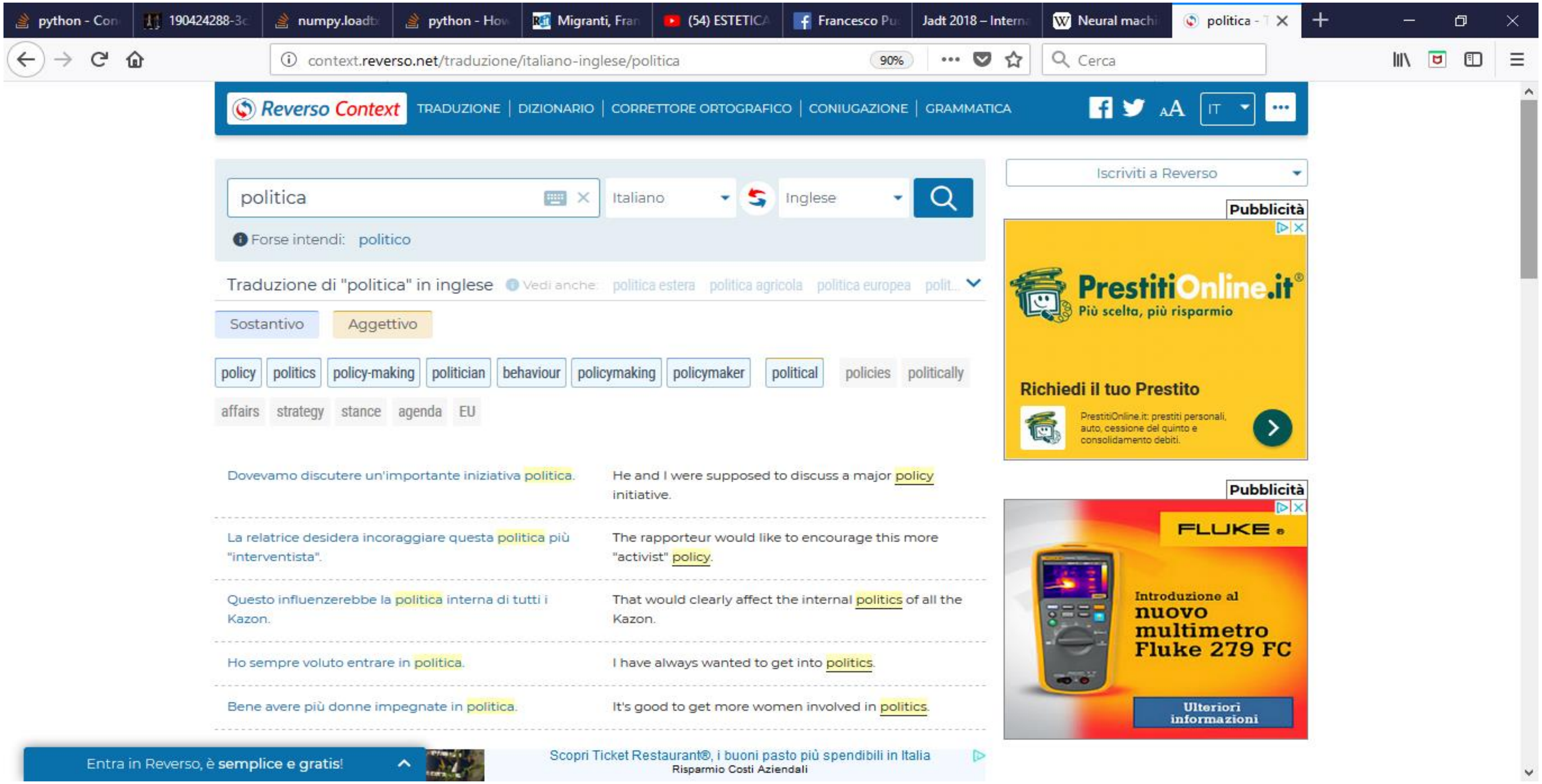
- Thanks to **word embeddings** semantics **RNNs** can recognize **nagations**, and complex **forms** of **language utterances**.

```
Tweet: This is a bad thing
- Sentiment: -0.72 - -1

Keywords: bad, thing, a, is

Tweet: This is not a bad thing
- Sentiment: 0.46 - +1

Keywords: not, thing, bad, a

Tweet: This is a positive thing
- Sentiment: 0.94 - +1

Keywords: positive, thing, a, is

Tweet: This is a very positive thing
- Sentiment: 0.91 - +1

Keywords: positive, very, thing, a

Tweet: I like Renzi politics
- Sentiment: 0.70 - +1

Keywords: like, renzi, politics, i

Tweet: I don't agree with Renzi Politics
- Sentiment: 0.16 - 0

Keywords: don't, agree, politics, renzi

Tweet: Renzi did a wrong international Politics
- Sentiment: -0.34 - -1

Keywords: wrong, did, renzi, international

Tweet: Renzi did a very good international Politics
- Sentiment: 0.74 - +1

Keywords: did, renzi, good, very

Tweet: Istat is a very good Institute of research
- Sentiment: 0.84 - +1

Keywords: good, very, research, istat

Tweet: Istat is not a good Institute of research - Sentiment: -0.78 - -1

Keywords: not, research, istat, institute
```

Multilingual Sentiment Analysis

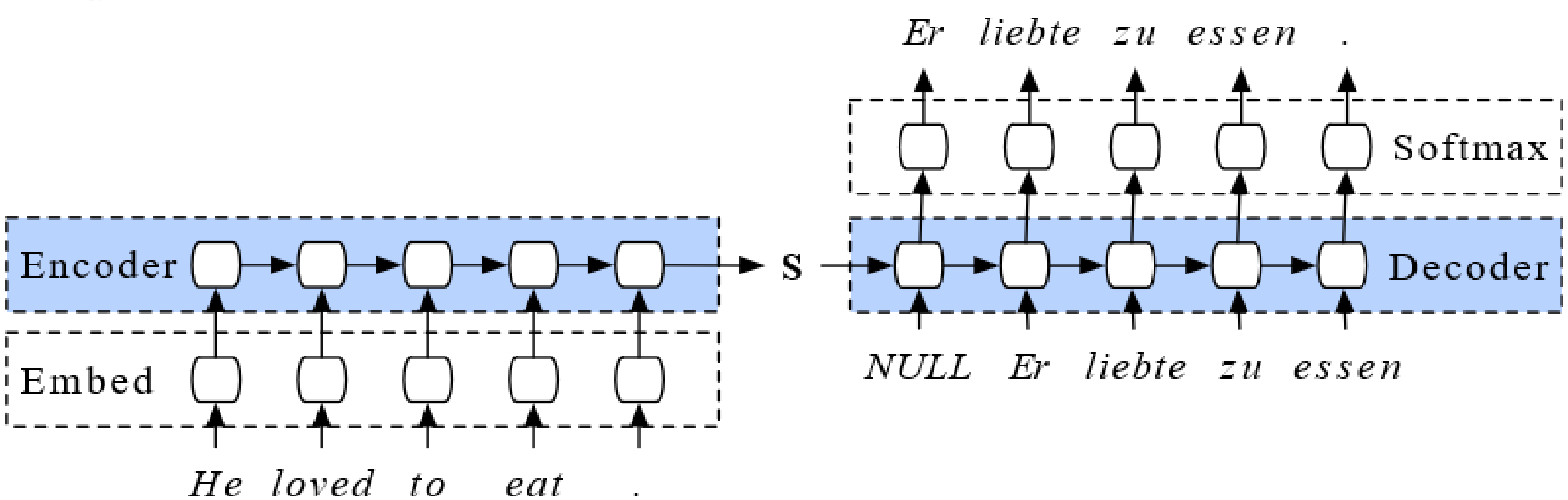
- During the training stage, the RCNN achieves 84% of accuracy on a validation set (selected at the 20% of the original dataset). On a test set of 380 tweets (provided by Semeval), the model returns around 82% of accuracy on positive tweets and 78% of accuracy on negatives, with an approximative 80% overall on a mixed tweets set.
- During the training we determined 3.2 millions of keywords, namely 2 for each tweet, the most important and the second in order of signinificancy.

Contextual Translations Web-sites



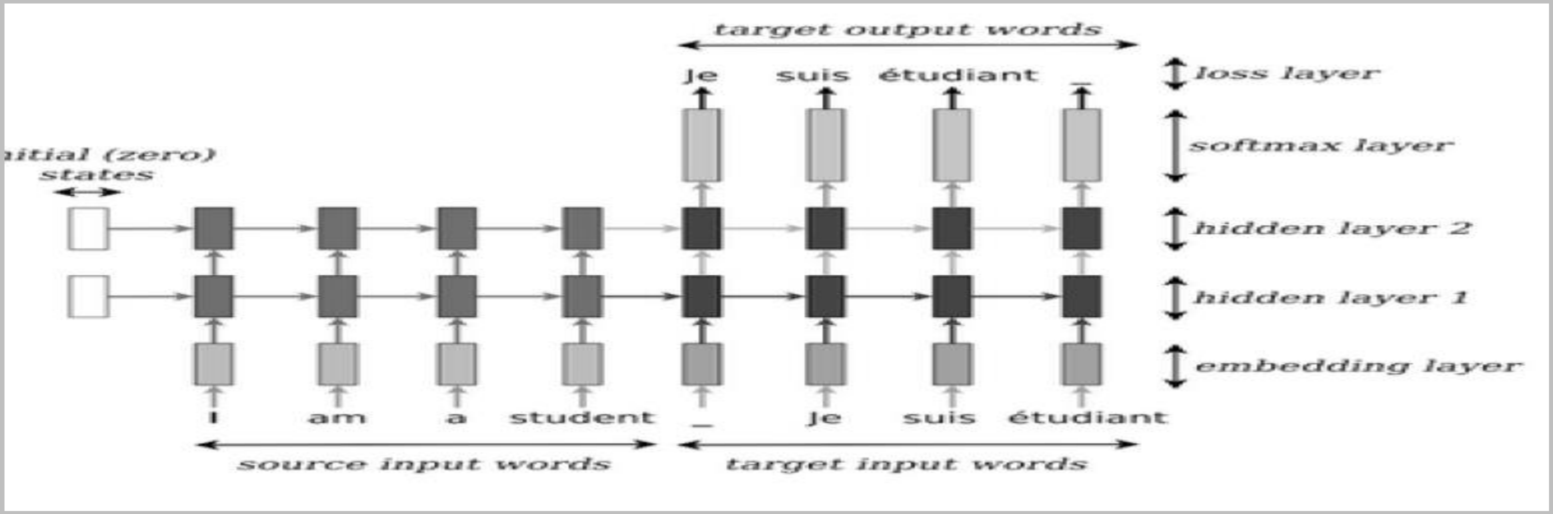
Neural Machine Translation

Input sentence:	Translation (PBMT):	Translation (GNMT):	Translation (human):
李克強此行將啟動中加總理年度對話機制，與加拿大總理杜魯多舉行兩國總理首次年度對話。	Li Keqiang premier added this line to start the annual dialogue mechanism with the Canadian Prime Minister Trudeau two prime ministers held its first annual session.	Li Keqiang will start the annual dialogue mechanism with Prime Minister Trudeau of Canada and hold the first annual dialogue between the two premiers.	Li Keqiang will initiate the annual dialogue mechanism between premiers of China and Canada during this visit, and hold the first annual dialogue with Premier Trudeau of Canada.



Neural Machine Translation

- **Neural machine translation (NMT) is an approach to "machine translation" that uses large ANN to predict the likelihood of a sequence of words, typically modeling entire sentences in a single integrated model (Bahdanau et al., 2014; Luong et Manning, 2016). .**



Neural Machine Translation

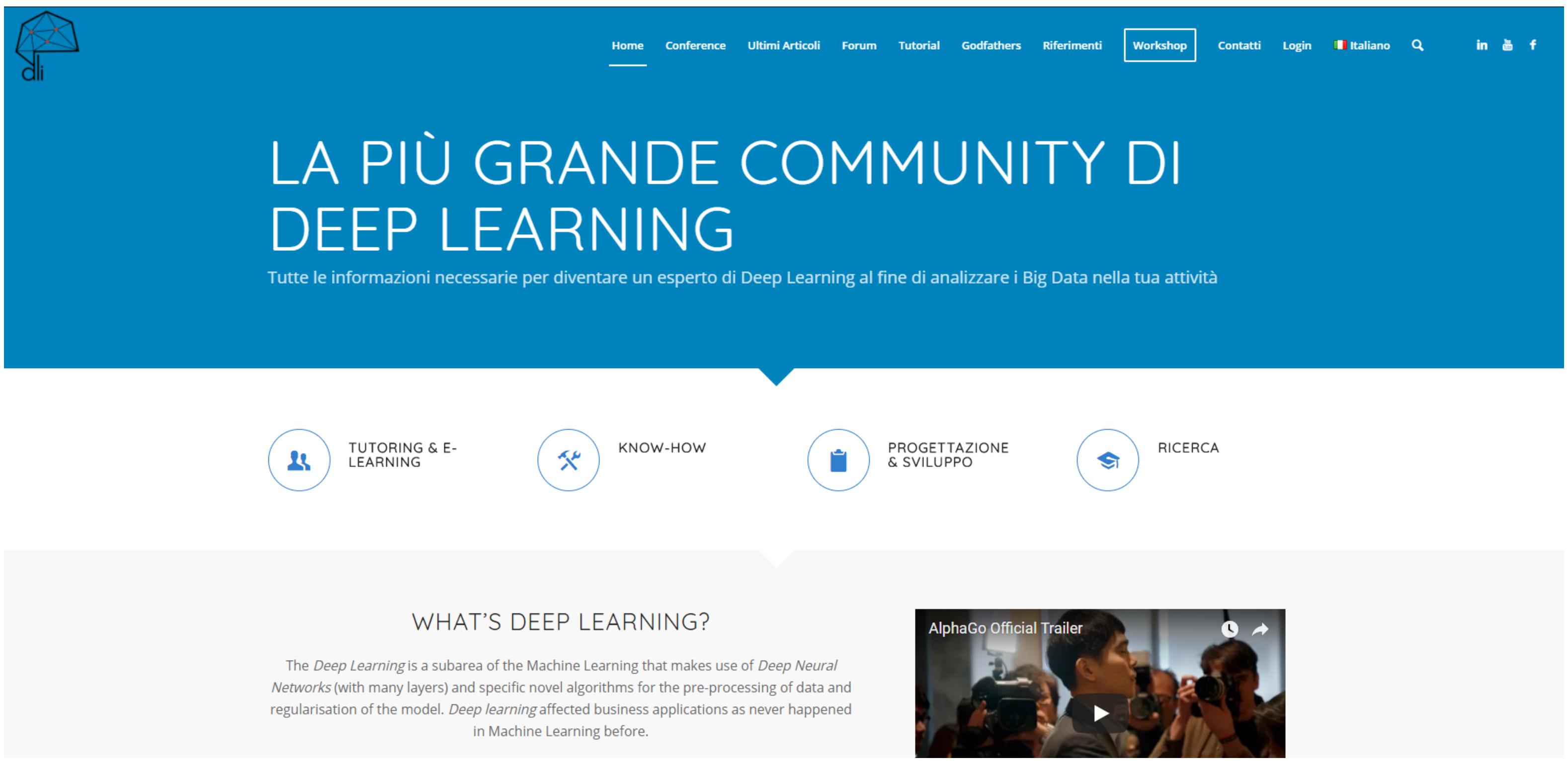
```
adottare un vocabolario condiviso è un suggerimento perfetto su come scrivere frasi comprensibili
adopt a shared vocabulary is a perfect suggestion on how to write understandable sentences

un altro suggerimento su come scrivere frasi semplici: evita le negazioni inutili
another suggestion about how to write simple sentences : avoid unnecessary <unk>

quasi 90 persone sono morte per una tempesta tropicale nelle filippine
nearly 90 people died for a tropical storm in the philippines
```

Figure 3. Some translations from Italian to English by means of the neural model trained by us.

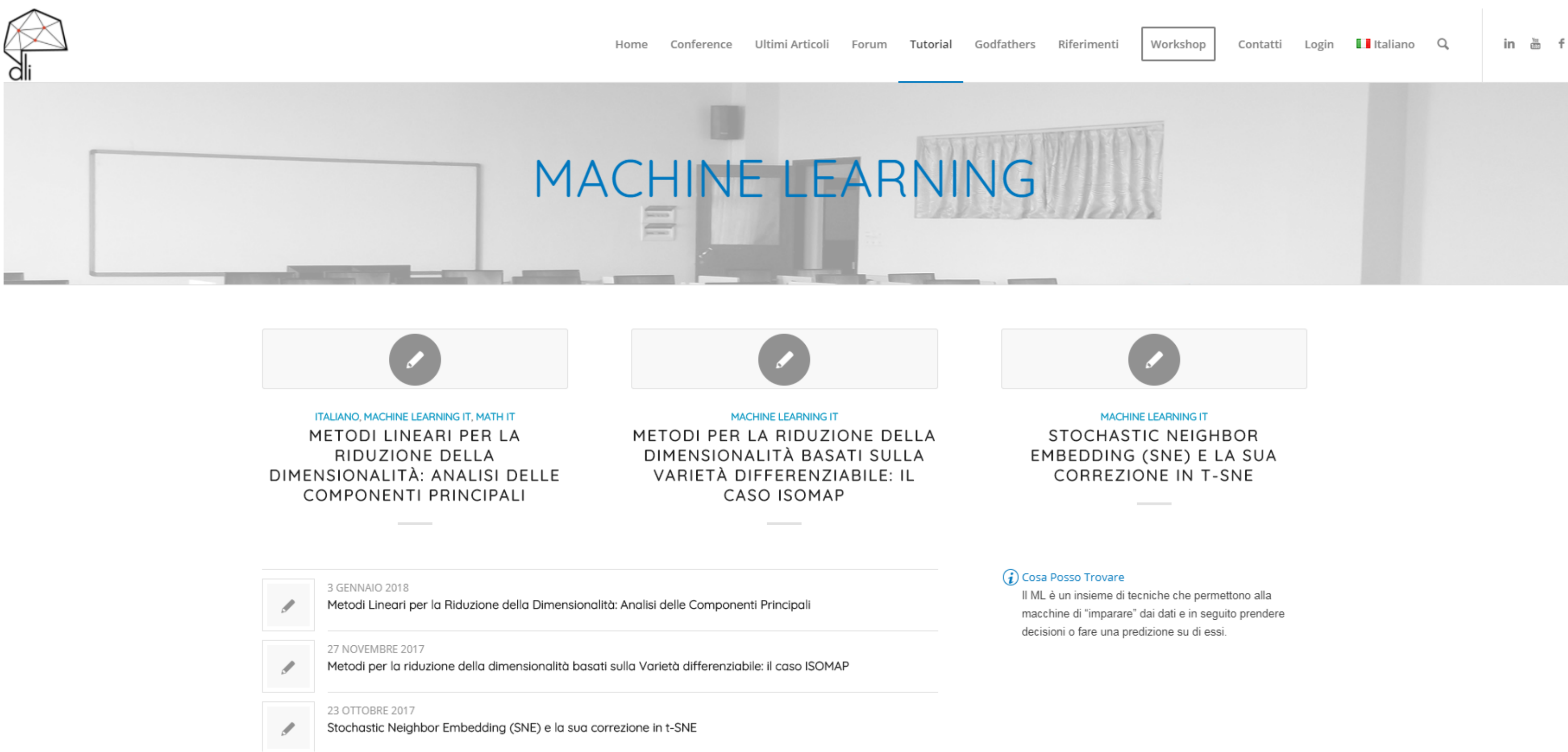
- We have tested the English RCNN model on the same italian SENTIPOLC 2016 test-set translated into English by our neural machine translation model. Results highlight a boost of performance : **78%** of accuracy on the test set versus the **43%** of the Italian trained RCNN model proving our strategy of stacking NMT and RCNN models is successful.

The Deep Learning Italia Project

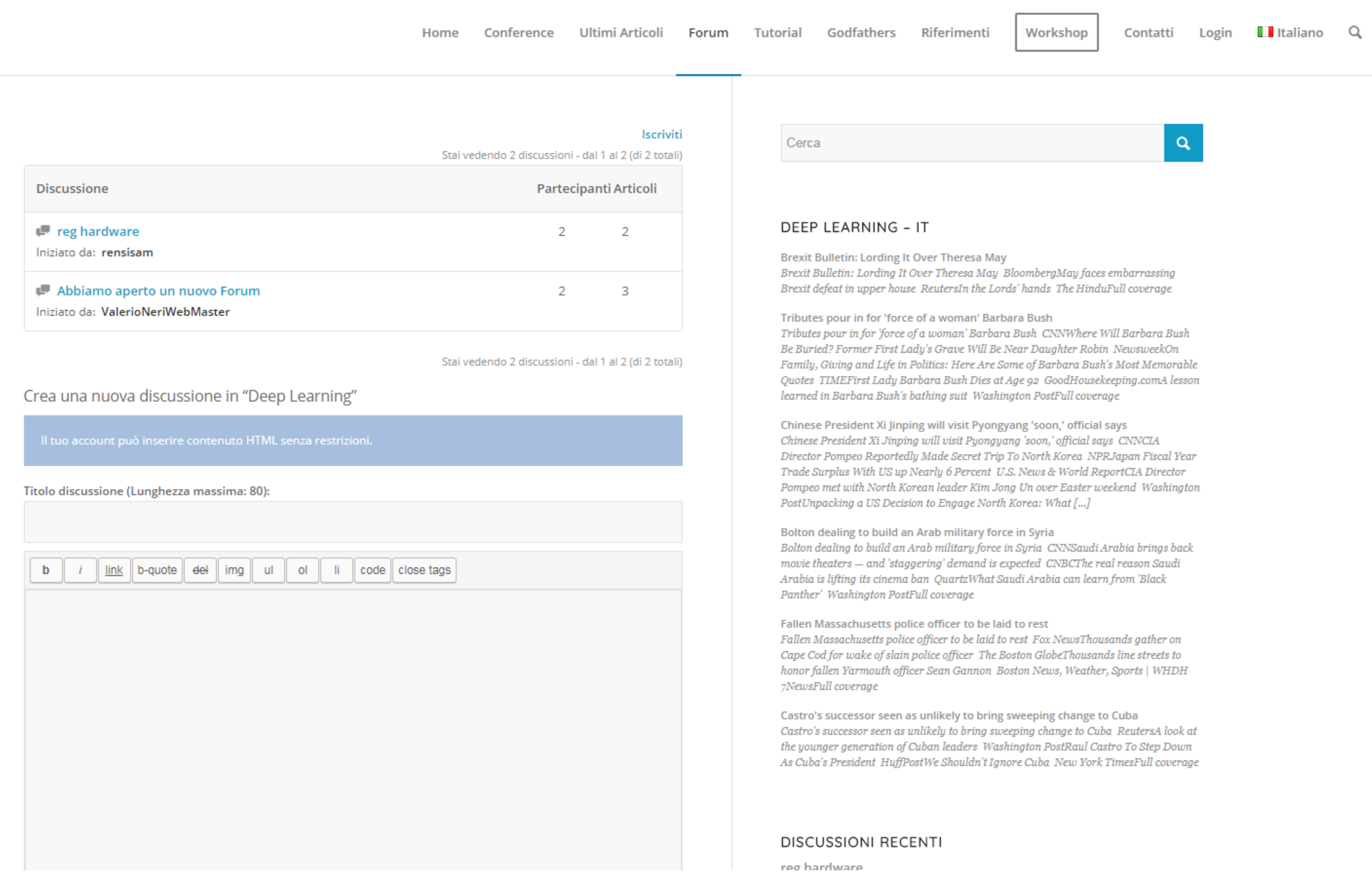
- A competence-sharing web-site designed **exclusively** for **Deep Learning**
- An **e-learning platform** for the disclosure of Deep Learning
- A **collector of professionals** around Deep Learning topics
- In the next future it will become a **complete development suite for Deep Learning**



Tutorials


Home
Conference
Ultimi Articoli
Forum
Tutorial
Godfathers
Riferimenti
Workshop
Contatti
Login
Italiano
MACHINE LEARNING
ITALIANO, MACHINE LEARNING IT, MATH IT
METODI LINEARI PER LA RIDUZIONE DELLA DIMENSIONALITÀ: ANALISI DELLE COMPONENTI PRINCIPALI
MACHINE LEARNING IT
METODI PER LA RIDUZIONE DELLA DIMENSIONALITÀ BASATI SULLA VARIETÀ DIFFERENZIABILE: IL CASO ISOMAP
MACHINE LEARNING IT
STOCHASTIC NEIGHBOR EMBEDDING (SNE) E LA SUA CORREZIONE IN T-SNE
3 GENNAIO 2018
Metodi Lineari per la Riduzione della Dimensionalità: Analisi delle Componenti Principali
27 NOVEMBRE 2017
Metodi per la riduzione della dimensionalità basati sulla Varietà differenziabile: il caso ISOMAP
23 OTTOBRE 2017
Stochastic Neighbor Embedding (SNE) e la sua correzione in t-SNE
Cosa Posso Trovare
Il ML è un insieme di tecniche che permettono alla macchine di "imparare" dai dati e in seguito prendere decisioni o fare una predizione su di essi.

Community


Home Conference Ultimi Articoli Forum Tutorial Godfathers Riferimenti Workshop Contatti Login Italiano

Iscriviti

Stai vedendo 2 discussioni - dal 1 al 2 (di 2 totali)

Discussione	Partecipanti	Articoli
reg hardware Iniziato da: rensisam	2	2
Abbiamo aperto un nuovo Forum Iniziato da: ValerioNeriWebMaster	2	3

Stai vedendo 2 discussioni - dal 1 al 2 (di 2 totali)

Crea una nuova discussione in "Deep Learning"

Il tuo account può inserire contenuto HTML senza restrizioni.

Titolo discussione (Lunghezza massima: 80):

b i link b-quote del img ul ol li code close tags

Cerca

DEEP LEARNING – IT

Brexit Bulletin: Lording It Over Theresa May
Brexit Bulletin: Lording It Over Theresa May BloombergMay faces embarrassing Brexit defeat in upper house ReutersIn the Lords' hands The HinduFull coverage

Tributes pour in for 'force of a woman' Barbara Bush
Tributes pour in for 'force of a woman' Barbara Bush CNNWhere Will Barbara Bush Be Buried? Former First Lady's Grave Will Be Near Daughter Robin NewsweekOn Family, Giving and Life in Politics: Here Are Some of Barbara Bush's Most Memorable Quotes TIMEFirst Lady Barbara Bush Dies at Age 92 GoodHousekeeping.comA lesson learned in Barbara Bush's bathing suit Washington PostFull coverage

Chinese President Xi Jinping will visit Pyongyang 'soon,' official says
Chinese President Xi Jinping will visit Pyongyang 'soon,' official says CNNCIA Director Pompeo Reportedly Made Secret Trip To North Korea NPRJapan Fiscal Year Trade Surplus With US up Nearly 6 Percent U.S. News & World ReportCIA Director Pompeo met with North Korean leader Kim Jong Un over Easter weekend Washington PostUnpacking a US Decision to Engage North Korea: What […]

Bolton dealing to build an Arab military force in Syria
Bolton dealing to build an Arab military force in Syria CNNSaudi Arabia brings back movie theaters — and 'staggering' demand is expected CNBCThe real reason Saudi Arabia is lifting its cinema ban QuartzWhat Saudi Arabia can learn from 'Black Panther' Washington PostFull coverage

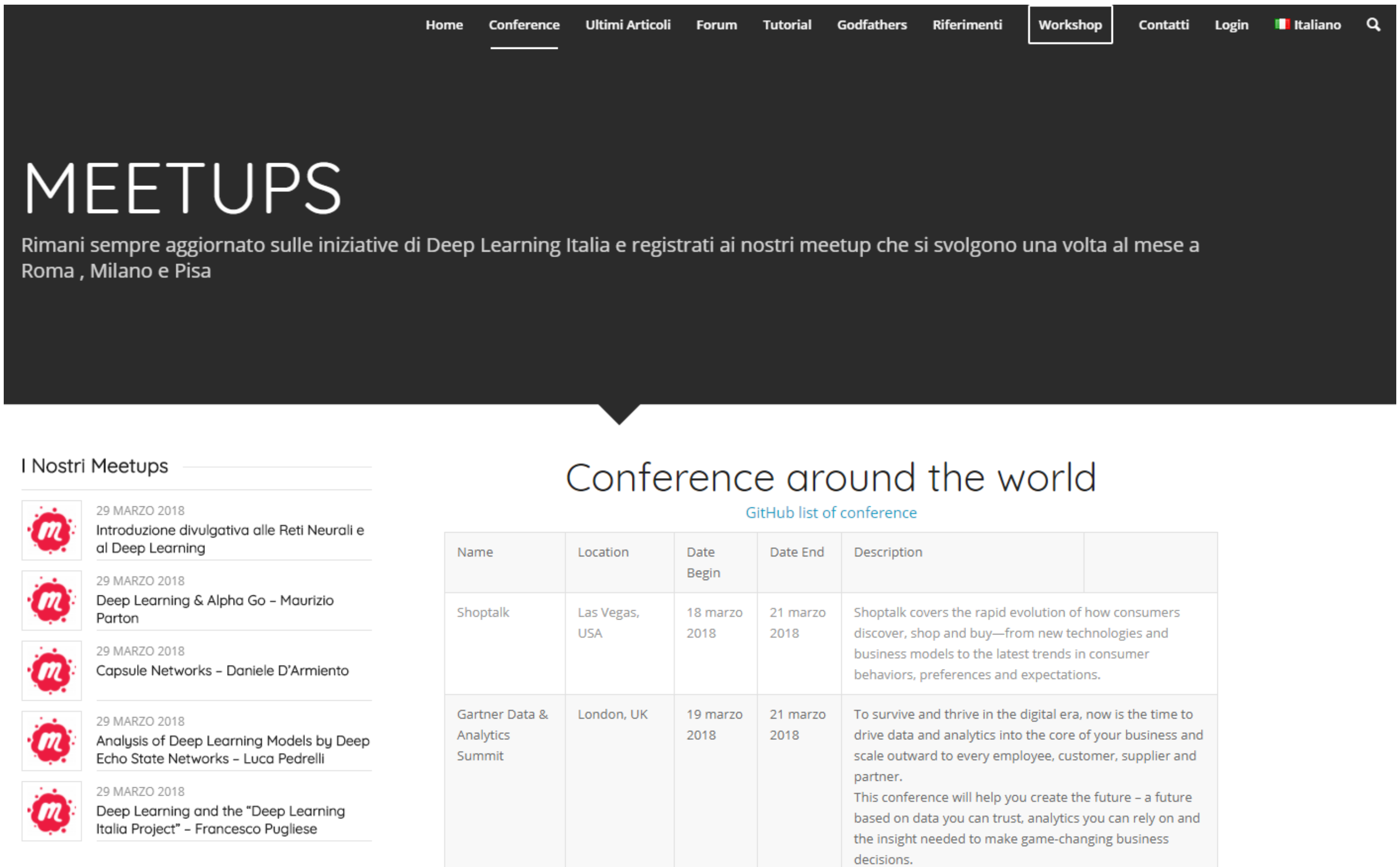
Fallen Massachusetts police officer to be laid to rest
Fallen Massachusetts police officer to be laid to rest Fox NewsThousands gather on Cape Cod for wake of slain police officer The Boston GlobeThousands line streets to honor fallen Yarmouth officer Sean Gannon Boston News, Weather, Sports | WHDH 7NewsFull coverage

Castro's successor seen as unlikely to bring sweeping change to Cuba
Castro's successor seen as unlikely to bring sweeping change to Cuba ReutersA look at the younger generation of Cuban leaders Washington PostRaul Castro To Step Down As Cuba's President HuffPostWe Shouldn't Ignore Cuba New York TimesFull coverage

DISCUSSIONI RECENTI

reg hardware

Meetup & Conferences



Home Conference Ultimi Articoli Forum Tutorial Godfathers Riferimenti Workshop Contatti Login Italiano

MEETUPS

Rimani sempre aggiornato sulle iniziative di Deep Learning Italia e registrati ai nostri meetup che si svolgono una volta al mese a Roma , Milano e Pisa

I Nostri Meetups

29 MARZO 2018
Introduzione divulgativa alle Reti Neurali e al Deep Learning

29 MARZO 2018
Deep Learning & Alpha Go – Maurizio Parton

29 MARZO 2018
Capsule Networks – Daniele D'Armiento

29 MARZO 2018
Analysis of Deep Learning Models by Deep Echo State Networks – Luca Pedrelli

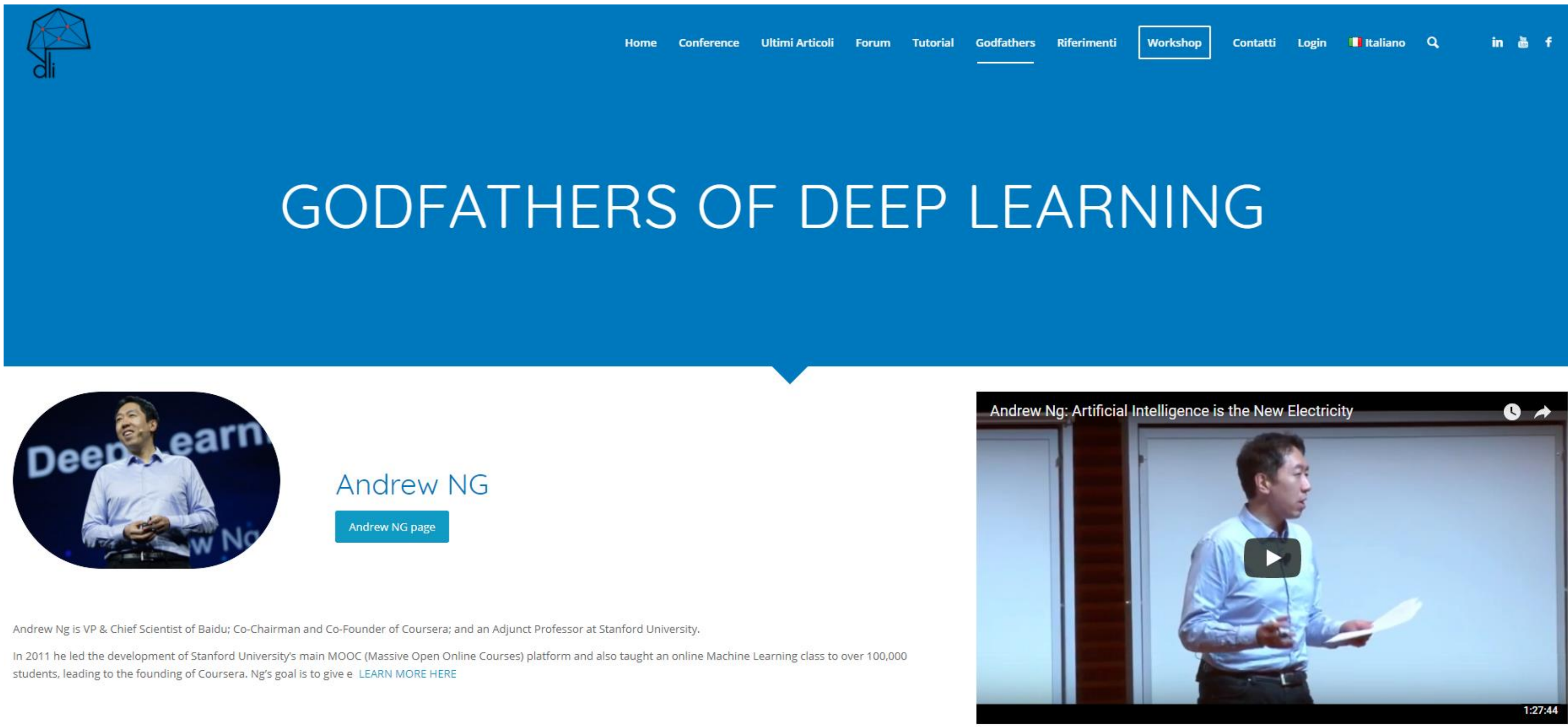
29 MARZO 2018
Deep Learning and the "Deep Learning Italia Project" – Francesco Pugliese

Conference around the world

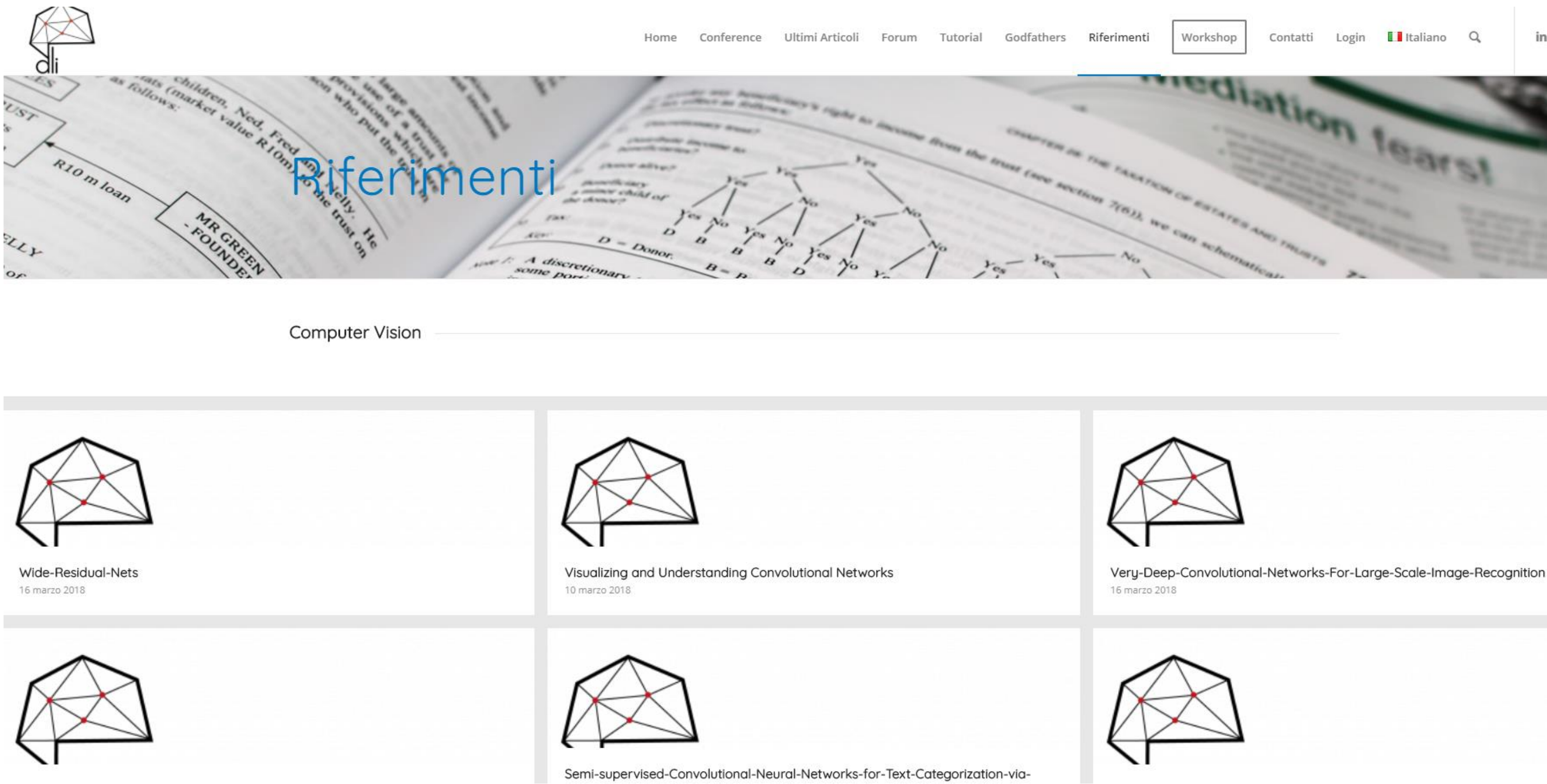
GitHub list of conference

Name	Location	Date Begin	Date End	Description	
Shoptalk	Las Vegas, USA	18 marzo 2018	21 marzo 2018	Shoptalk covers the rapid evolution of how consumers discover, shop and buy—from new technologies and business models to the latest trends in consumer behaviors, preferences and expectations.	
Gartner Data & Analytics Summit	London, UK	19 marzo 2018	21 marzo 2018	To survive and thrive in the digital era, now is the time to drive data and analytics into the core of your business and scale outward to every employee, customer, supplier and partner. This conference will help you create the future – a future based on data you can trust, analytics you can rely on and the insight needed to make game-changing business decisions.	

Goodfathers



REFERENCES


dli
Home
Conference
Ultimi Articoli
Forum
Tutorial
Godfathers
Riferimenti
Workshop
Contatti
Login
Italiano
in
Riferimenti
Computer Vision
Wide-Residual-Nets
16 marzo 2018
Visualizing and Understanding Convolutional Networks
10 marzo 2018
Very-Deep-Convolutional-Networks-For-Large-Scale-Image-Recognition
16 marzo 2018
Semi-supervised-Convolutional-Neural-Networks-for-Text-Categorization-via-

Workshop

WORKSHOP

DEEP LEARNING MODEL HANDS-ON (2 DAYS)

In questo corso si vedranno nel dettaglio tecnico e di codice i diversi modelli di Deep Learning con applicazioni pratiche su casi reali.

FROM 0 TO EXPERT IN DEEP LEARNING (3 DAYS)

In questo corso si affronteranno teoricamente e praticamente tutti i concetti che hanno portato al grande successo del deep learning.

COME CAPIRE LE ESIGENZE DEL CLIENTE E VEDERE UNA SOLUZIONE AI (1 DAY)

In questo corso si cercherà di capire come interpretare le esigenze del cliente che si affaccia per la prima volta al mondo dell'Artificial Intelligence (AI). Questo ci aiuterà a capire se e come vedere una soluzione AI.

COME CREARE E GESTIRE UN GRUPPO DI DATA SCIENTISTS (1 DAY)

In questo corso ci sarà un introduzione sui concetti principali di Artificial Intelligence (AI) e come possono essere trasferiti in Azienda. Si affronteranno casi d'uso che hanno portato al successe molte aziende che hanno deciso di utilizzare l'AI per migliorare il proprio business.

INTELLIGENZA ARTIFICIALE PER LE STRATEGIE AZIENDALI (1/2 DAY)

In questo corso si affronterà il tema di come l'AI può impattare le strategie aziendali e migliorare diversi processi e il decision making in ambito manageriale.

COME CAPIRE SE LA TUA AZIENDA È PRONTA PER UNA SOLUZIONE DI INTELLIGENZA ARTIFICIALE (1/2 DAY)

Capire se la propria azienda è pronta e ha i mezzi/dati per utilizzare al meglio l'AI è un processo molto lungo e dispendioso se non si sa bene cosa si deve cercare e di cosa si ha bisogno. In questo corso discuteremo i passi fondamentali da fare quando si ci muove verso soluzioni AI.

CORSO INTRODUTTIVO ALL'USO DELL'ARTIFICIAL INTELLIGENCE IN AZIENDA (2 DAY)

In questo corso ci sarà un introduzione sui concetti principali di Artificial Intelligence (AI) e come possono essere trasferiti in Azienda. Si affronteranno casi d'uso che hanno portato al successe molte aziende che hanno deciso di utilizzare l'AI per migliorare il proprio business.

COME USARE IL DEEP LEARNING E BIG DATA PER INCREMENTARE IL TUO BUSINESS (2 GIORNI)

In questo corso si affronteranno le tematiche inerenti ai Big Data e il Deep Learning e come queste de aree si uniscono per aiutare le aziende a trarre valore dai propri dati.

Inviaci email

Nome *

E-Mail *

Oggetto *

Workshop

Select workshop

Messaggio *

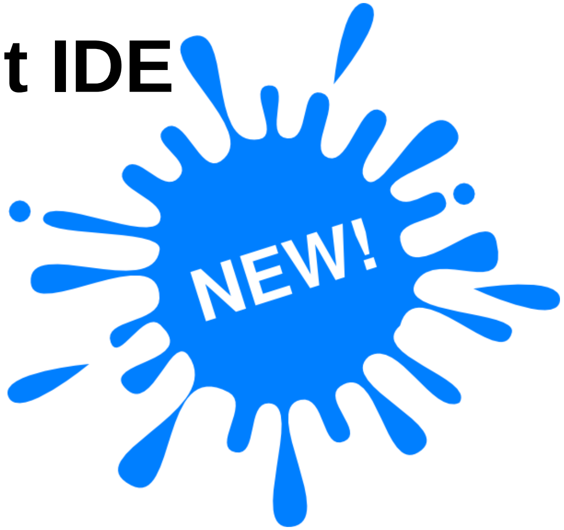
Si prega di risolvere la semplice equazione *

1 + 3 = ?

Inviare

New Features

- Deep Learning **Development IDE**
- **Repository** of Datasets



REFERENCES

Chellappa, R. (2012). Mathematical statistics and computer vision. *Image and Vision Computing*, *30*(8), 467-468.

Tsukahara, J. S., Harrison, T. L., & Engle, R. W. (2016). The relationship between baseline pupil size and intelligence. *Cognitive Psychology*, *91*, 109-123.

Hubel, D. H., & Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, *195*(1), 215-243.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929-1958.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (pp. 807-814).

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

Sánchez, J., Perronnin, F., Mensink, T., & Verbeek, J. (2013). Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, *105*(3), 222-245.

REFERENCES

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Abate, T., Van Huis, A., & Ampofo, J. K. O. (2000). Pest management strategies in traditional agriculture: an African perspective. *Annual review of entomology*, *45*(1), 631-659.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, *2*(1), 1-127

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI magazine*, *17*(3), 37

Hinton, G. E., Osindero, S., and Teh, Y. (2006). A fast learning algorithm for deep belief nets. Neural Comput. 18, 1527–1554. doi: 10.1162/neco.2006.18.7.1527

Holland, J., H., (1975). Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor.

REFERENCES

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989)
Backpropagation applied to handwritten zip code recognition. Neural Comput., 1(4):541{551.

McCulloch, W., S., and Pitts, W., (1943).“A logical calculus of the ideas immanent in nervous activity, *Bulletin of mathematical biology*, *5*(4), 115-133.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81-106.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-9).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778).

Johnson, R. C. (2015). Microsoft, Google beat humans at image recognition. *EE Times*.

Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 234-241). Springer International Publishing.

REFERENCES

Scott, A. J., & Knott, M. (1974). A cluster analysis method for grouping means in the analysis of variance. *Biometrics*, 507-512.

Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PloS one, 12(7), e0180944.

Bodyanskiy Y, Popov S. (2006) Neural network approach to forecasting of quasiperiodic financial time series. Eur J Oper Res. 175(3):1357–66.

Nourani V, Komasi M, Mano A. (2009) A multivariate ANN-wavelet approach for rainfall-runoff modeling. Water Resources Management. 2009;23(14):2877.

Kim TY, Oh KJ, Kim C, Do JD. (2004) Artificial neural networks for non-stationary time series. Neurocomputing.

Kim TY, Oh KJ, Kim C, Do JD. Artificial neural networks for non-stationary time series. Neurocomputing. 2004;61(C):439–47.

REFERENCES

Hsieh TJ, Hsiao HF, Yeh WC. (2011) Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm. Applied Soft Computing. 2011;11(2):2510–25. 61(C):439–47.

Sutskever I, Hinton GE (2008). Deep, narrow sigmoid belief networks are universal approximators. Neural Computation. 20(11):2629–36. pmid:18533819

.

Roux NL, Bengio Y. (2010) Deep Belief Networks Are Compact Universal Approximators. Neural Computation.22(8):2192–207.

Bengio Y, Lamblin P, Popovici D, Larochelle H. (2007) Greedy layer-wise training of deep networks. Advances in neural information processing systems. 19:153.

Ramsey JB. (1999) The contribution of wavelets to the analysis of economic and financial data. Philosophical Transactions of the Royal Society B Biological Sciences. 357(357):2593–606.

Palangi H, Ward R, Deng L. (2016) Distributed Compressive Sensing: A Deep Learning Approach. IEEE Transactions on Signal Processing. 64(17):4504–18.

Amari, S. I., Cichocki, A., & Yang, H. H. (1995, December). Recurrent neural networks for blind separation of sources. In *Proc. Int. Symp. NOLTA* (pp. 37-42).

REFERENCES

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Hochreiter S, Schmidhuber J. Long Short-Term Memory (1997). Neural Computation. 1997;9(8):1735–80. pmid:9377276

Bliemel F. Theil's (1973) Forecast Accuracy Coefficient: A Clarification. Journal of Marketing Research. 10(4):444.

AKNOWLEDGEMENTS

THANK YOU
FOR YOUR ATTENTION

Francesco Pugliese