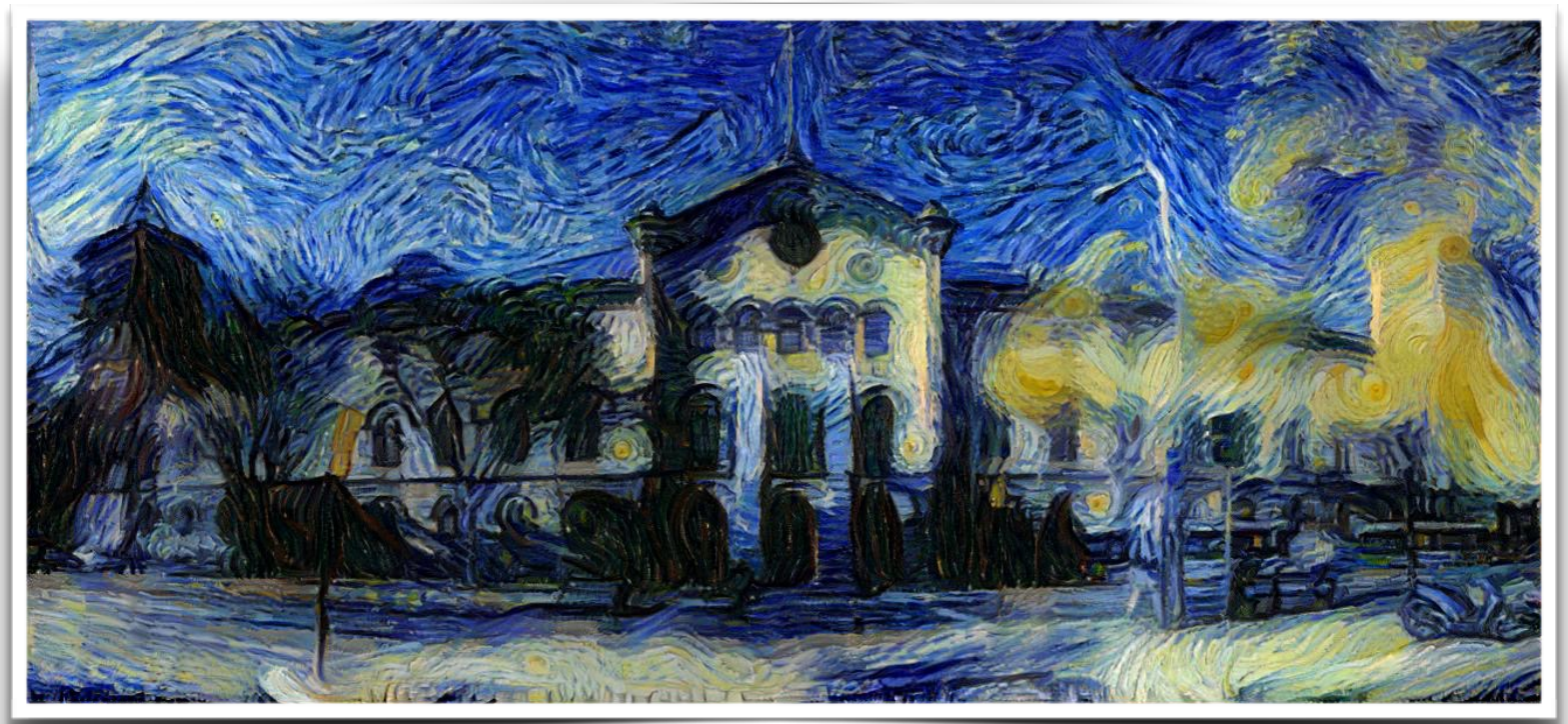
UNIVERSITAT DE BARCELONA

DATASCIENCE@UB



Deep Learning From Scratch

June 16-17, 2016, Barcelona

wifi.ub.edu

Identificador:

agnnzv.tmp

Contraseña:

mpfg47

Course Agenda

Thu, 10:00-10:10, Welcome	Fri, 09:00-11:00, Convolutions. CNN models.
Thu, 10:10-11:00, Introduction to Deep Learning and its applications. Docker. Using the Jupyter notebook. What's a neural network?	Fri, 11:00-11:15, Coffee Break
Thu, 11:00-11:15, Coffee Break	Fri, 11:15-13:30, Recurrent Neural Netwoks
Thu, 11:15-13:30, Basic Concepts: Score & Loss functions, Optimization (SGD), Linear Regression	Fri, 13:30-15:00, Lunch
Thu, 13:30-15:00, Lunch	Fri, 15:00-16:15, Unsupervised Learning
Thu, 15:00-16:15, Automated differentiation, Backpropagation, Training a Neural Netwotk from Scratch	Fri, 16:15-16:30, Coffee Break
Thu, 16:15-16:30, Coffee Break	Fri, 16:30-17:15, Advanced Applications: Neural art, colorization, music generation.
Thu, 16:30-18:00, Tensorflow programming model.	Fri, 17:15-17:30, Summary of the course and closing.

Who

DataScience@UB

Francesc Dantí, Adjunct Lecturer at UB.

Oriol Pujol, Associate Professor at UB.

Santi Seguí, Lecturer at UB.

Jordi Vitrià. Full Professor at UB.

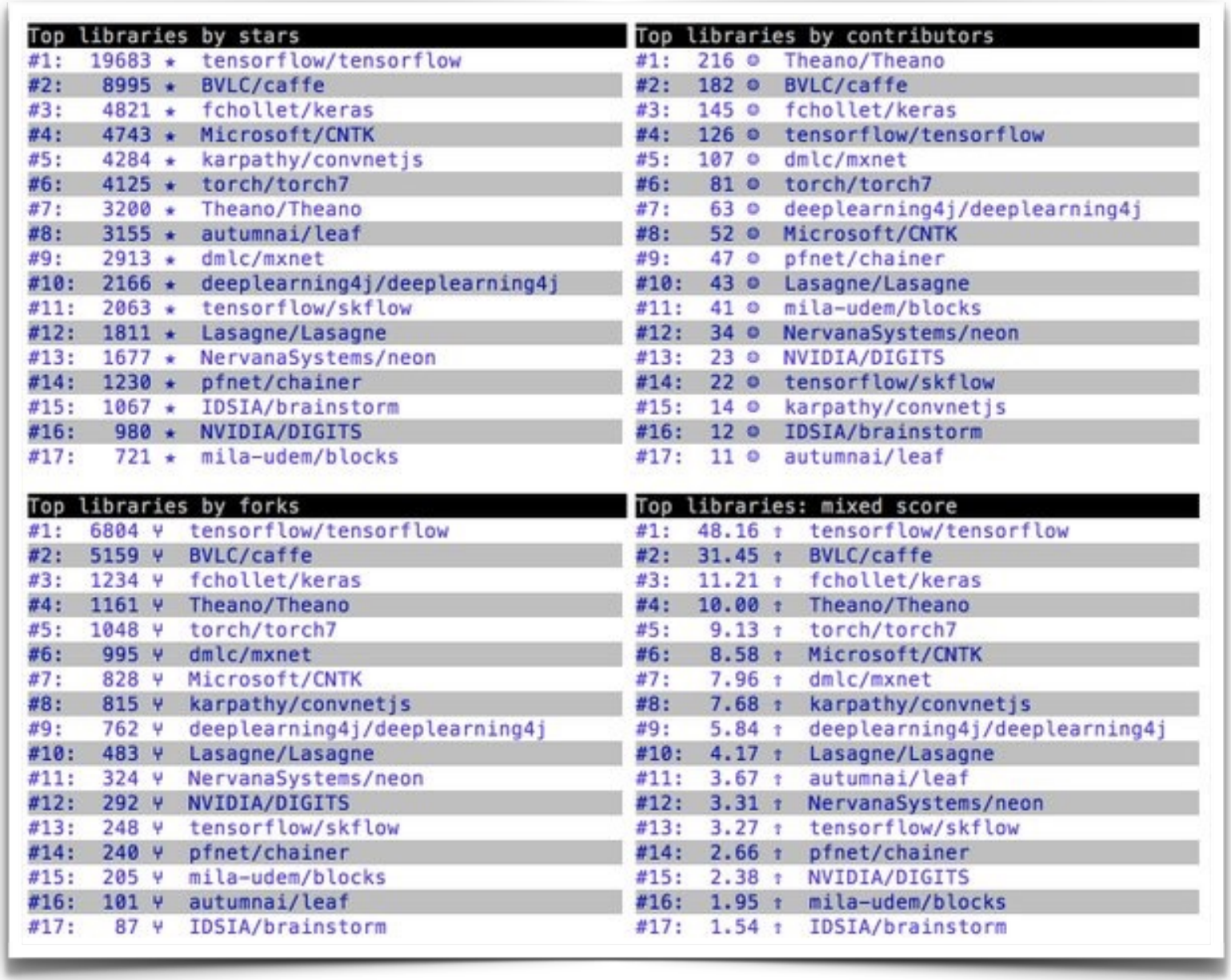
Axel Brando

Guillem Pascual

Carles Riera

What

1. Describe how a (deep) neural network works and combine different types of layers and activation functions. Deep Learning is not magic.
2. Describe how these models can be applied in computer vision, text analytics, time series analysis, etc. Deep Learning is not the final machine learning method.
3. Develop your own models in **Tensorflow** and derivates. You can train (small) deep models in your laptop.



Top libraries by stars		
#1:	19683 ★	tensorflow/tensorflow
#2:	8995 ★	BVLC/caffe
#3:	4821 ★	fchollet/keras
#4:	4743 ★	Microsoft/CNTK
#5:	4284 ★	karpathy/convnetjs
#6:	4125 ★	torch/torch7
#7:	3200 ★	Theano/Theano
#8:	3155 ★	autumnai/leaf
#9:	2913 ★	dmlc/mxnet
#10:	2166 ★	deeplearning4j/deeplearning4j
#11:	2063 ★	tensorflow/skflow
#12:	1811 ★	Lasagne/Lasagne
#13:	1677 ★	NervanaSystems/neon
#14:	1230 ★	pfnet/chainer
#15:	1067 ★	IDSIA/brainstorm
#16:	980 ★	NVIDIA/DIGITS
#17:	721 ★	mila-udem/blocks

Top libraries by contributors		
#1:	216 ⚙	Theano/Theano
#2:	182 ⚙	BVLC/caffe
#3:	145 ⚙	fchollet/keras
#4:	126 ⚙	tensorflow/tensorflow
#5:	107 ⚙	dmlc/mxnet
#6:	81 ⚙	torch/torch7
#7:	63 ⚙	deeplearning4j/deeplearning4j
#8:	52 ⚙	Microsoft/CNTK
#9:	47 ⚙	pfnet/chainer
#10:	43 ⚙	Lasagne/Lasagne
#11:	41 ⚙	mila-udem/blocks
#12:	34 ⚙	NervanaSystems/neon
#13:	23 ⚙	NVIDIA/DIGITS
#14:	22 ⚙	tensorflow/skflow
#15:	14 ⚙	karpathy/convnetjs
#16:	12 ⚙	IDSIA/brainstorm
#17:	11 ⚙	autumnai/leaf

Top libraries by forks		
#1:	6804 ⑂	tensorflow/tensorflow
#2:	5159 ⑂	BVLC/caffe
#3:	1234 ⑂	fchollet/keras
#4:	1161 ⑂	Theano/Theano
#5:	1048 ⑂	torch/torch7
#6:	995 ⑂	dmlc/mxnet
#7:	828 ⑂	Microsoft/CNTK
#8:	815 ⑂	karpathy/convnetjs
#9:	762 ⑂	deeplearning4j/deeplearning4j
#10:	483 ⑂	Lasagne/Lasagne
#11:	324 ⑂	NervanaSystems/neon
#12:	292 ⑂	NVIDIA/DIGITS
#13:	248 ⑂	tensorflow/skflow
#14:	240 ⑂	pfnet/chainer
#15:	205 ⑂	mila-udem/blocks
#16:	101 ⑂	autumnai/leaf
#17:	87 ⑂	IDSIA/brainstorm

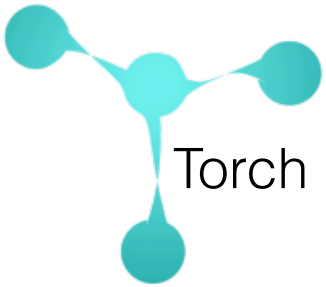
Top libraries: mixed score		
#1:	48.16 ↑	tensorflow/tensorflow
#2:	31.45 ↑	BVLC/caffe
#3:	11.21 ↑	fchollet/keras
#4:	10.00 ↑	Theano/Theano
#5:	9.13 ↑	torch/torch7
#6:	8.58 ↑	Microsoft/CNTK
#7:	7.96 ↑	dmlc/mxnet
#8:	7.68 ↑	karpathy/convnetjs
#9:	5.84 ↑	deeplearning4j/deeplearning4j
#10:	4.17 ↑	Lasagne/Lasagne
#11:	3.67 ↑	autumnai/leaf
#12:	3.31 ↑	NervanaSystems/neon
#13:	3.27 ↑	tensorflow/skflow
#14:	2.66 ↑	pfnet/chainer
#15:	2.38 ↑	NVIDIA/DIGITS
#16:	1.95 ↑	mila-udem/blocks
#17:	1.54 ↑	IDSIA/brainstorm

BackEnds

Python, C++

MultiGPU

Distributed



Lua

MultiGPU

theano

Python

Large amount of sample code

Caffe

Python, C++

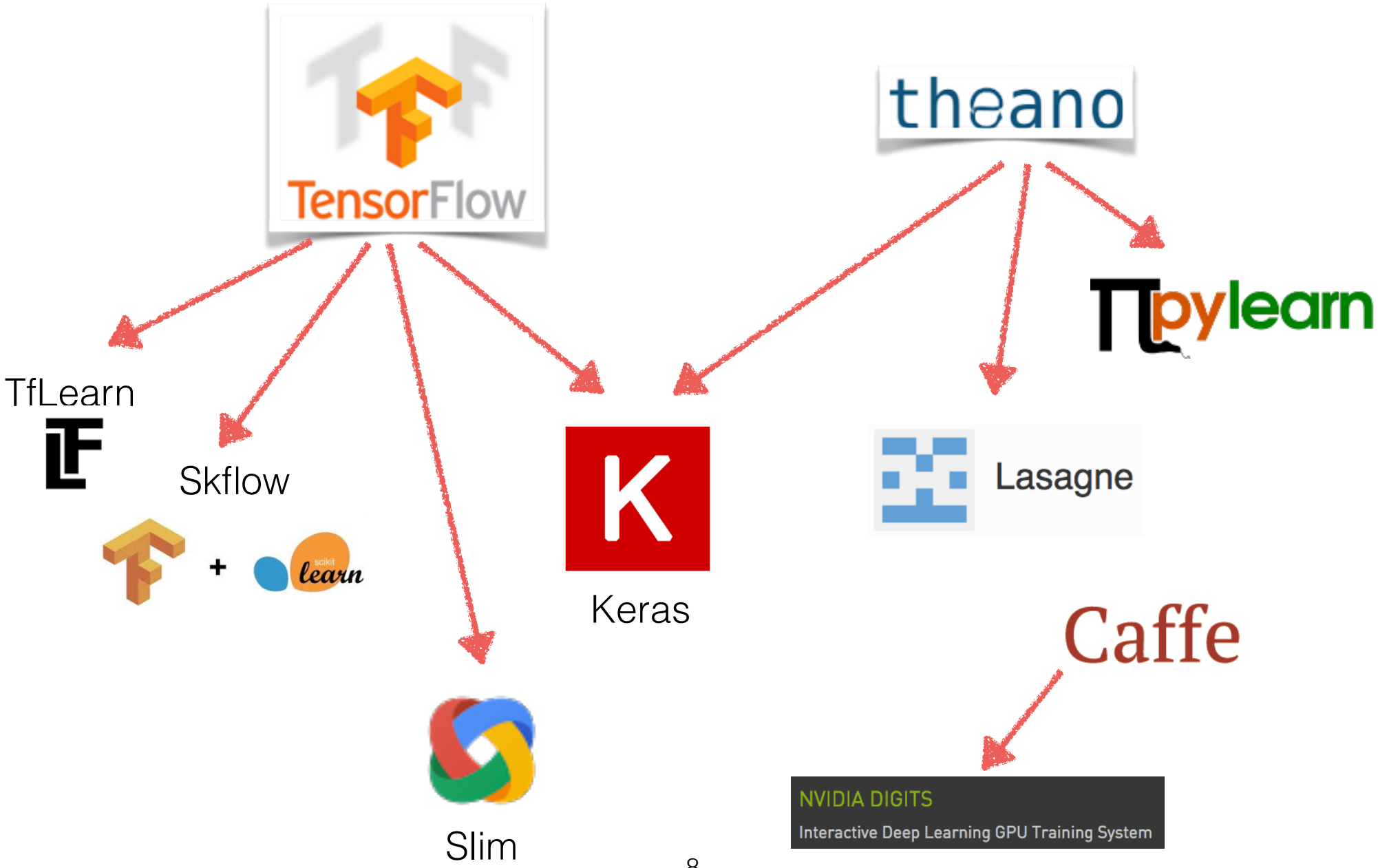
MultiGPU

Google

Université de Montréal

Berkeley
UNIVERSITY OF CALIFORNIA

Tools


TensorFlow
theano
pylearn
TfLearn
Skflow
scikit learn
Keras
Lasagne
Caffe
Slim
NVIDIA DIGITS
Interactive Deep Learning GPU Training System

Other BackEnds

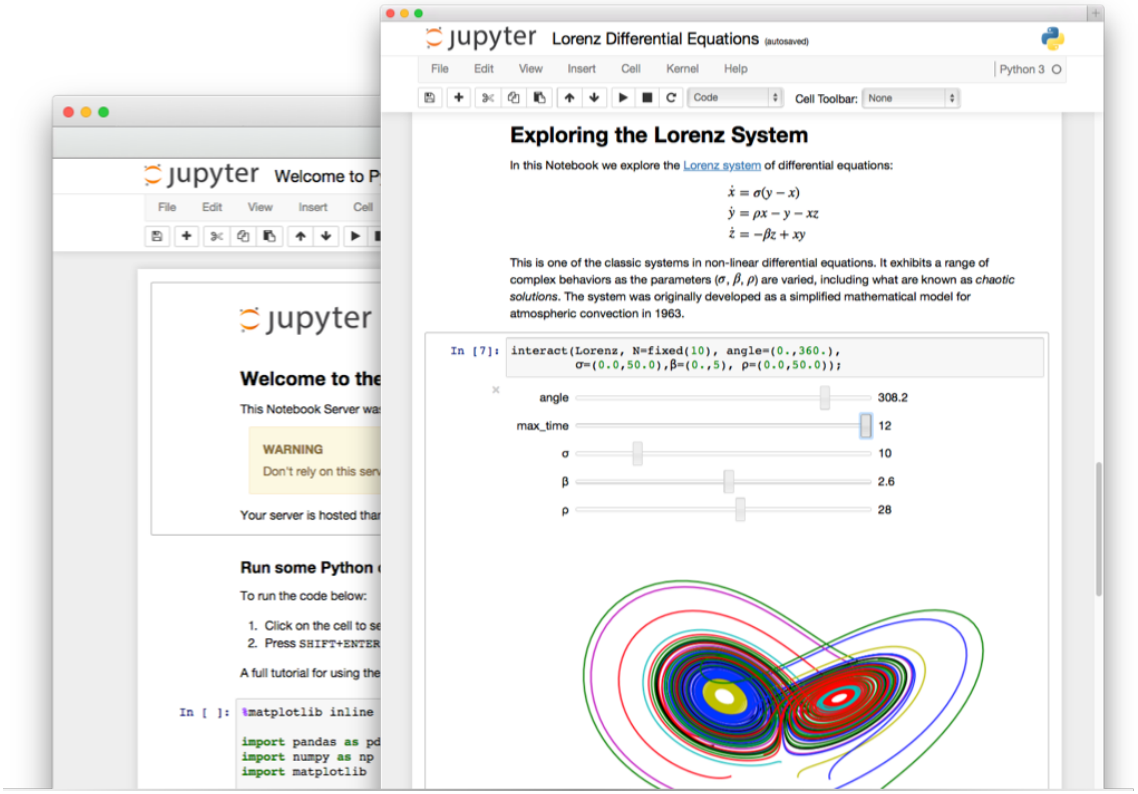
Neon

ConvNetJS
Deep Learning in your browser

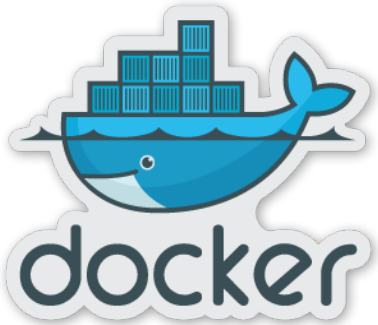
Approach

We will illustrate all contents with Jupyter notebooks, a web application that allows you to create and share documents that contain live code, equations, visualizations and explanatory text.

Skflow



Approach



We will use a **Jupyter-driven Docker Container.**

Docker provides the ability to build a runtime environment that not only remains isolated from other running containers, but also can be deployed to multiple locations in a repeatable way. Docker also uses a text document – a Dockerfile – that contains all the commands to assemble an image, which will meet our need to document the build environment. Finally, Docker's runtime options enable us to attach GPU devices when deploying on remote servers.

```
docker pull deepub/deepub
docker run -p 8888:8888 deepub/deepub
```

```
docker ps
docker rm -fv CONTAINER_ID
```

Keep in mind that the changes made inside Docker container are not persistent.

Why Deep Learning?

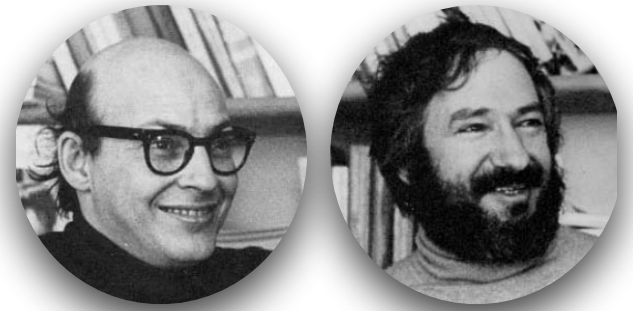

It’s funny!
It’s not rocket science!
It’s powerful!

- In 1943, neurophysiologist **Warren McCulloch** and mathematician **Walter Pitts** wrote a paper on how neurons might work. In order to describe how neurons in the brain might work, they modeled a simple neural network using electrical circuits.

- In 1949, Donald **Hebb** wrote *The Organization of Behavior*, a work which pointed out the fact that neural pathways are strengthened each time they are used, a concept fundamentally essential to the ways in which humans learn. If two nerves fire at the same time, he argued, the connection between them is enhanced.

- In 1957 **Frank Rosenblatt** attempted to build a kind of mechanical brain called the Perceptron, which was billed as “a machine which senses, recognizes, remembers, and responds like the human mind”.

- In 1962, **Widrow & Hoff** developed a learning procedure that examines the value before the weight adjusts it (i.e. 0 or 1) according to the rule: Weight Change = (Pre-Weight line value) * (Error / (Number of Inputs)). It is based on the idea that while one active perceptron may have a big error, one can adjust the weight values to distribute it across the network, or at least to adjacent perceptrons.



- A critical book written in 1969 by **Marvin Minsky** and his collaborator **Seymour Papert** showed that Rosenblatt's original system was painfully limited, literally blind to some simple logical functions like "exclusive-or" (As in, you can have the cake or the pie, but not both). What had become known as the field of "neural networks" all but disappeared.

First neural network winter is coming

- In 1982, interest in the field was renewed. **John Hopfield** of Caltech presented a paper to the National Academy of Sciences. His approach was to create more useful machines by using bidirectional lines. Previously, the connections between neurons was only one way.

- In 1986, the problem was how to extend the Widrow-Hoff rule to multiple layers. Three independent groups of researchers, which included **David E. Rumelhart, Geoffrey E. Hinton** and **Ronald J. Williams**, came up with similar ideas which are now called back propagation networks because it distributes pattern recognition errors throughout the network.

- From 1986 to mid 90's new developments arised: convolutional neural networks (**Y.LeCun**), unsupervised learning (**Y.Bengio**), RBM (**G.Hinton**), etc. But, by this point **new machine learning methods** had begun to also emerge, and people were again beginning to be skeptical of neural nets since they seemed so intuition-based and since computers were still barely able to meet their computational needs.

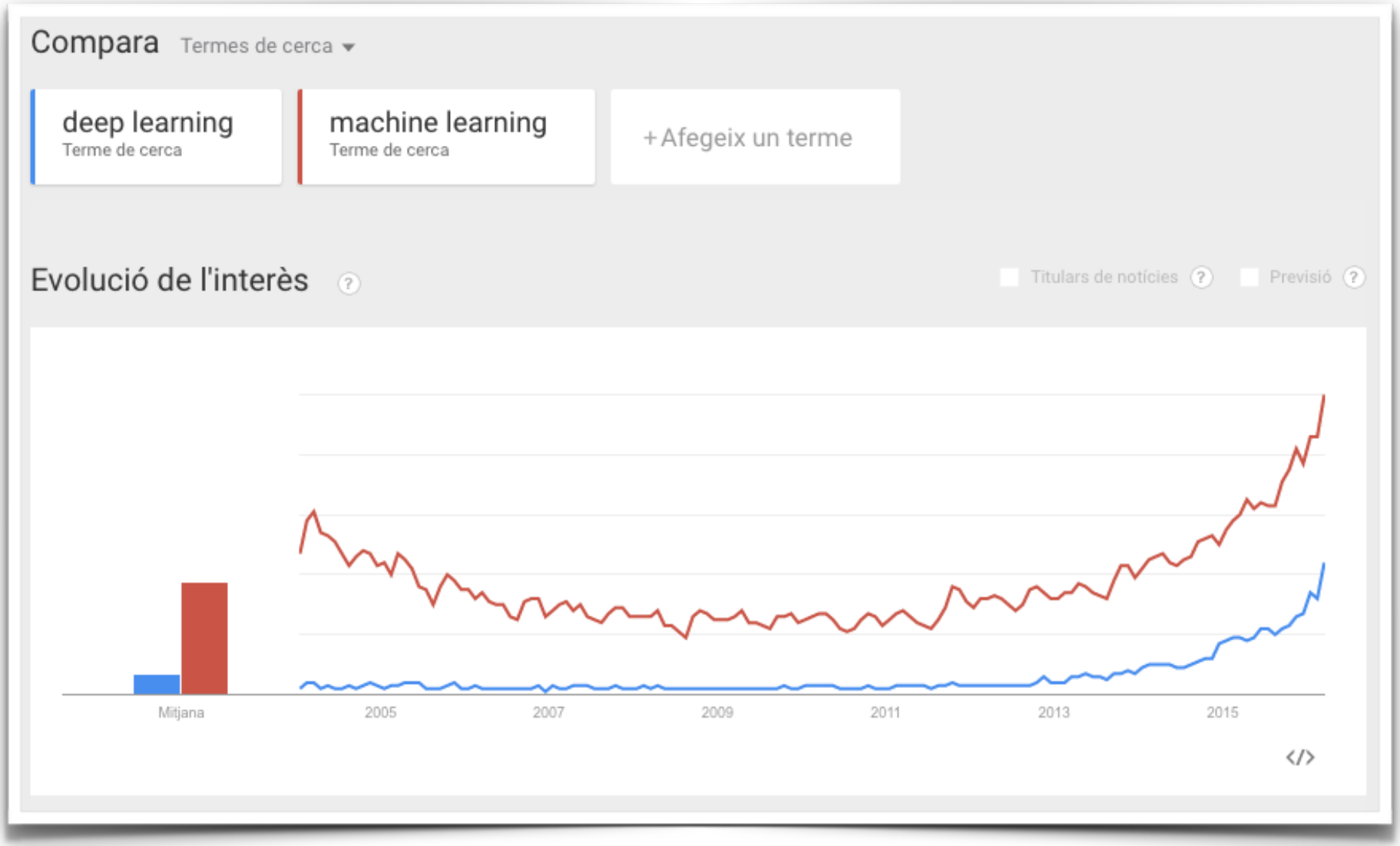

Second neural network winter is coming

- With the ascent of Support Vector Machines and the failure of backpropagation, the early 2000s were a dark time for neural net research.

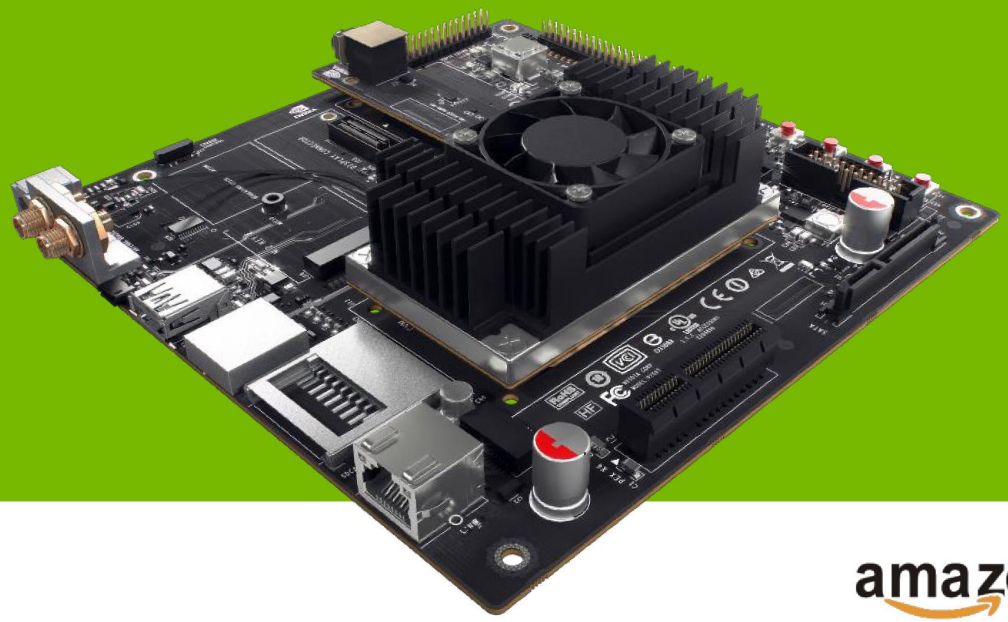
- Then, what every researcher must dream of actually happened: G.Hinton, S.Osindero, and Y.W.Teh published a paper in 2006 that was seen as a breakthrough, a breakthrough significant enough to rekindle interest in neural nets: *A fast learning algorithm for* ***deep*** *belief nets.*

- After that, following Moore's law, computers got dozens of times faster (GPUs) since the slow days of the 90s, making learning with large datasets and many layers much more tractable.

Neural Networks Reborn



Google Trends



Jetson TX1 Developer Kit

$599 retail

$299 edu

Pre-order Nov 12

Shipping Nov 16 (US)

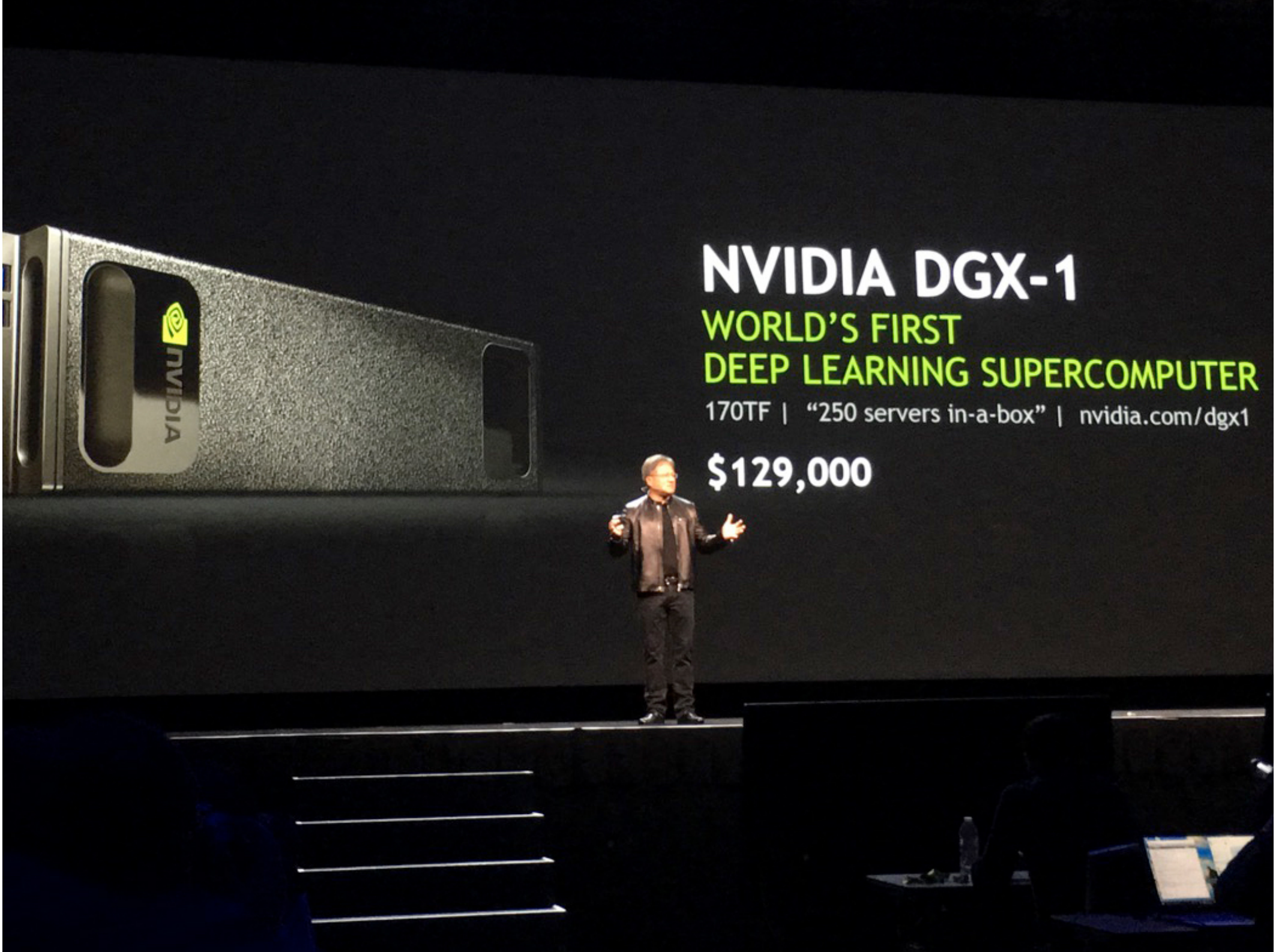
Intl to follow

amazon

newegg.com

MICRO CENTER computers & electronics

NVIDIA

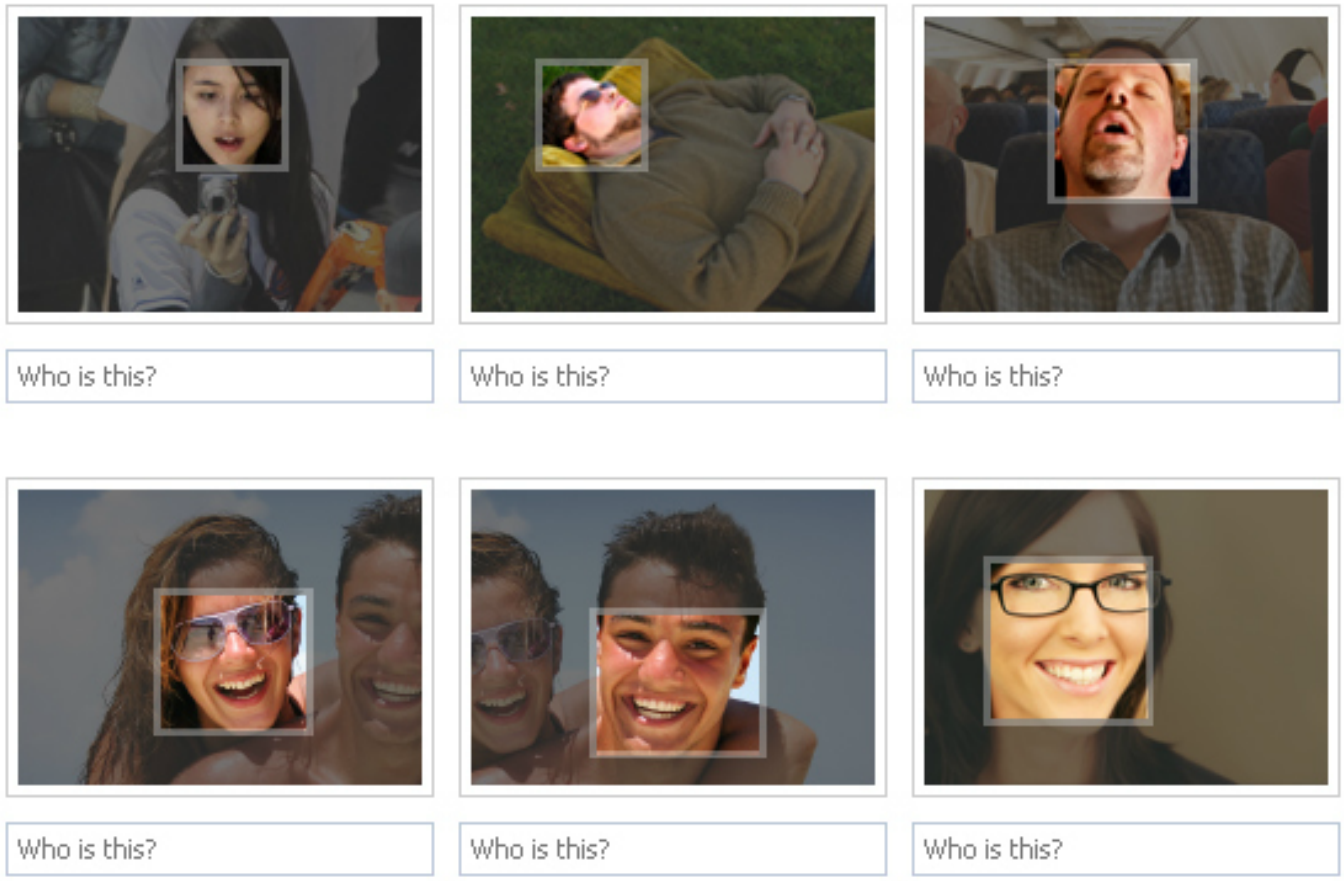

NVIDIA DGX-1
WORLD'S FIRST
DEEP LEARNING SUPERCOMPUTER
170TF | "250 servers in-a-box" | nvidia.com/dgx1
$129,000
NVIDIA

Definitions

- **Neural Networks (NN)** is a beautiful biologically-inspired programming paradigm which enables a computer to learn from observational data.
- **Deep Learning (DL)** is a powerful set of techniques for learning in neural networks.
- NN and DL currently provide the best solutions to many problems in image recognition, speech recognition, and natural language processing.

“Classical” applications:
object classification, detection and segmentation.

Face recognition.



DeepFace (Facebook): Accuracy of 97.35%

New applications: navigation and mapping.

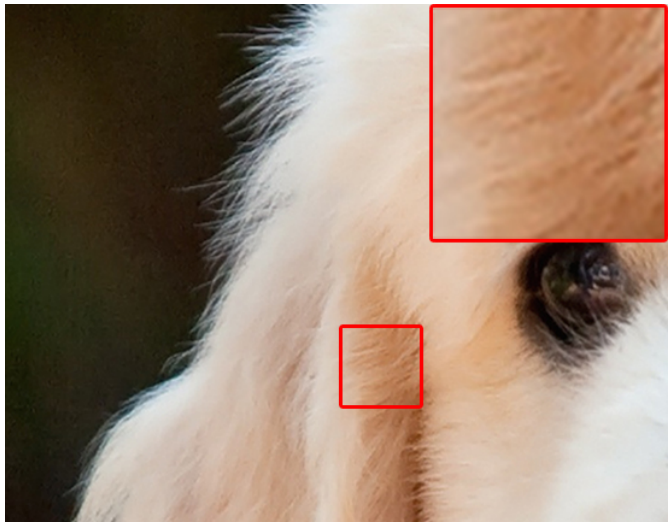
New applications: Image Upscaling (Flipboard)

http://engineering.flipboard.com/2015/05/scaling-convnets/

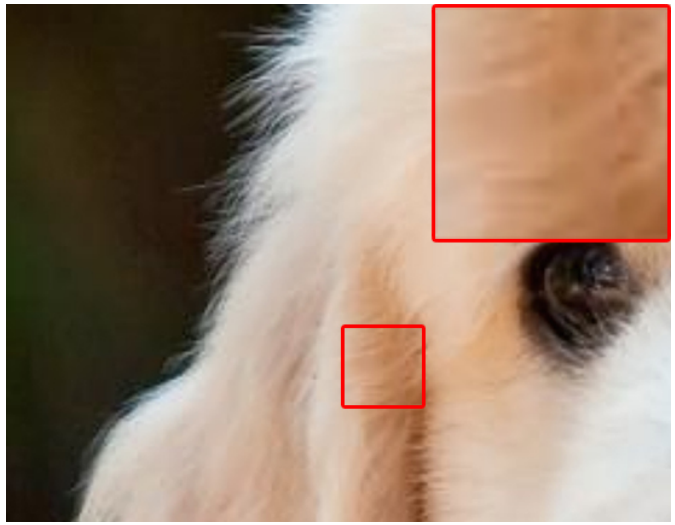
New applications: Image Upscaling (Flipboard)

http://engineering.flipboard.com/2015/05/scaling-convnets/

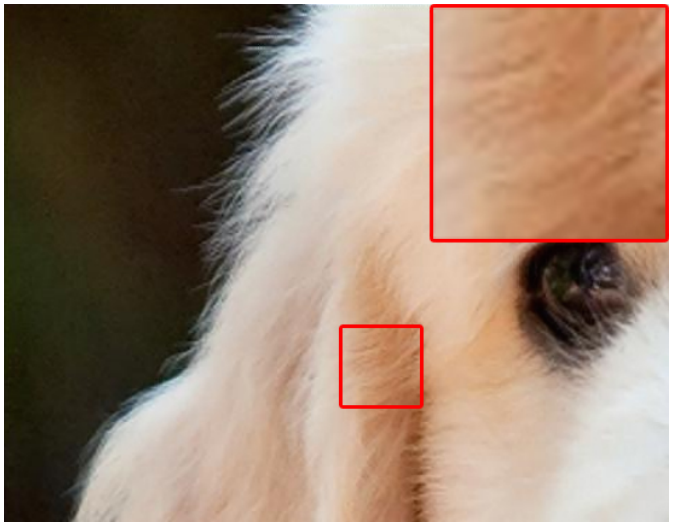
New applications: Image Upscaling (Flipboard)



Original



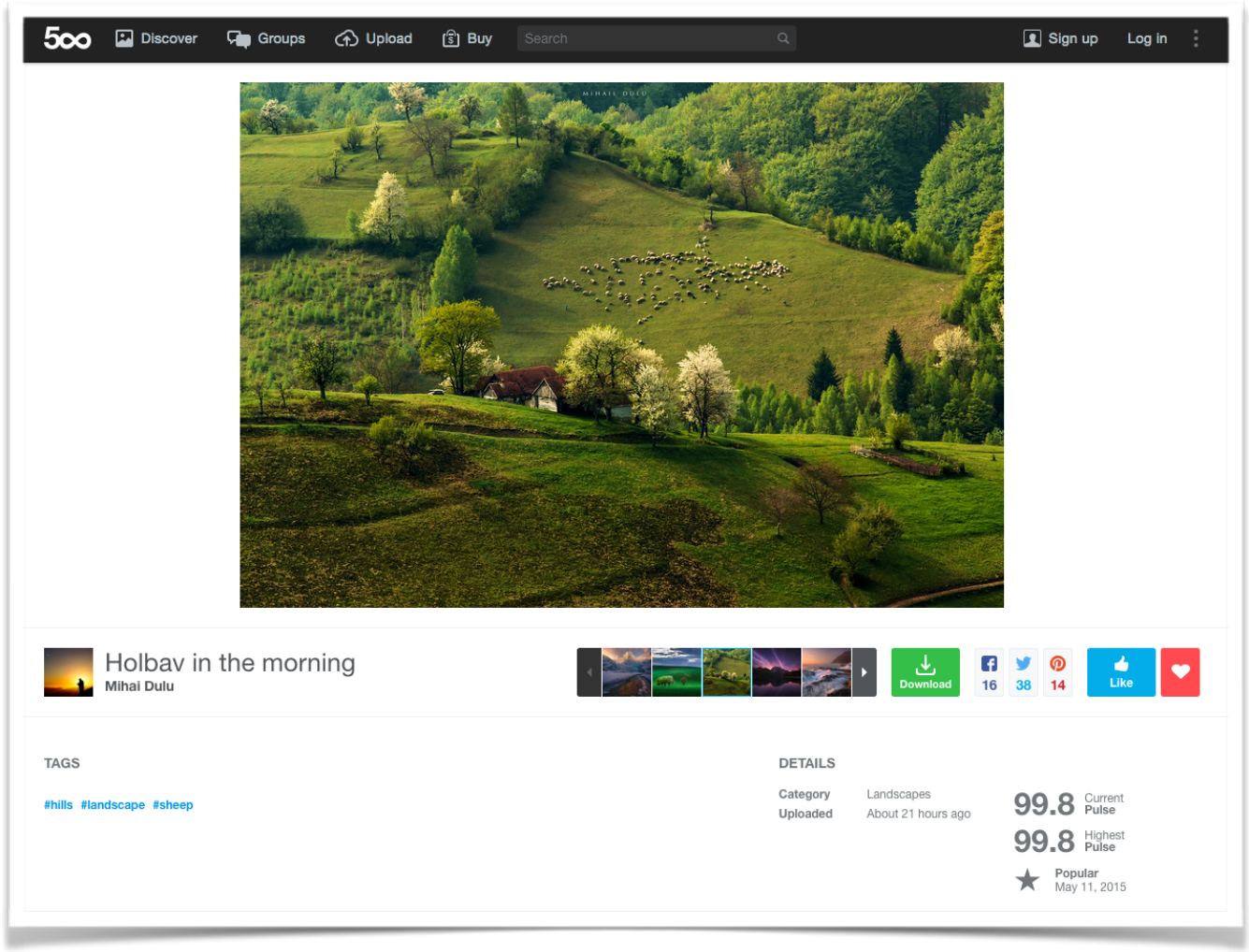
Bicubic



Model

http://engineering.flipboard.com/2015/05/scaling-convnets/

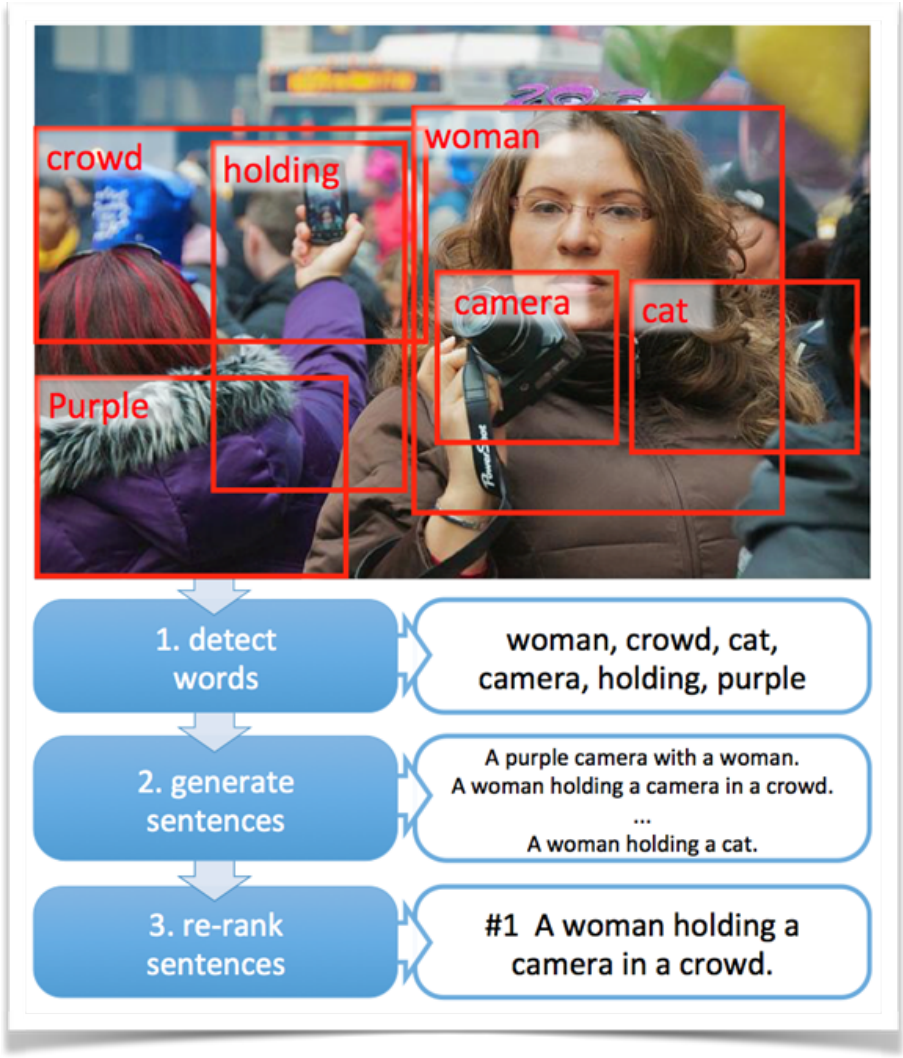
New applications: Non visual data prediction



What is Pulse?

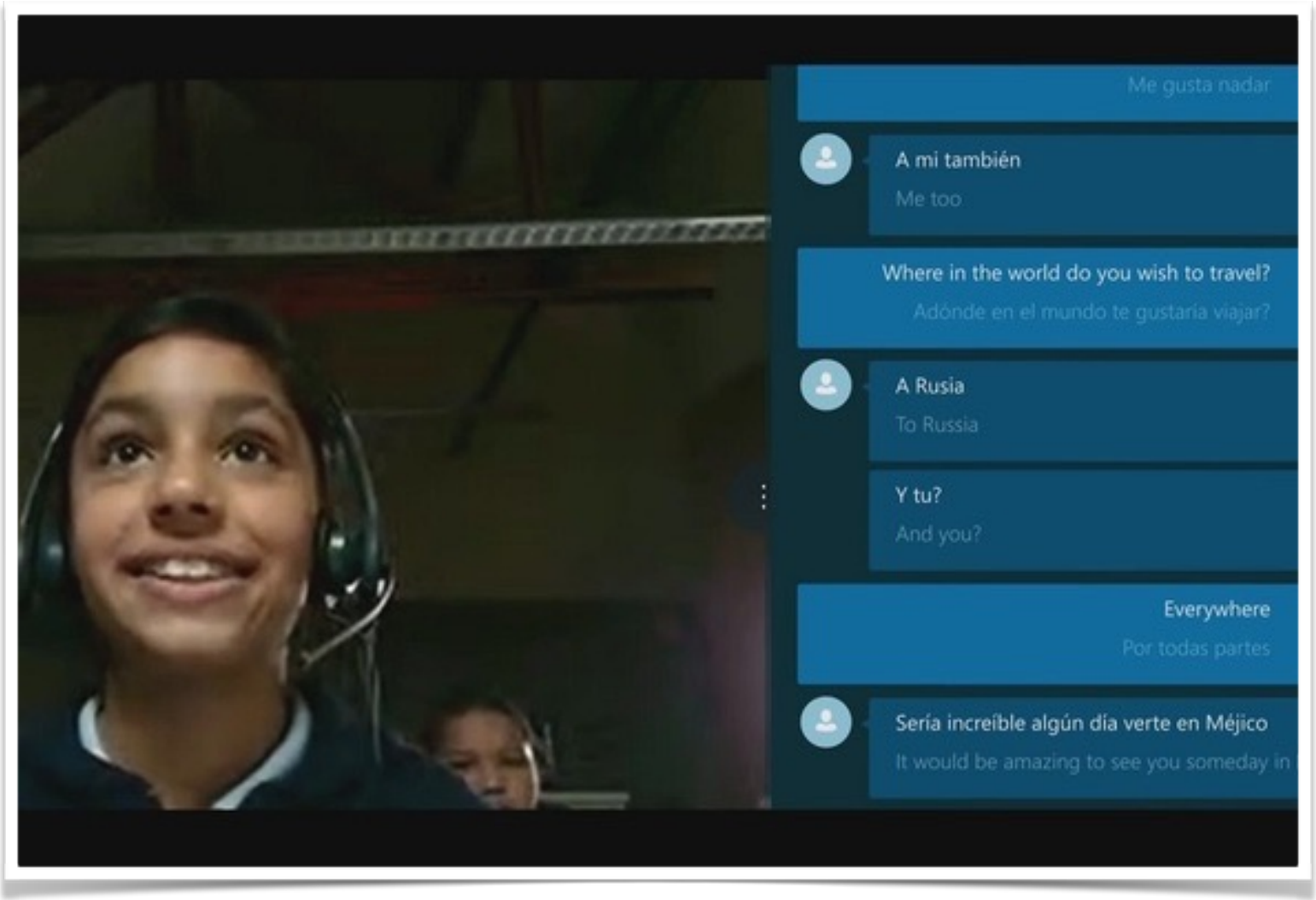
Pulse is a score out of 100 points that measures how **popular** a photo is. Pulse is calculated by an algorithm, which is unique to 500px and is based on votes (Likes & Favorites) on your photo from the community. The Pulse algorithm was designed to promote daily exposure of new photographs and photographers. It is not necessarily a measure of photograph's quality.

New applications: Automatic Image Captioning

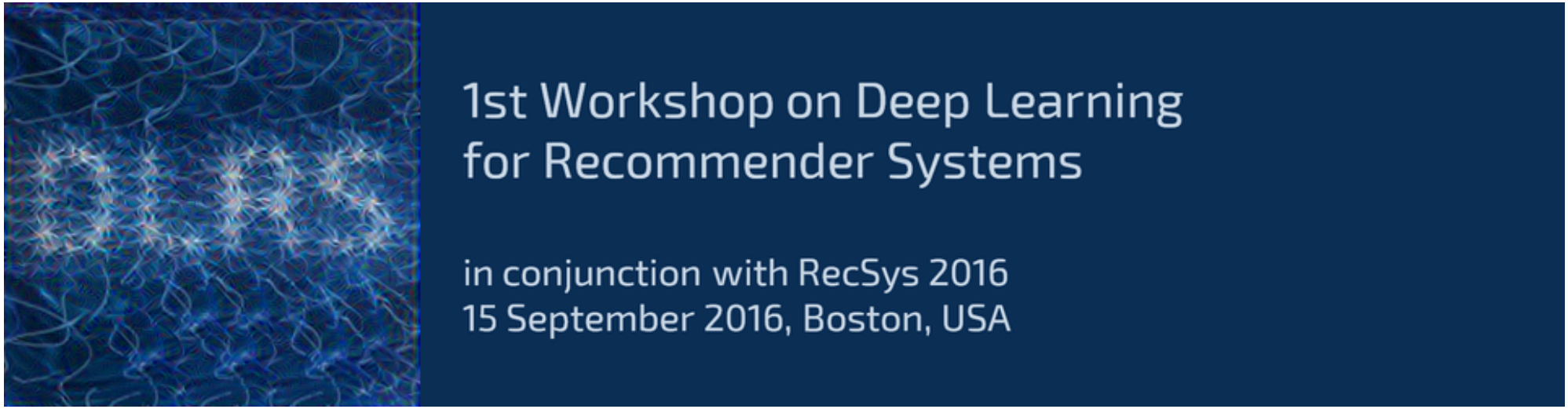


http://blogs.technet.com/b/machinelearning/archive/2014/11/18/rapid-progress-in-automatic-image-captioning.aspx

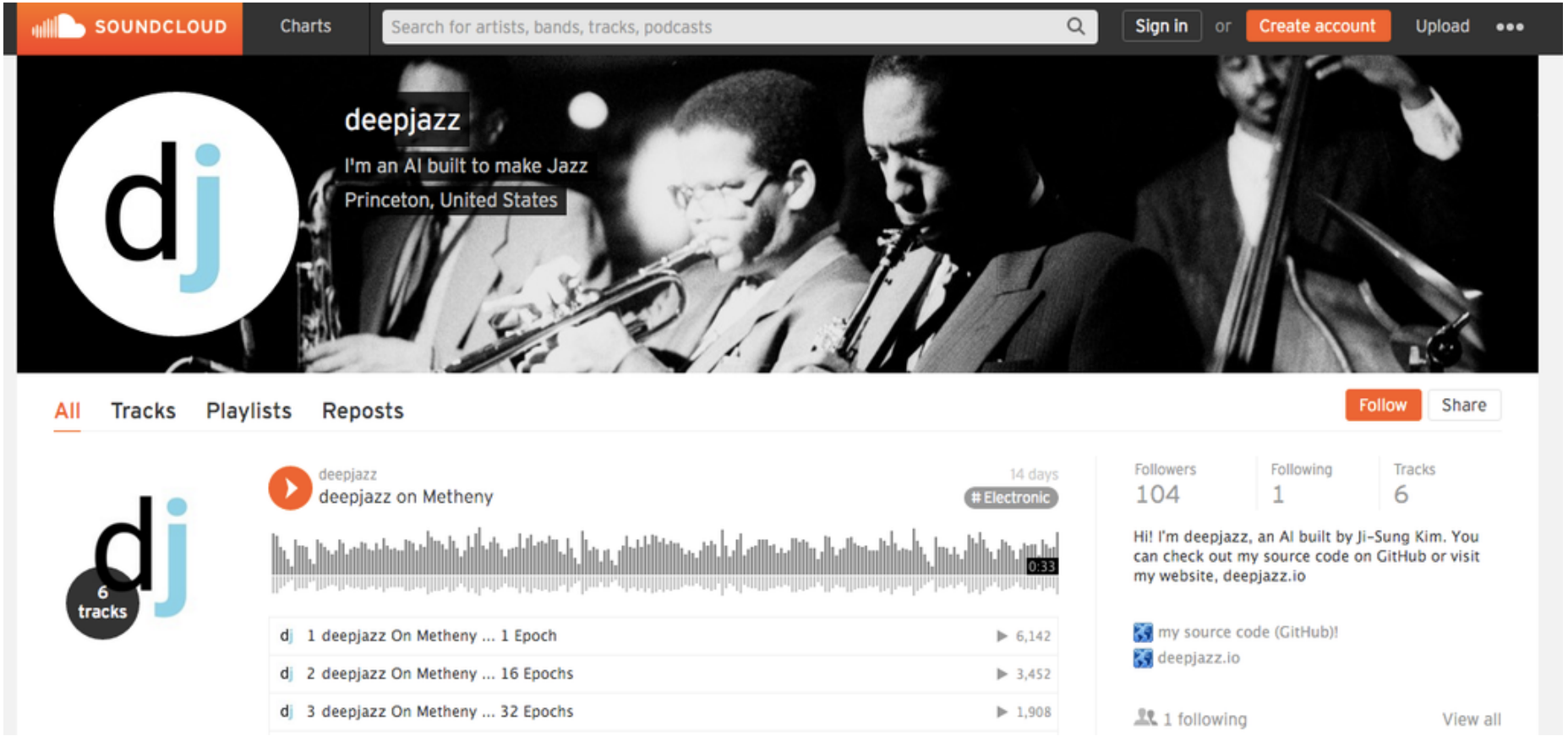
Speech translation



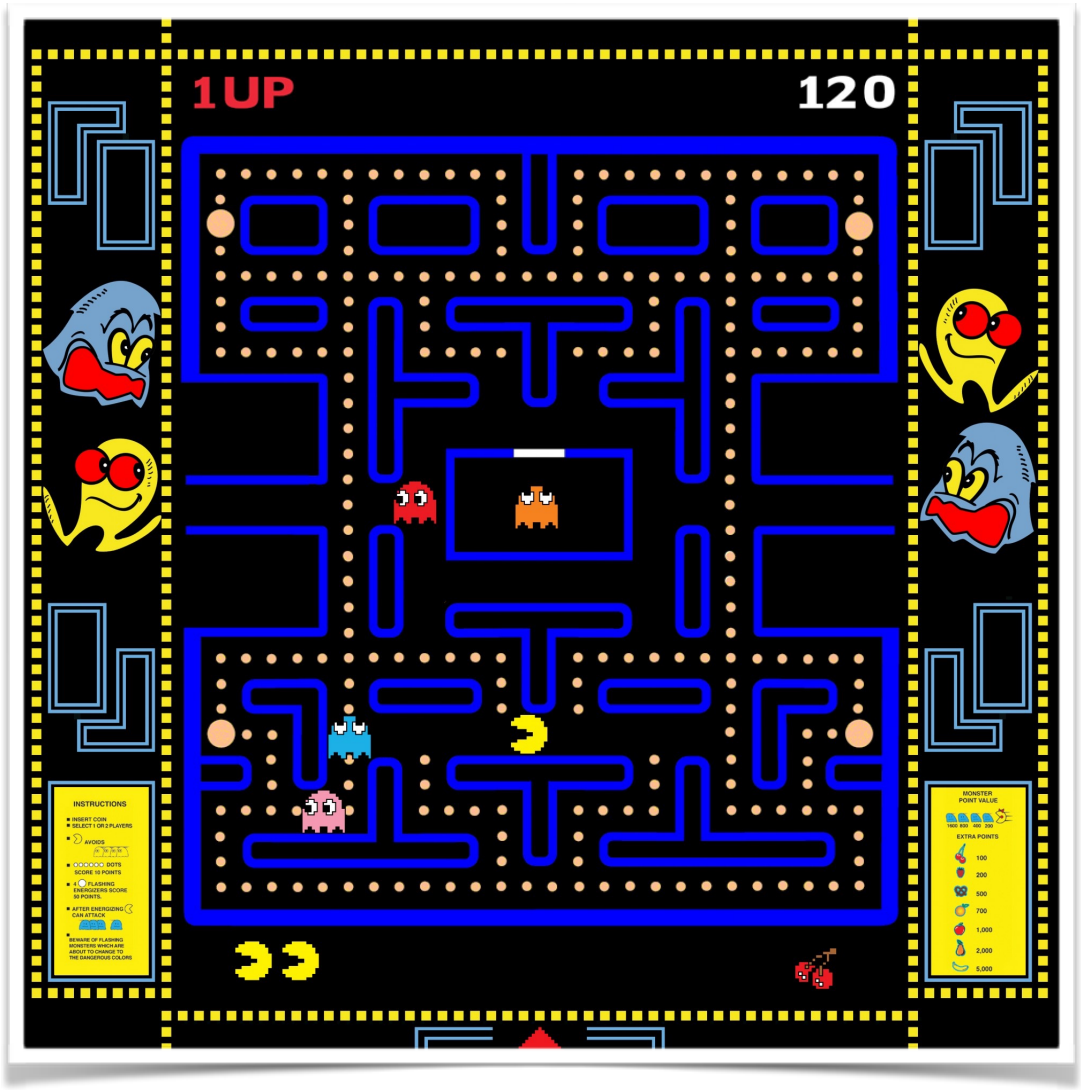
Recommenders



Music Generation



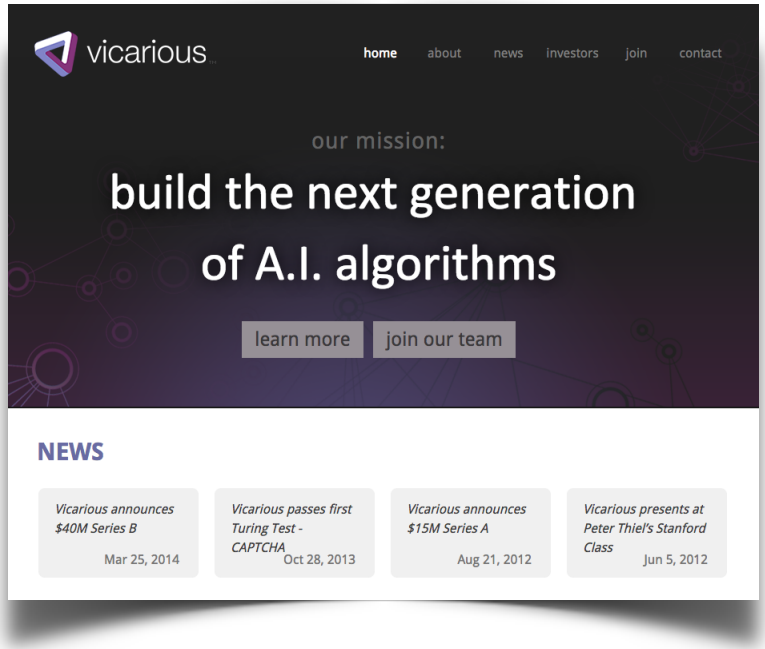
Reinforcement learning.



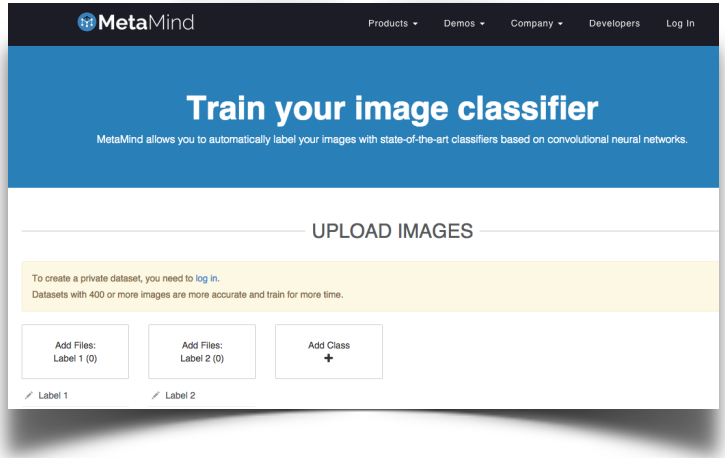
Go

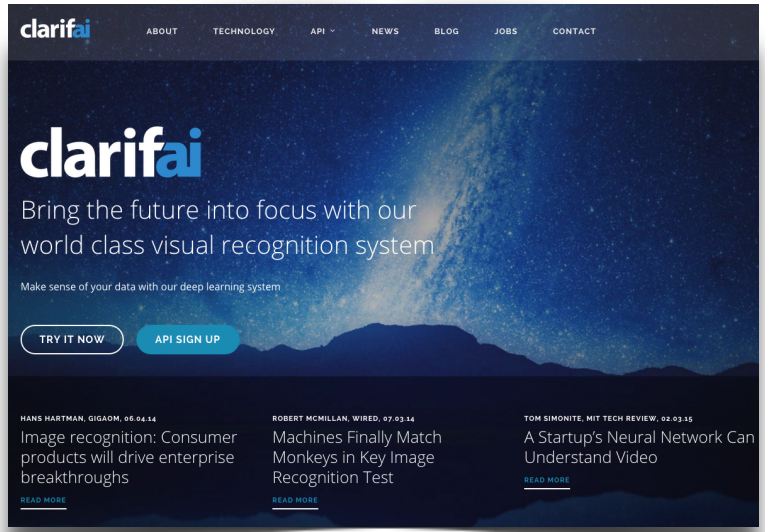

nature
THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE
At last — a computer program that can beat a champion Go player PAGE 484
ALL SYSTEMS GO
CONSERVATION
SONGBIRDS À LA CARTE
Illegal harvest of millions of Mediterranean birds
PAGE 452
RESEARCH ETHICS
SAFEGUARD TRANSPARENCY
Don't let openness backfire on individuals
PAGE 459
POPULAR SCIENCE
WHEN GENES GOT 'SELFISH'
Dawkins's calling card forty years on
PAGE 462
NATURE.COM/NATURE
28 January 2016 £10
Vol. 529, No. 7587
04
9 770028 083095

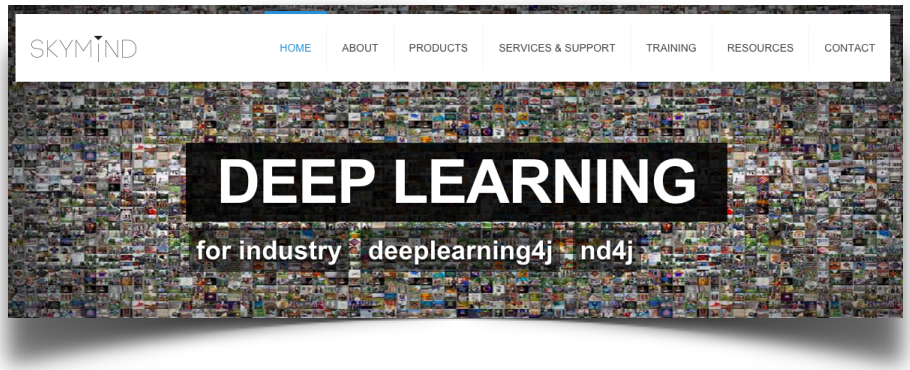
Start Ups

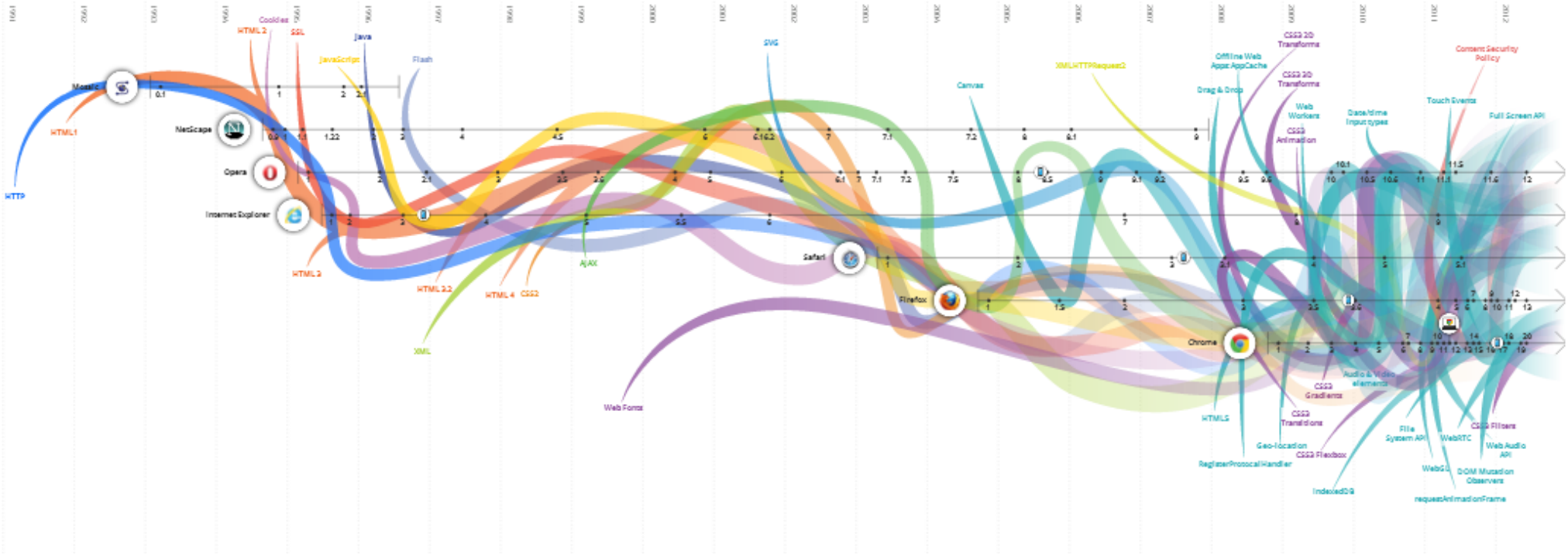

vicarious
home
about
news
investors
join
contact
our mission:
build the next generation of A.I. algorithms
learn more
join our team
NEWS
Vicarious announces $40M Series B
Mar 25, 2014
Vicarious passes first Turing Test - CAPTCHA
Oct 28, 2013
Vicarious announces $15M Series A
Aug 21, 2012
Vicarious presents at Peter Thiel's Stanford Class
Jun 5, 2012


Dato
Create Intelligence
PRODUCTS
USES
LEARN
EVENTS
COMPANY
BLOG
DOWNLOAD
MACHINE LEARNING
that scales with your business
DOWNLOAD GRAPHLAB CREATE
Ultra-Fast Data Analytics
Wrangle terabytes of data at interactive speeds, even on your laptop. With SFrames, the most scalable data structure, optimized for machine learning, large-scale data transformations and feature engineering are easy now.
Best-In-Class Predictive Modeling
Build your models with the best applied predictive technologies and machine learning algorithms, including Python scikit-learn and GraphLab Create™. Model selection, parameter search, metrics and visualization are easy now.
Production-Ready Data Science
Make your predictive models available as RESTful services, on AWS or on-premises. Integrate with Hadoop and Spark for distributed execution. Deploying distributed learning and predictive services are easy now.


MetaMind
Products
Demos
Company
Developers
Log In
Train your image classifier
MetaMind allows you to automatically label your images with state-of-the-art classifiers based on convolutional neural networks.
UPLOAD IMAGES
To create a private dataset, you need to log in.
Datasets with 400 or more images are more accurate and train for more time.
Add Files: Label 1 (0)
Add Files: Label 2 (0)
Add Class +
Label 1
Label 2


clarifai
ABOUT
TECHNOLOGY
API
NEWS
BLOG
JOBS
CONTACT
clarifai
Bring the future into focus with our world class visual recognition system
Make sense of your data with our deep learning system
TRY IT NOW
API SIGN UP
HANS HARTMAN, GIGAOM, 06.04.14
Image recognition: Consumer products will drive enterprise breakthroughs
READ MORE
ROBERT MCMILLAN, WIRED, 07.03.14
Machines Finally Match Monkeys in Key Image Recognition Test
READ MORE
TOM SIMONITE, MIT TECH REVIEW, 02.03.15
A Startup's Neural Network Can Understand Video
READ MORE


SKYMIND
HOME
ABOUT
PRODUCTS
SERVICES & SUPPORT
TRAINING
RESOURCES
CONTACT
DEEP LEARNING
for industry
deeplearning4j
nd4j



What is Deep Learning?

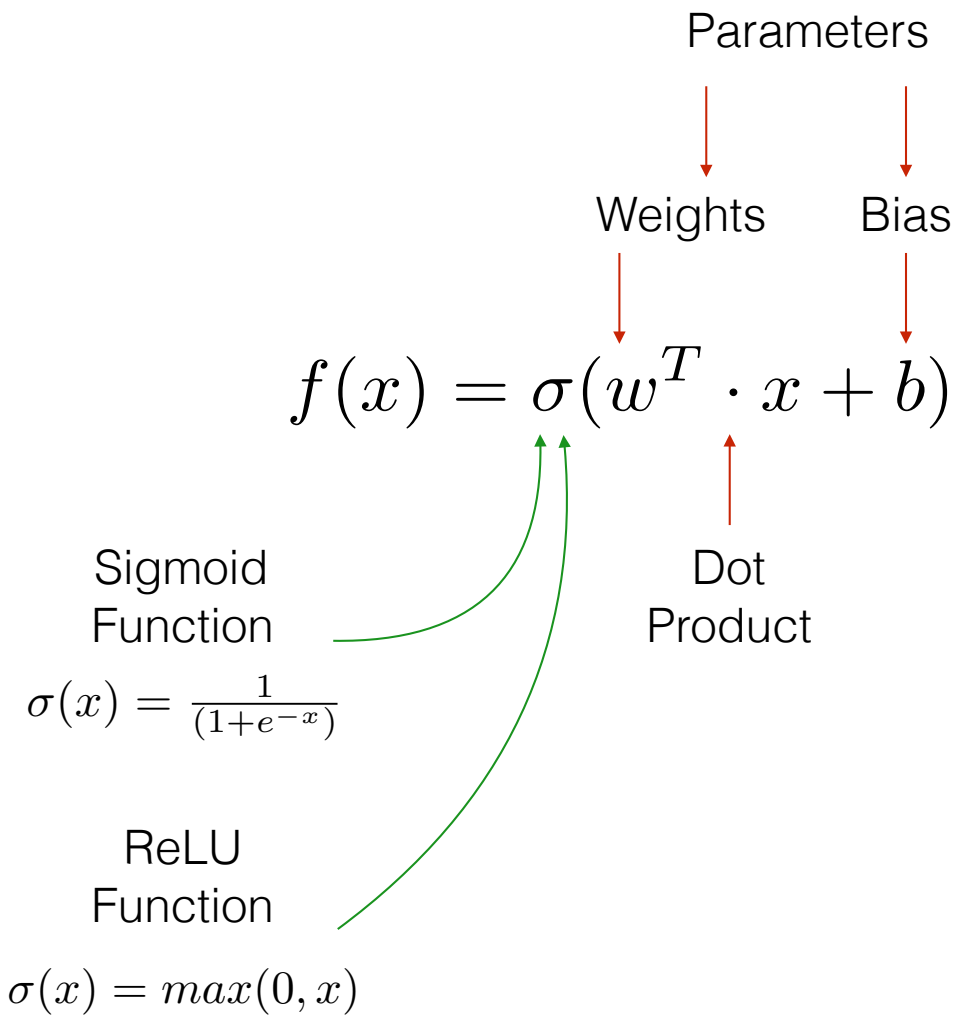
Learning from Data

Training data: a set of $(x^{(m)}, y^{(m)})$ pairs.
Learn a function $f_w : x \to y$ to predict on new inputs x.

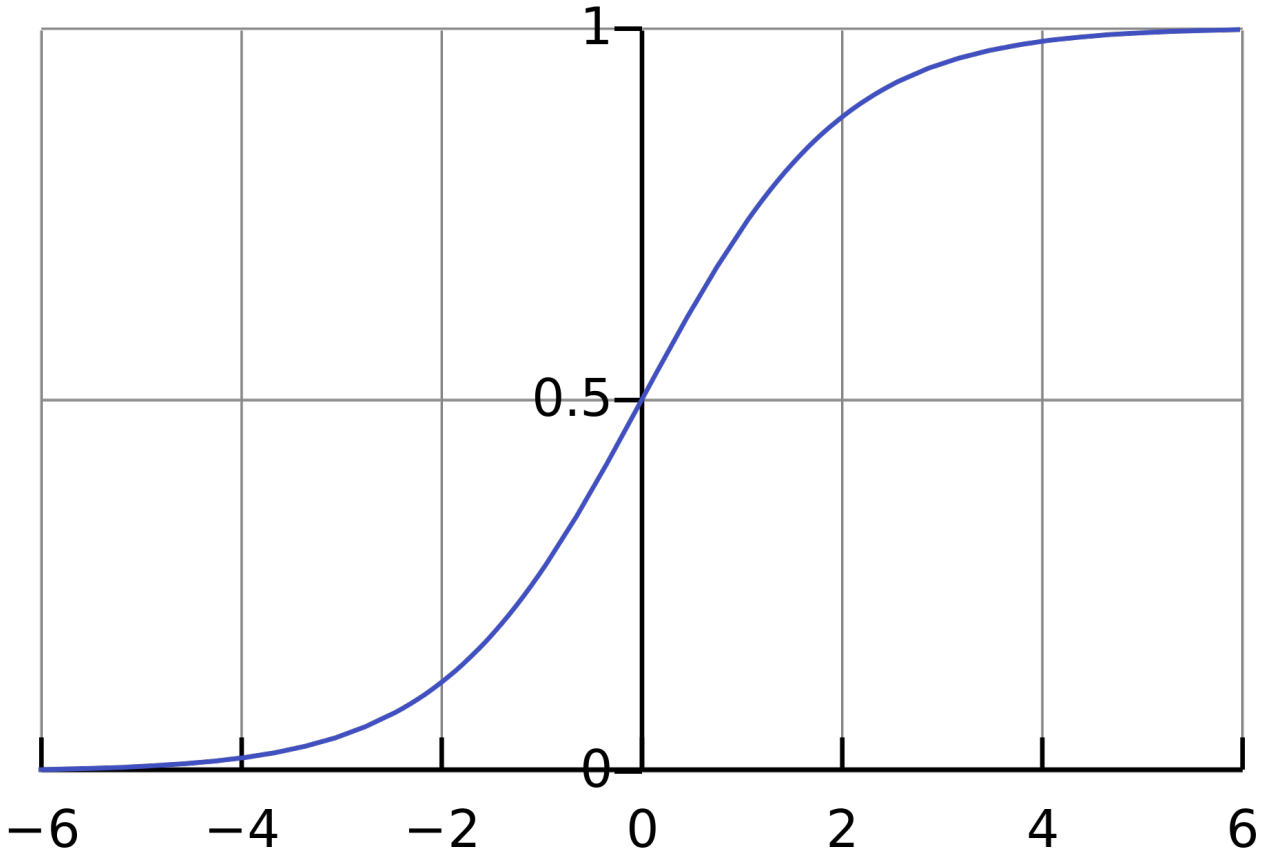
1. Choose a model function family f_w.

2. Optimize parameters w.

1-layer neural net model

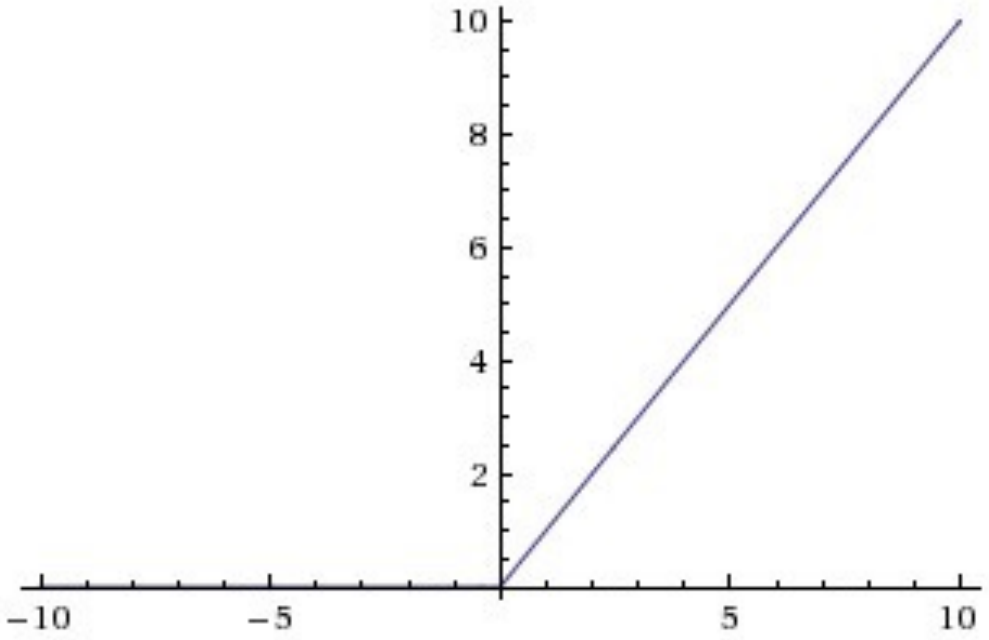


1-layer neural net model



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

1-layer neural net model

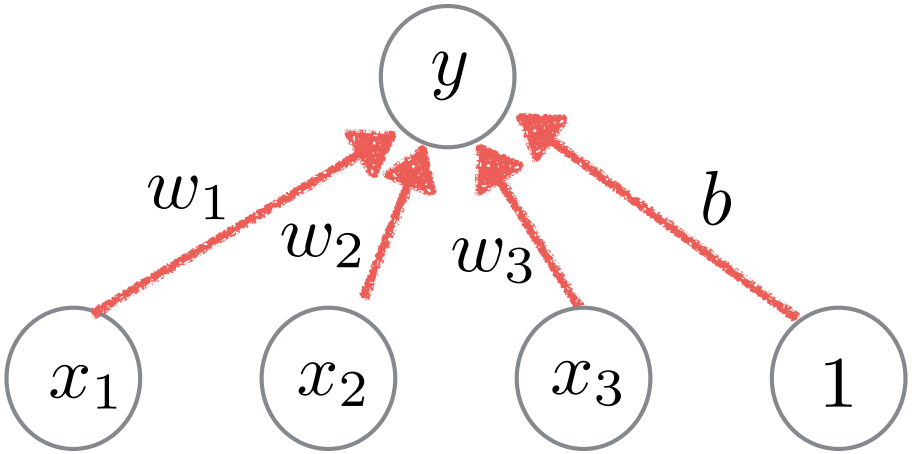


$$\sigma(x) = max(0, x)$$

Table 3: Non-linearities tested.

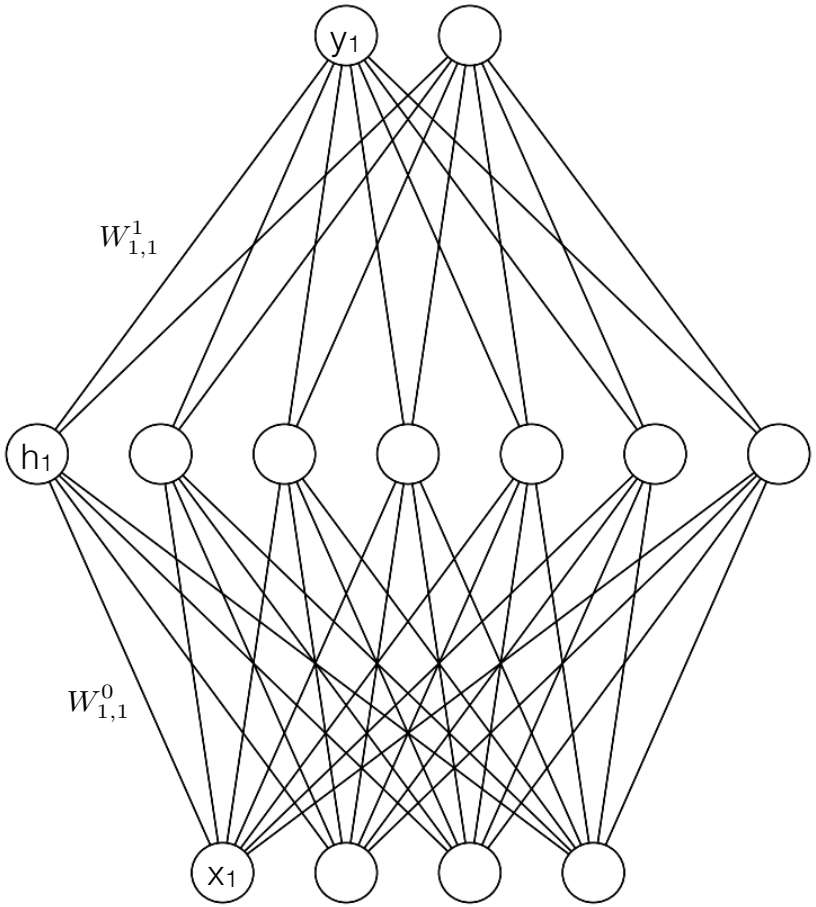
Name	**Formula**	**Year**
none	$y = x$	-
sigmoid	$y = \frac{1}{1+e^{-x}}$	1986
tanh	$y = \frac{e^{2x}-1}{e^{2x}+1}$	1986
ReLU	$y = \max(x, 0)$	2010
(centered) SoftPlus	$y = \ln(e^x + 1) - \ln 2$	2011
LReLU	$y = \max(x, \alpha x)$, $\alpha \approx 0.01$	2011
maxout	$y = \max(W_1 x + b_1, W_2 x + b_2)$	2013
APL	$y = \max(x,0) + \sum_{s=1}^{S} a_i^s \max(0, -x + b_i^s)$	2014
VLReLU	$y = \max(x, \alpha x)$, $\alpha \in 0.1, 0.5$	2014
RReLU	$y = \max(x, \alpha x)$, $\alpha = \text{random}(0.1, 0.5)$	2015
PReLU	$y = \max(x, \alpha x)$, α is learnable	2015
ELU	$y = x$, if $x \geq 0$, else $\alpha(e^x - 1)$	2015

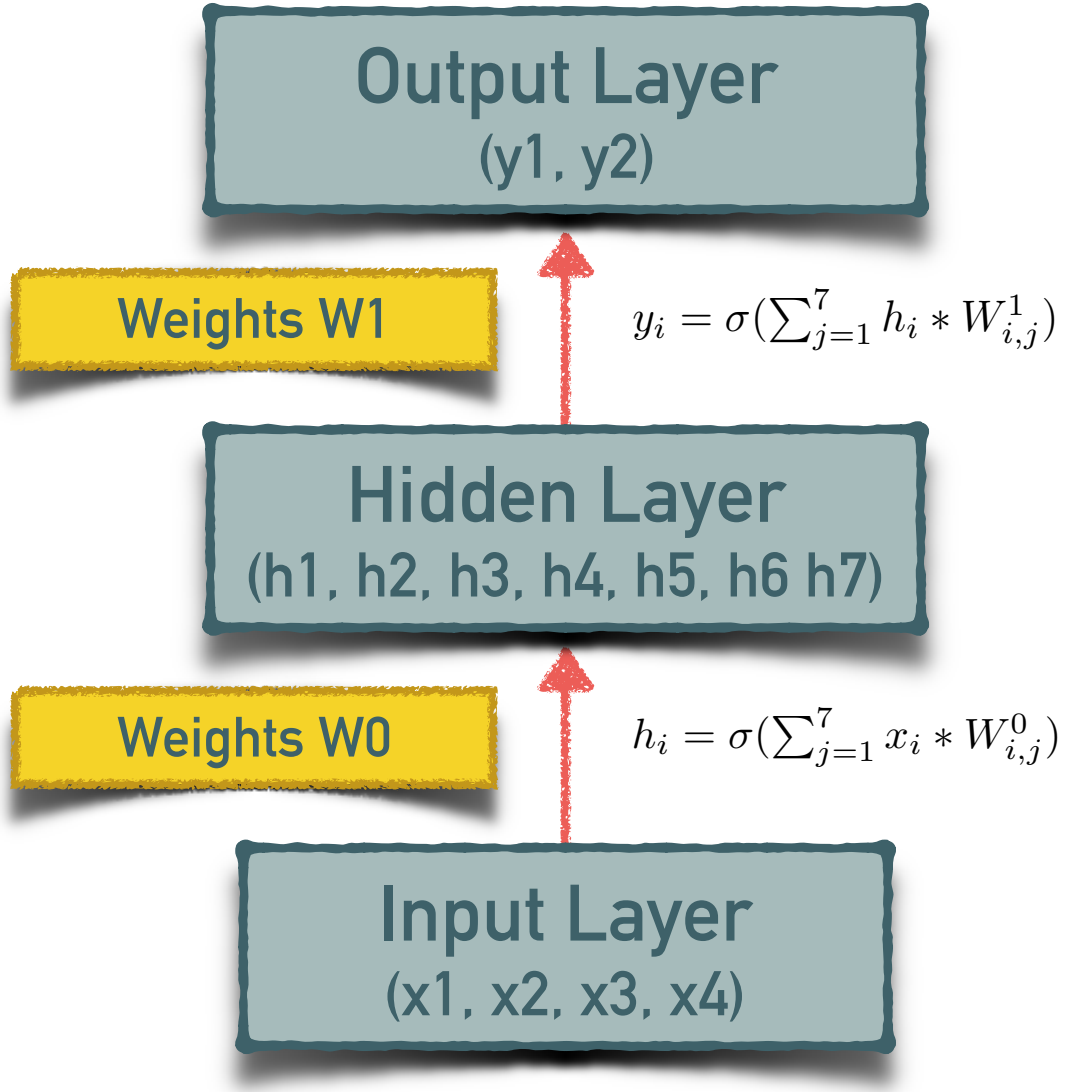
1-layer neural net model

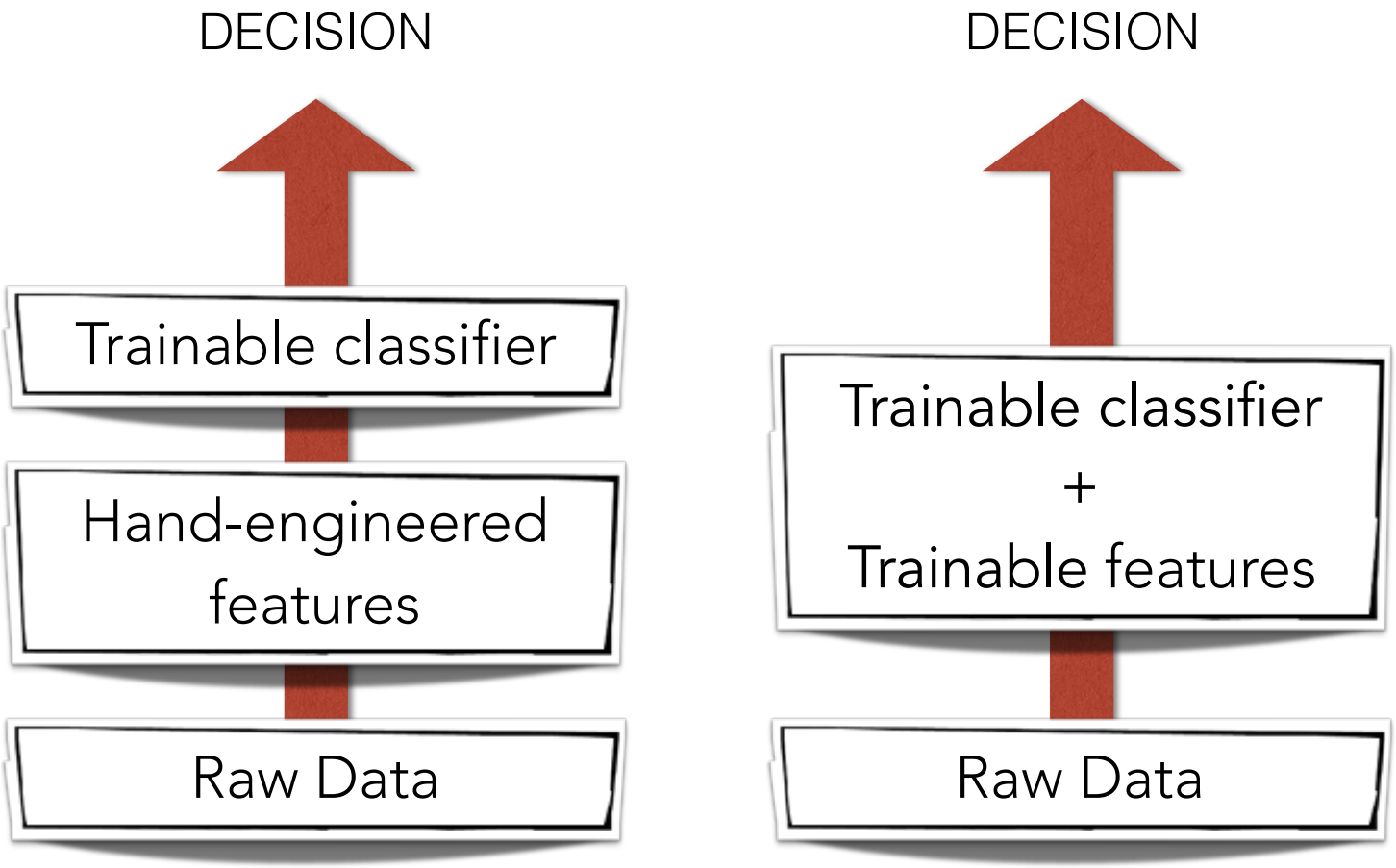


Graphical Representation

2-layer neural net model

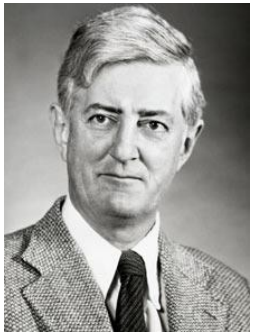


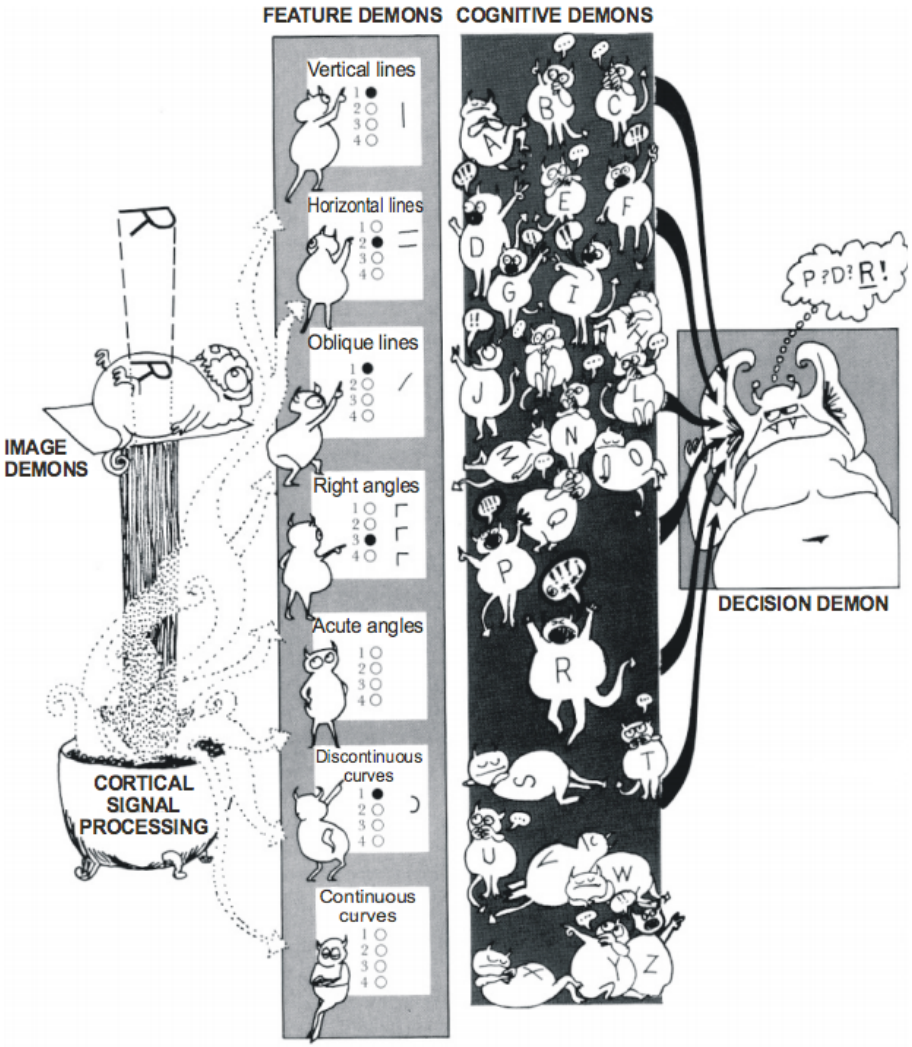




STANDARD MACHINE LEARNING

DEEP LEARNING





Pandemonium

An early architecture of vision, proposed by Oliver Selfridge in 1959.

Hype is not new

1958 New York Times...

NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather Bureau's $2,000,000 "704" computer—learned to differentiate between right and left after fifty attempts in the Navy's demonstration for newsmen.

The service said it would use this principle to build the first of its Perceptron thinking machines that will be able to read and write. It is expected to be finished in about a year at a cost of $100,000.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human beings, Perceptron will make mistakes at first, but will grow wiser as it gains experience, he said.

Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

Without Human Controls

The Navy said the perceptron would be the first non-living mechanism "capable of receiving, recognizing and identifying its surroundings without any human training or control."

The "brain" is designed to remember images and information it has perceived itself. Ordinary computers remember only what is fed into them on punch cards or magnetic tape.

Later Perceptrons will be able to recognize people and call out their names and instantly translate speech in one language to speech or writing in another language, it was predicted.

Mr. Rosenblatt said in principle it would be possible to build brains that could reproduce themselves on an assembly line and which would be conscious of their existence.

In today's demonstration, the "704" was fed two cards, one with squares marked on the left side and the other with squares on the right side.

Learns by Doing

In the first fifty trials, the machine made no distinction between them. It then started registering a "Q" for the left squares and "O" for the right squares.

Dr. Rosenblatt said he could explain why the machine learned only in highly technical terms. But he said the computer had undergone a "self-induced change in the wiring diagram."

The first Perceptron will have about 1,000 electronic "association cells" receiving electrical impulses from an eye-like scanning device with 400 photo-cells. The human brain has 10,000,000,000 responsive cells, including 100,000,000 connections with the eyes.