An Introduction to Deep Learning

Patrick Emami

University of Florida

Department of Computer and Information Science and Engineering

September 7, 2017

Overview

1. What is Deep Learning?
   - The General Framework
   - A Brief History of Deep Neural Networks
2. Why is Deep Learning so successful?
   - Big Data Era
3. Applications and Architectures
   - Computer Vision
   - Natural Language Processing
   - Training Deep Neural Networks

What is Deep Learning?

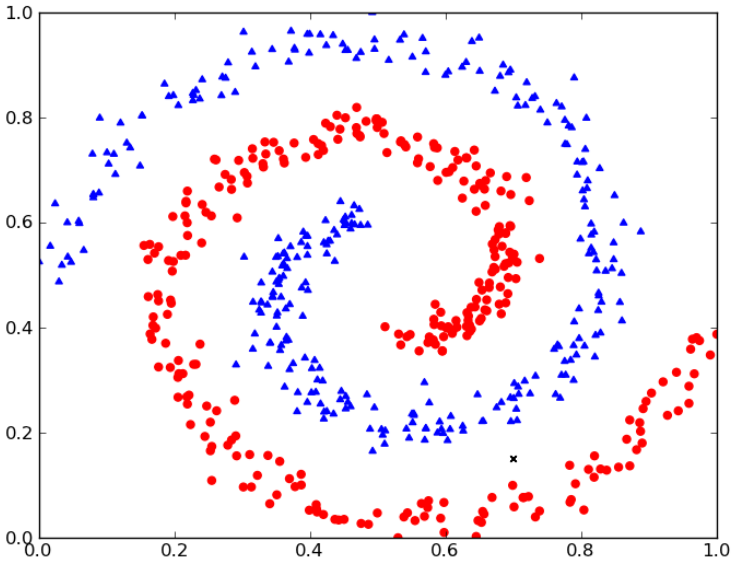
Simple Definition

Deep Learning can be viewed as the composition of many functions for the purpose of mapping input values to output values in such a way so as to encourage the discovery of representations of data.

Function Approximation

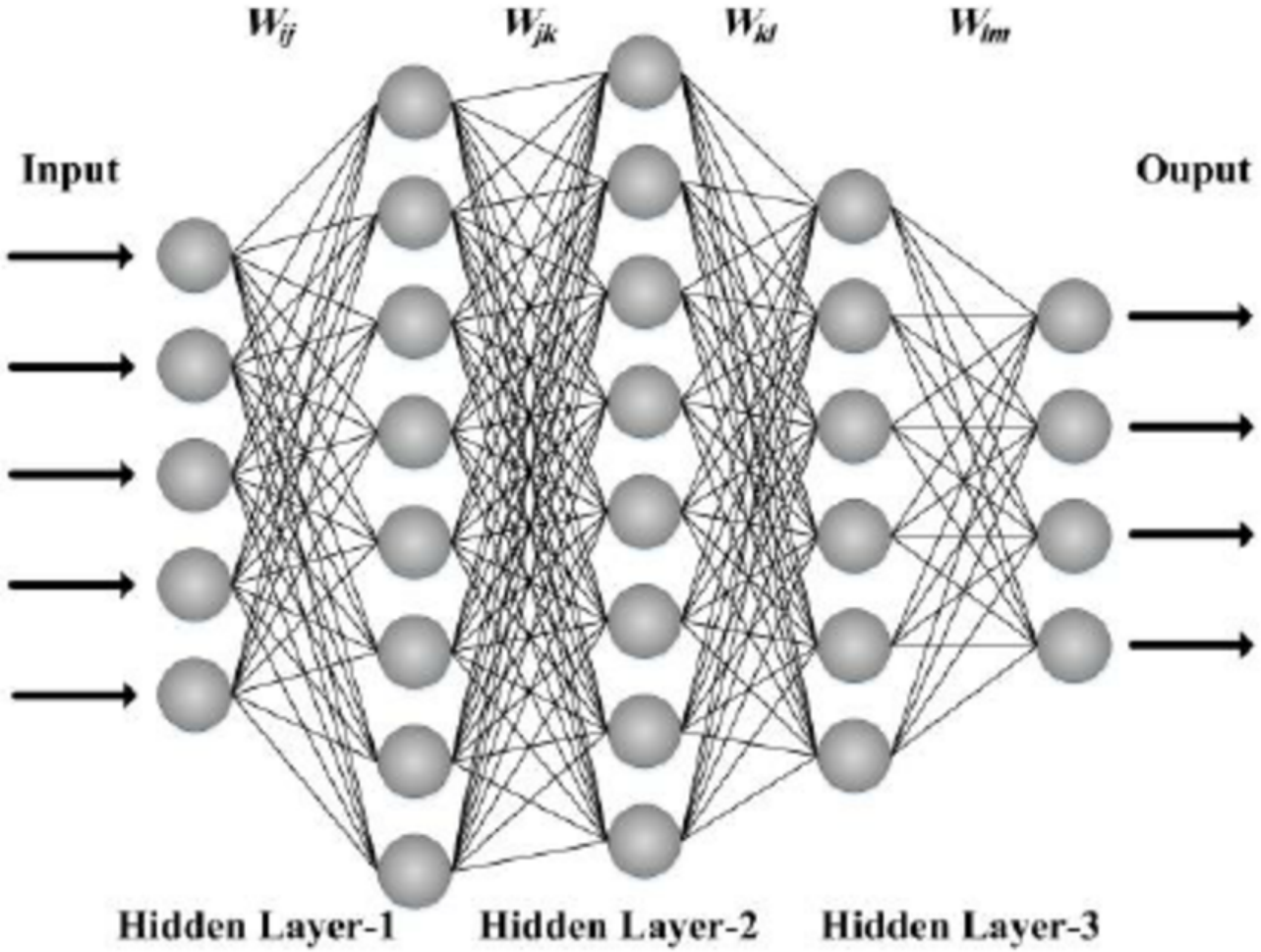
Many machine learning problems can be framed as function approximation.

Example: Given a sample of data points $x_i \in \mathbb{R}^n$, $i = 1, ..., N$ and binary labels $y_i \in \{0, 1\}$ from a dataset, find parameters θ such that $L(y, f(x, \theta))$ is minimized over all other data points x and true labels y in the dataset, for some loss function L and some family of parameterized functions f.



Source:
http://people.cs.uchicago.edu/ amr/122-w12/assignments/hw1a/index.html

Multi-Layer Perceptron (MLP)



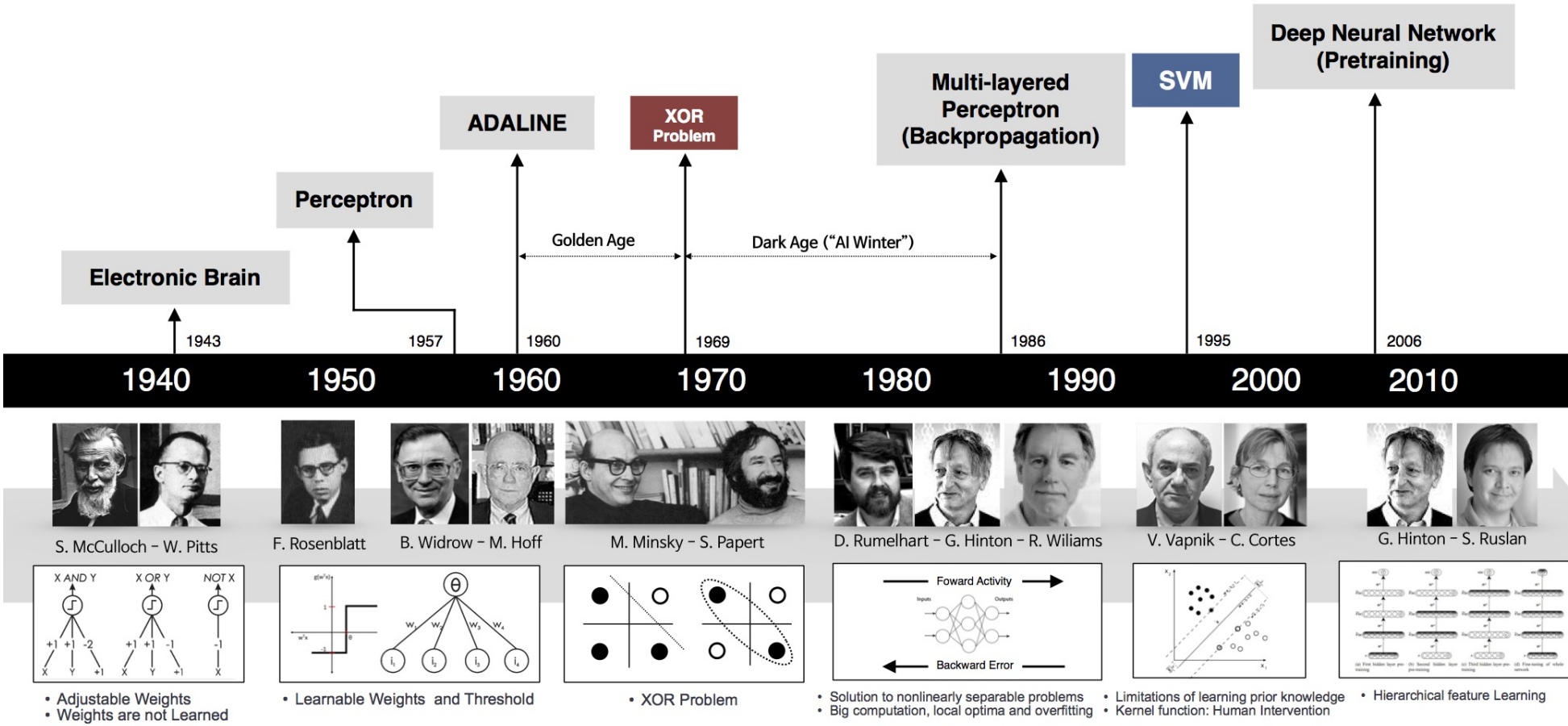
In Deep Learning, we try to approximate functions with Deep Neural Networks

Source:
https://www.researchgate.net/publication/287209604_Prediction_of_Final_Concentrate_Grade_Using_Artificial_Neural_Networks_from_Gol-E-Gohar_Iron_Ore_Plant

Universal Function Approximation

It was shown in [Hornik, 1991] that a multi-layer perceptron is a **universal function approximator**. This means that, given enough hidden units, it can model any suitably smooth function to any desired level of accuracy

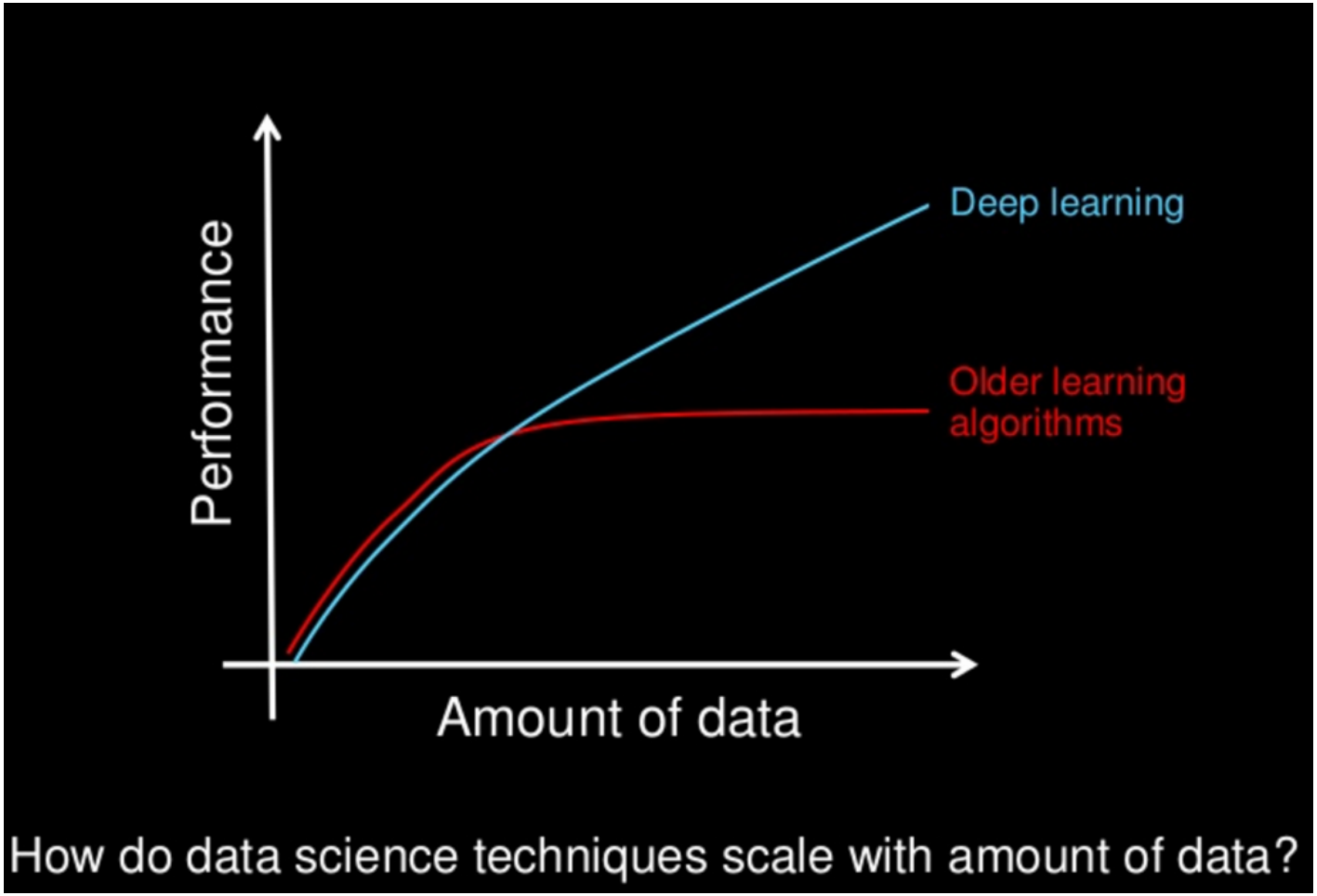
History of Neural Networks



Source:
https://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html

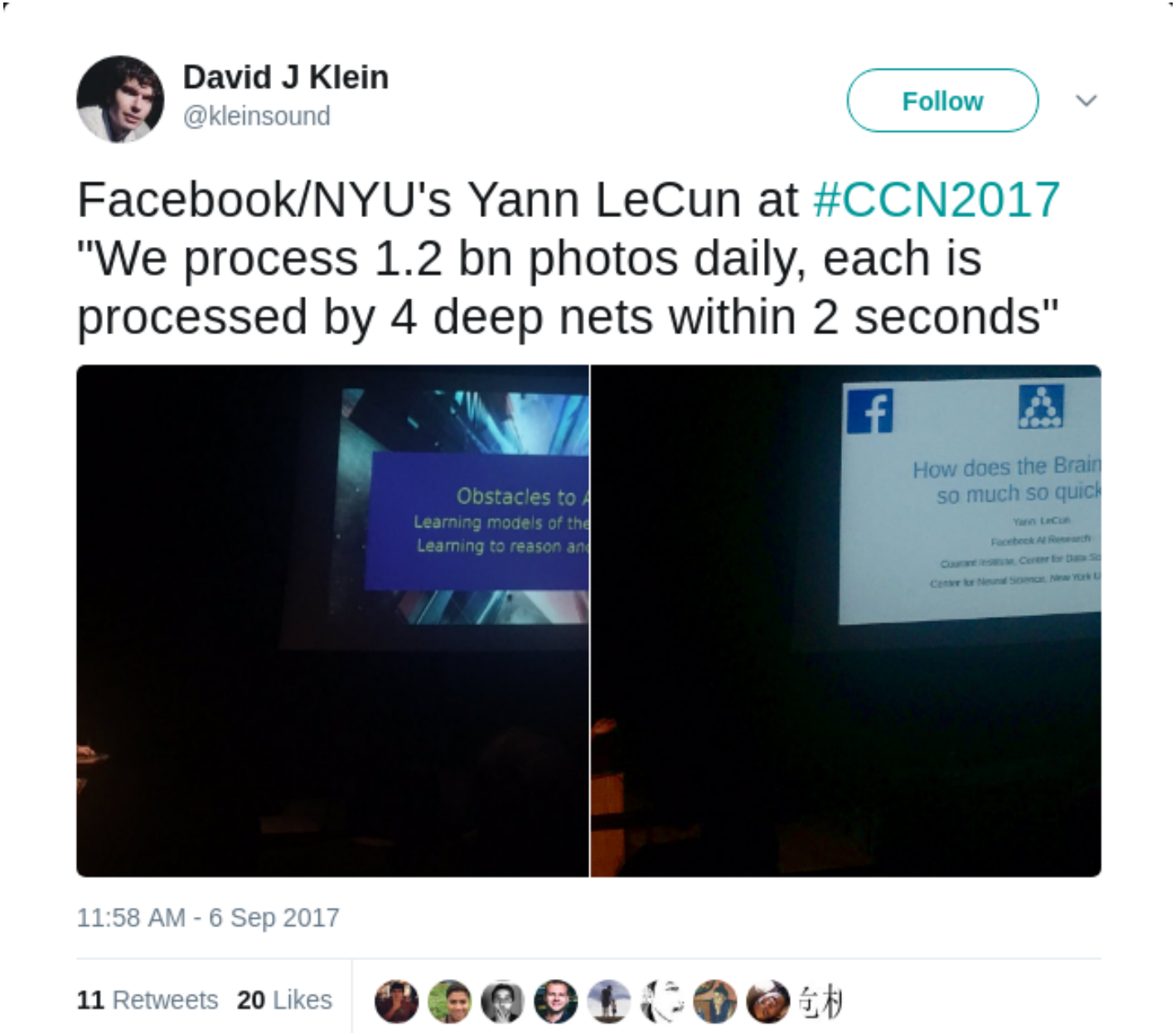
Why is Deep Learning so successful?

Scalability

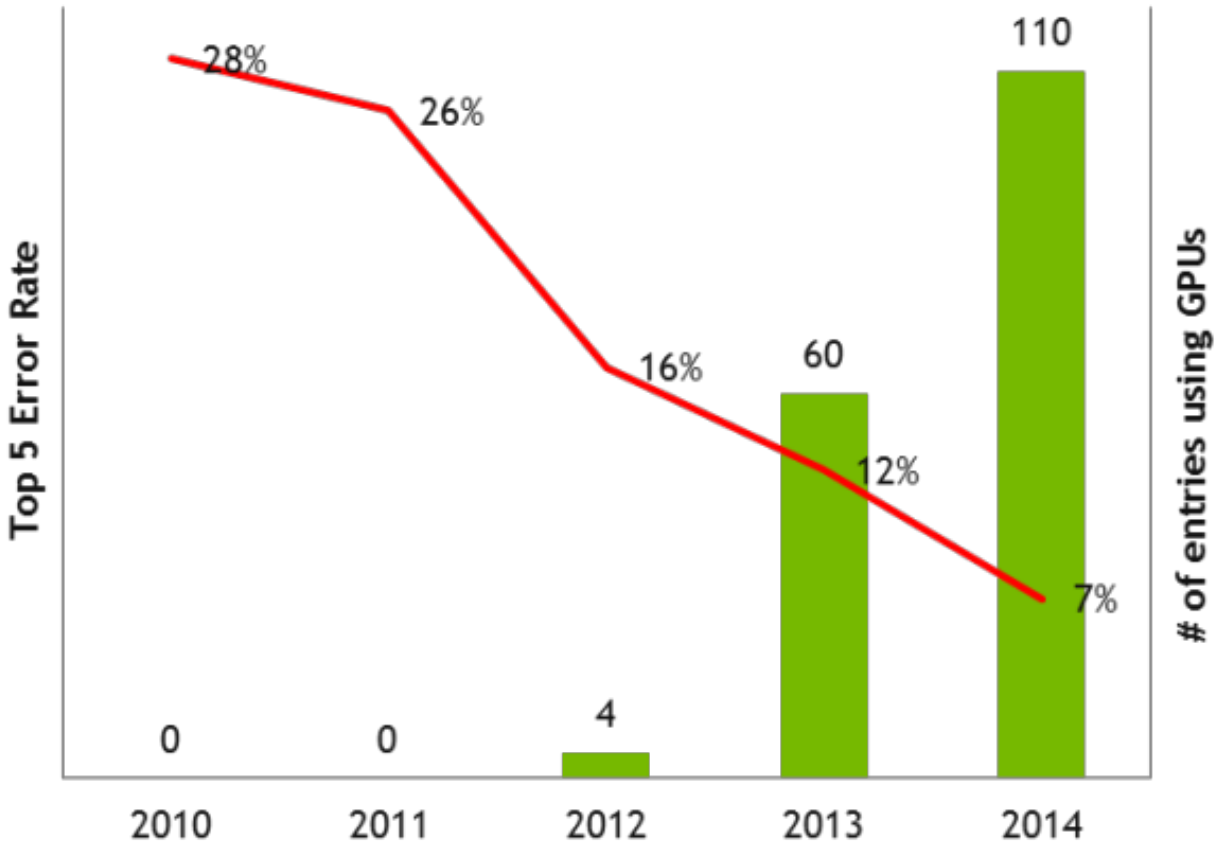


Source: https://machinelearningmastery.com/what-is-deep-learning/

Scalability



IMAGENET



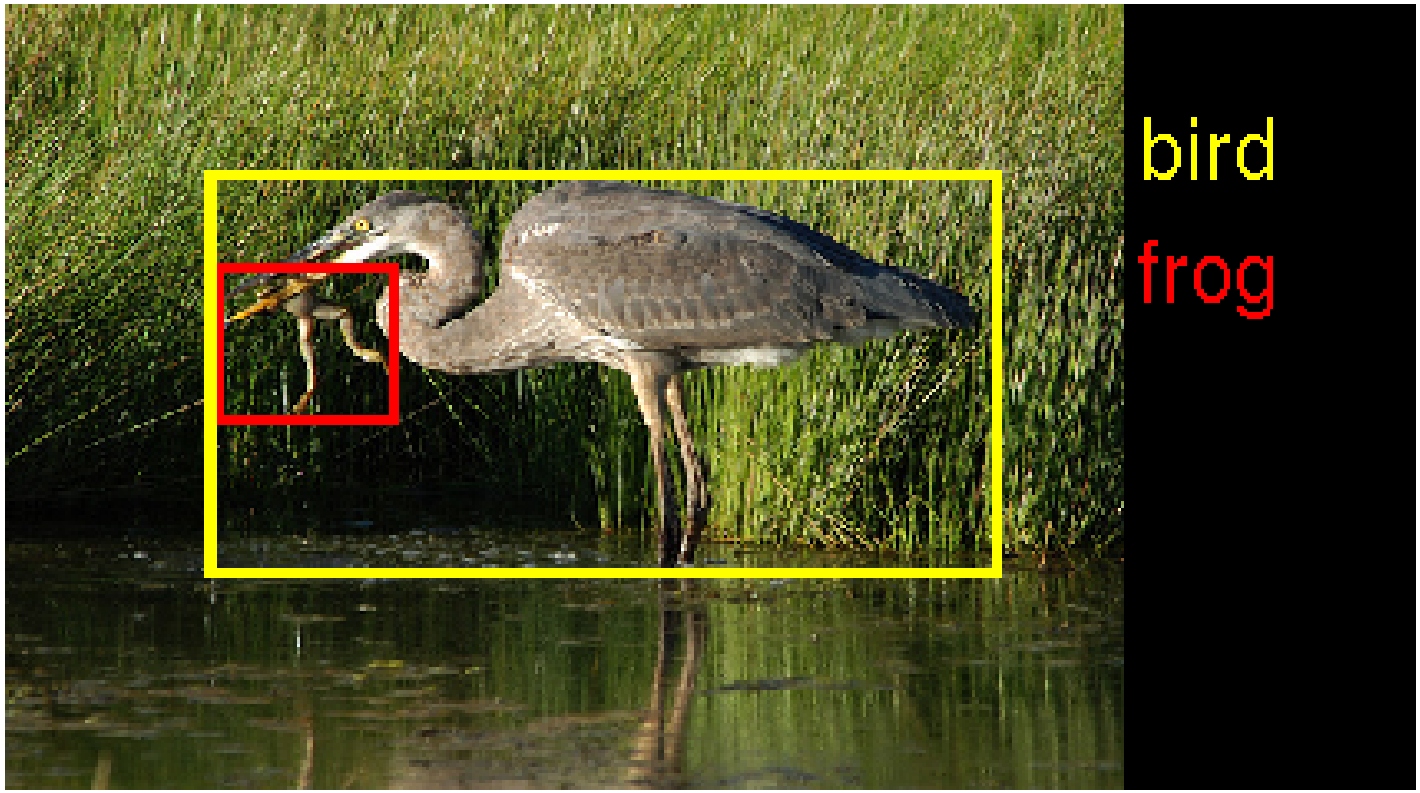
NVIDIA's graphics cards and CUDA library allows for extremely fast matrix operations on DNNs with millions of parameters

Source:
https://devblogs.nvidia.com/parallelforall/nvidia-ibm-cloud-support-imagenet-large-scale-visual-recognition-challenge/

Deep Learning Frameworks

Applications and Architectures

Computer Vision



Object Detection

Source:
https://www.kaggle.com/c/imagenet-object-detection-challenge

Computer Vision

Semantic Segmentation

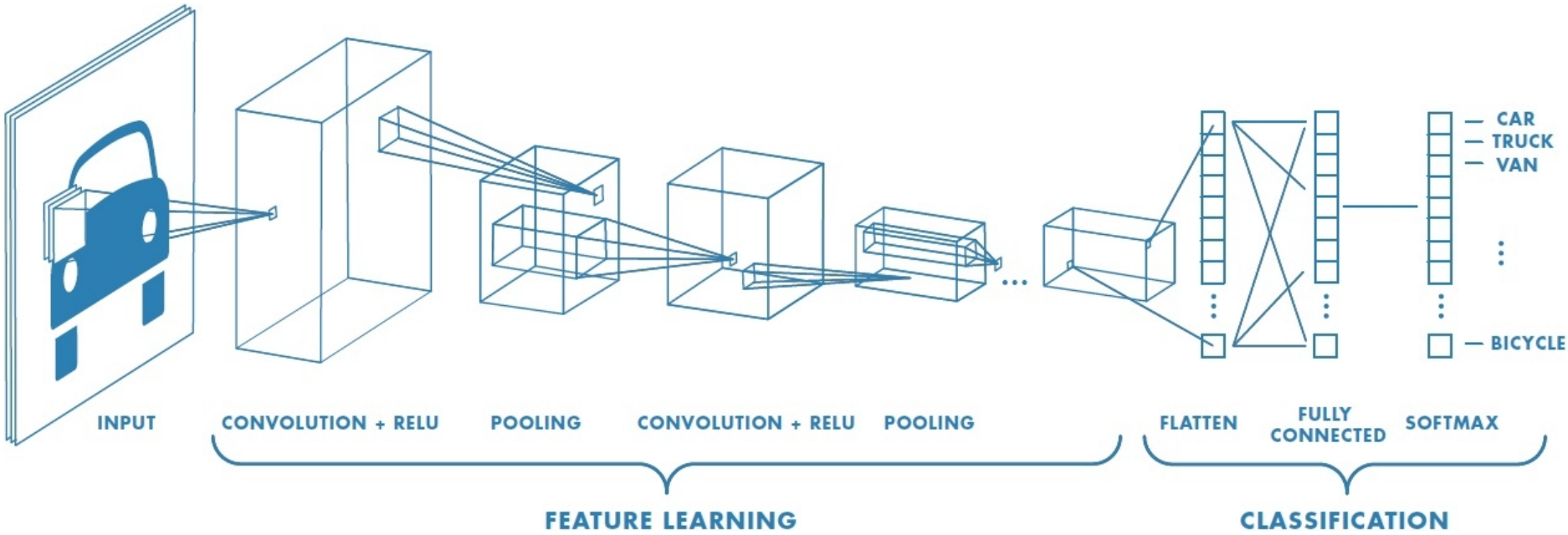
Source:
http://nicolovaligi.com/deep-learning-models-semantic-segmentation.html

Computer Vision

Multi-Object Tracking

Source:
https://www.youtube.com/watch?v=C4ZtzG4CkZs

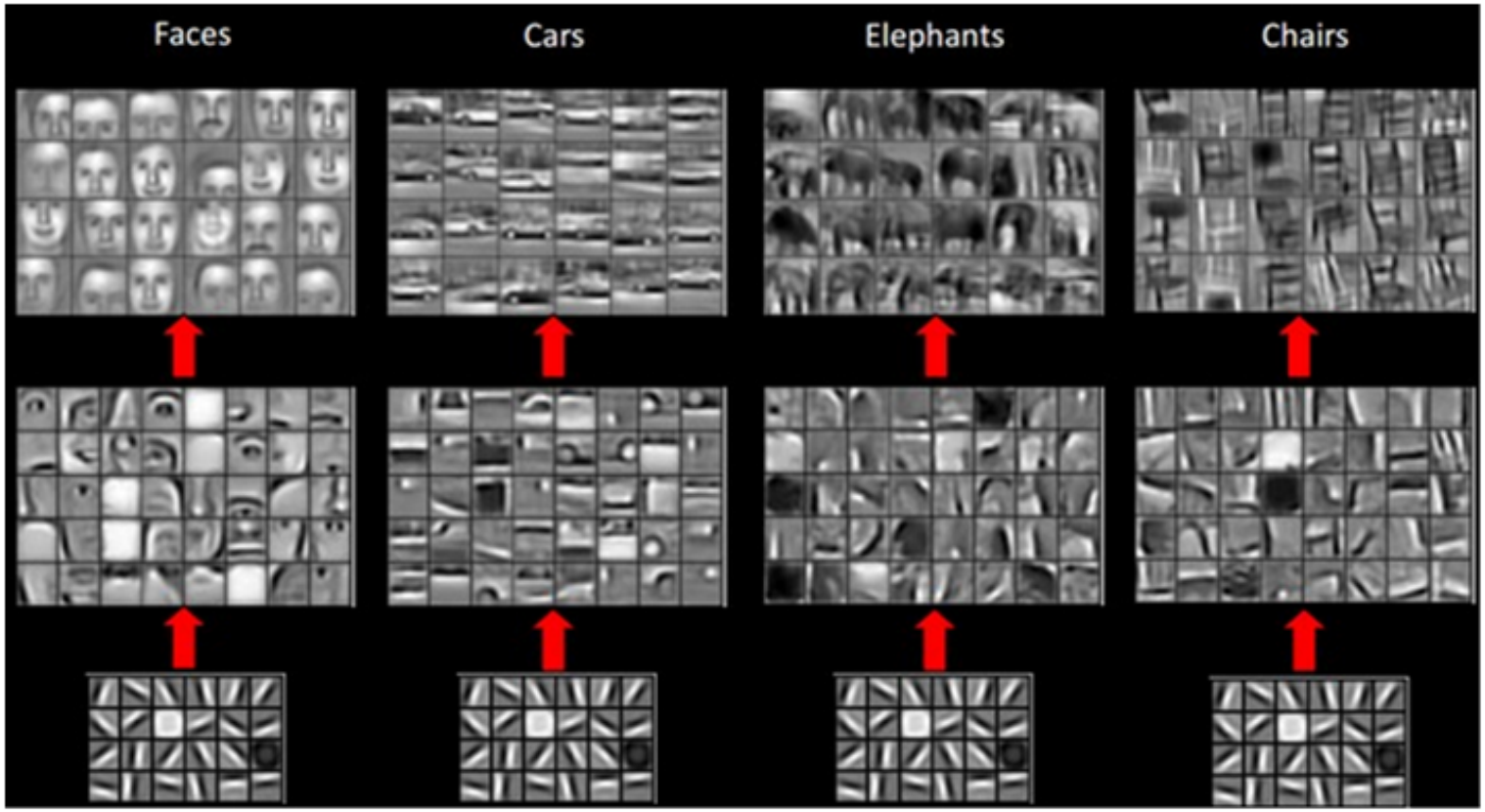
Convolutional Neural Networks



A CNN [Krizhevsky, 2012] for multi-class classification. CNNs can also be used for many other learning tasks such as regression by changing the output layer.

Source: https://www.mathworks.com/discovery/convolutional-neural-network.html

Learned Representations



Source:
https://stats.stackexchange.com/questions/146413/why-convolutional-neural-networks-belong-to-deep-learning

Binary Classification with CNNs

The negative log-likelihood for 0-1 binary classification with CNNs:

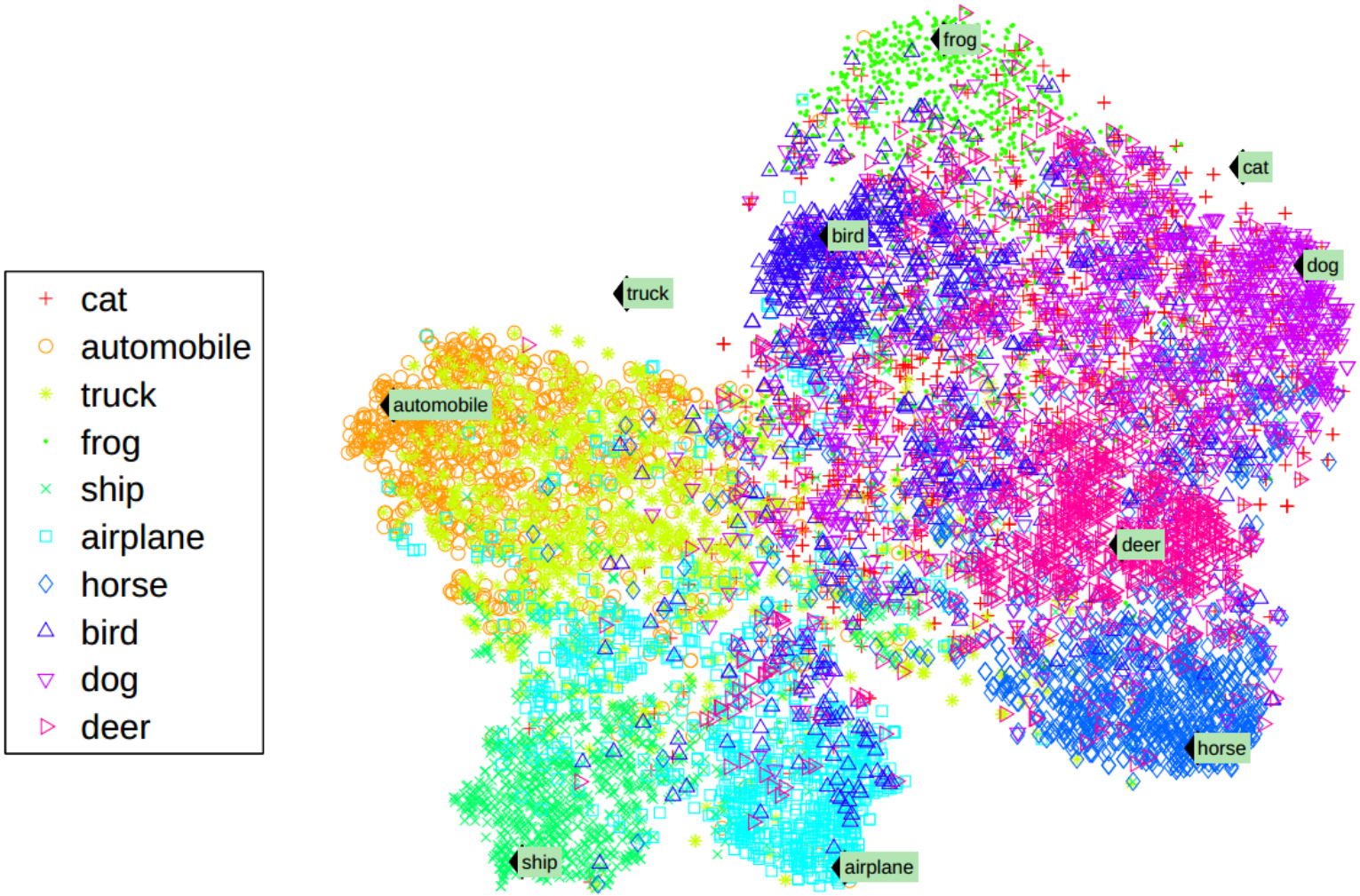
$$p(y|x,\theta) = \text{Bernoulli}(y|\sigma(\mathbf{w}^\intercal g(x,\theta) + \mathbf{b}))$$

Setting $\sigma(\mathbf{w}^\intercal g(x,\theta) + \mathbf{b})$ to p, $p \in \{0,1\}$,

$$= p^y(1-p)^{1-y}$$
$$\text{NLL}(x,\theta) = -(y \log p + (1-y)\log(1-p)).$$

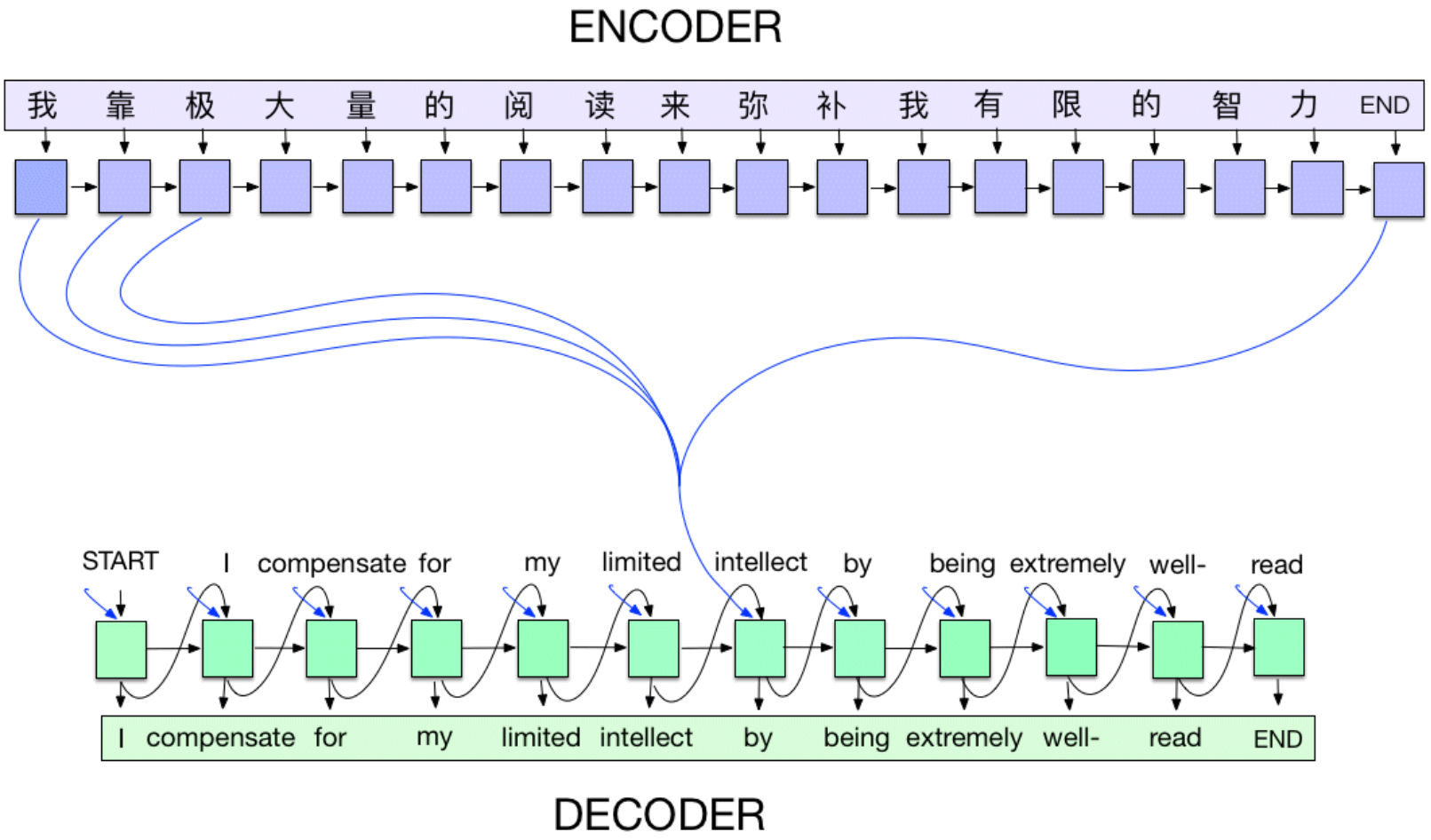
So for $p < 0.5$, your CNN should predict $y = 1$, and for $p >= 0.5$, it should predict $y = 0$. Nonlinear and non-convex optimization problem!

Natural Language Processing



Distributed Word Representations

Source: http://colah.github.io/posts/2014-07-NLP-RNNs-Representations/



Machine Translation

Source: https://opensource.googleblog.com/2017/04/tf-seq2seq-sequence-to-sequence-framework-in-tensorflow.html

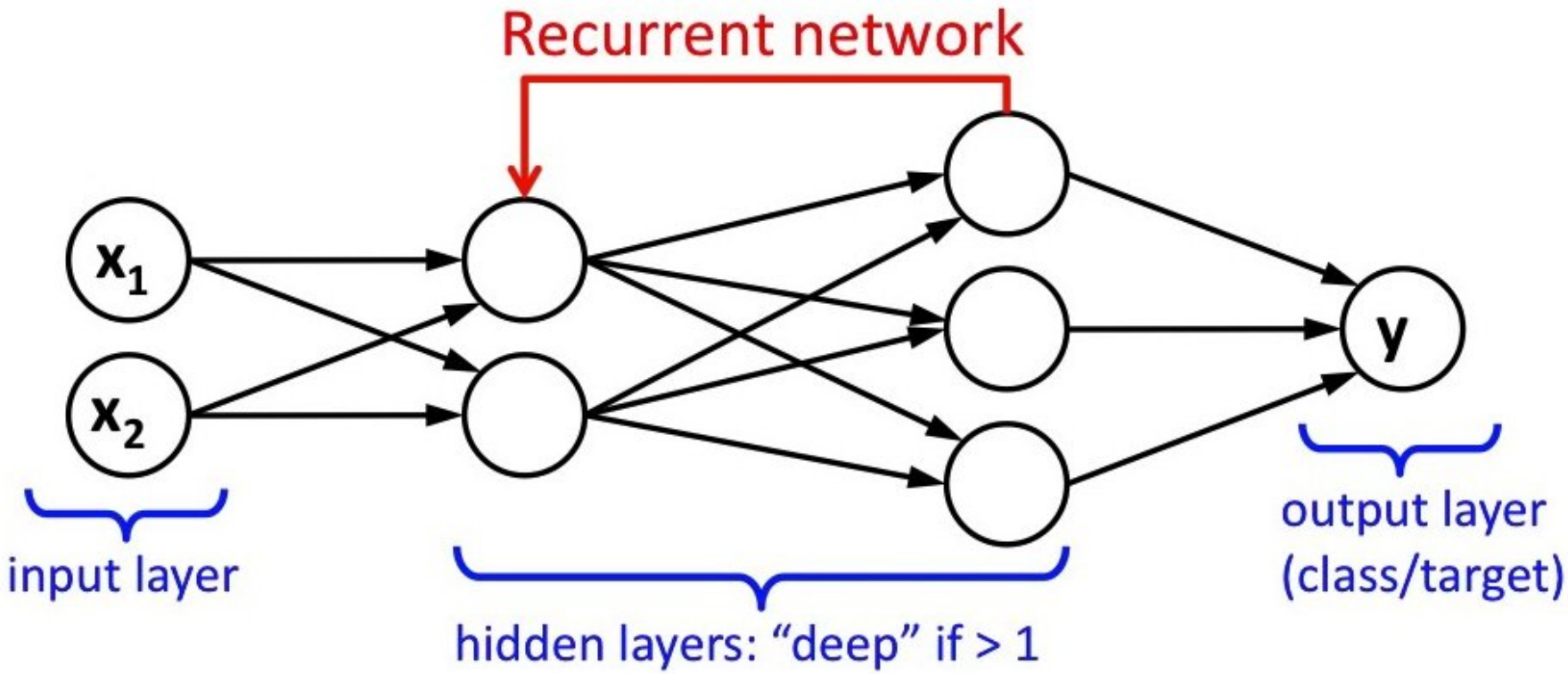
Natural Language Processing

Input: Article 1st sentence	Model-written headline
metro-goldwyn-mayer reported a third-quarter net loss of dlrs 16 million due mainly to the effect of accounting rules adopted this year	mgm reports 16 million net loss on higher revenue
starting from july 1, the island province of hainan in southern china will implement strict market access control on all incoming livestock and animal products to prevent the possible spread of epidemic diseases	hainan to curb spread of diseases
australian wine exports hit a record 52.1 million liters worth 260 million dollars (143 million us) in september, the government statistics office reported on monday	australian wine exports hit record high in september

Text Summarization

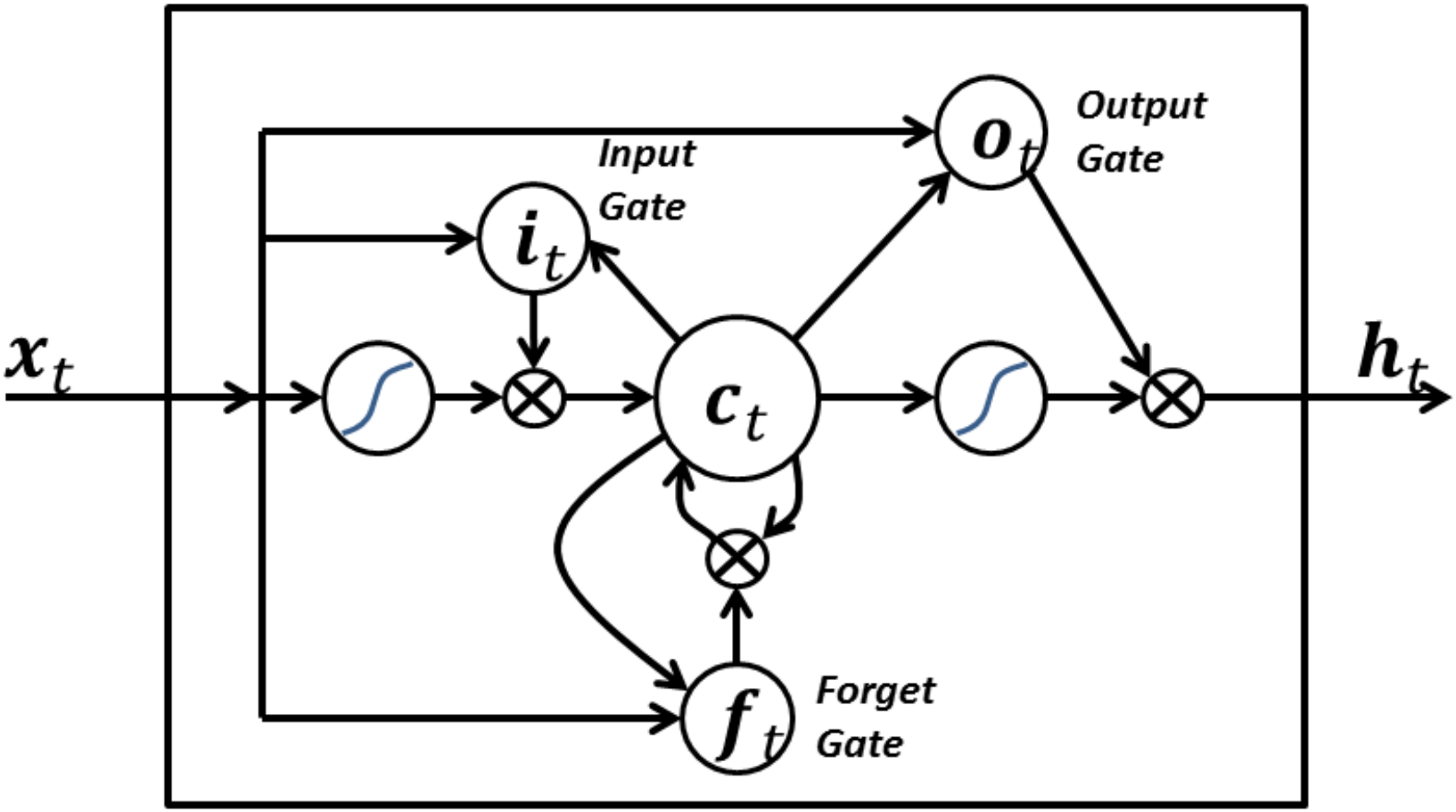
Source:
http://www.kdnuggets.com/2016/09/deep-learning-august-update-part-2.html

Recurrent Neural Networks



Source:
https://leonardoaraujosantos.gitbooks.io/artificial-inteligence/content/recurrent_neural_networks.html

Long Short-Term Memory



The LSTM cell, well suited for large bodies of text [Hochreiter, 1997]

Source:
https://commons.wikimedia.org/wiki/File:Long_Short_Term_Memory.png

Backpropagation

Goal: Find the optimal set of parameters for the Deep Neural Network that minimizes the loss on the training set without overfitting.

Solution: With your training set, compute the gradient of the loss with respect to the parameters in each layer and set this equal to 0. Use the chain rule!

Gradients flow "backwards" from the output to the input layer. Auto-differentiation engines, like Tensorflow, handle this for us nowadays.

Stochastic Gradient Descent

Use mini-batch stochastic gradient descent to update parameters, since using the full dataset can be too expensive. The following is an example of updating a single weight w using our negative log-likelihood loss from earlier.

$$\triangle = \frac{1}{B} \sum_{i=1}^{B} \nabla_w \text{NLL}(x_i, \theta)$$

$$w' = w - \alpha \triangle$$

Resources

1. http://www.fast.ai/
2. https://www.udacity.com/course/deep-learning--ud730
3. http://www.deeplearningbook.org/
4. https://keras.io/

References

Hornik, Kurt (1991)

Approximation capabilities of multilayer feedforward networks

Neural networks 4(2), 251 – 257

Krizhevsky, Alex and Sutskever, Ilya and Hinton, Geoffrey E (2012)

Imagenet classification with deep convolutional neural networks

Advances in neural information processing systems

Hochreiter, Sepp and Schmidhuber, Juergen (1997)

Long Short-Term Memory

Neural Computation 9(8), 1735 – 1780

Questions?