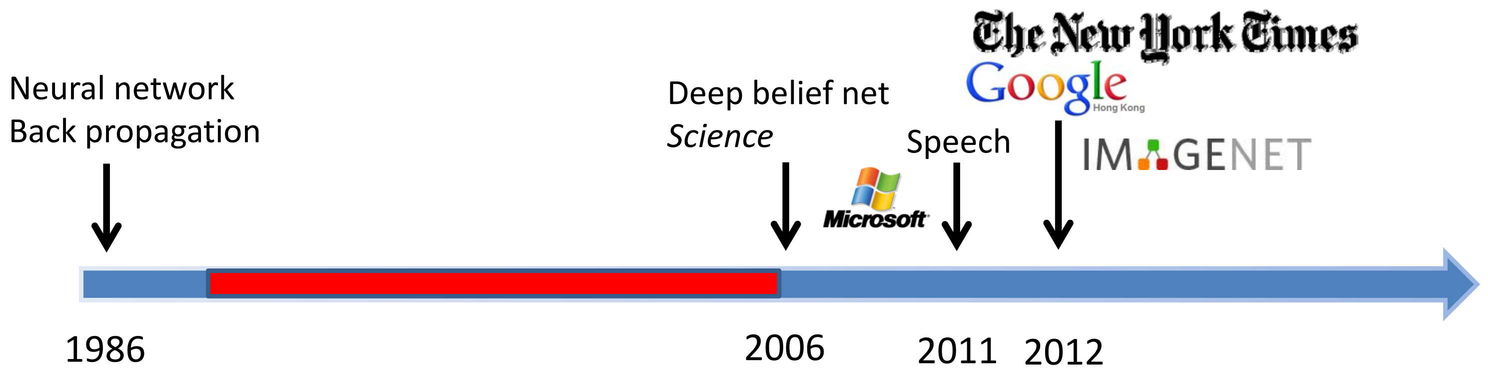
Lecture 18: Learning 7

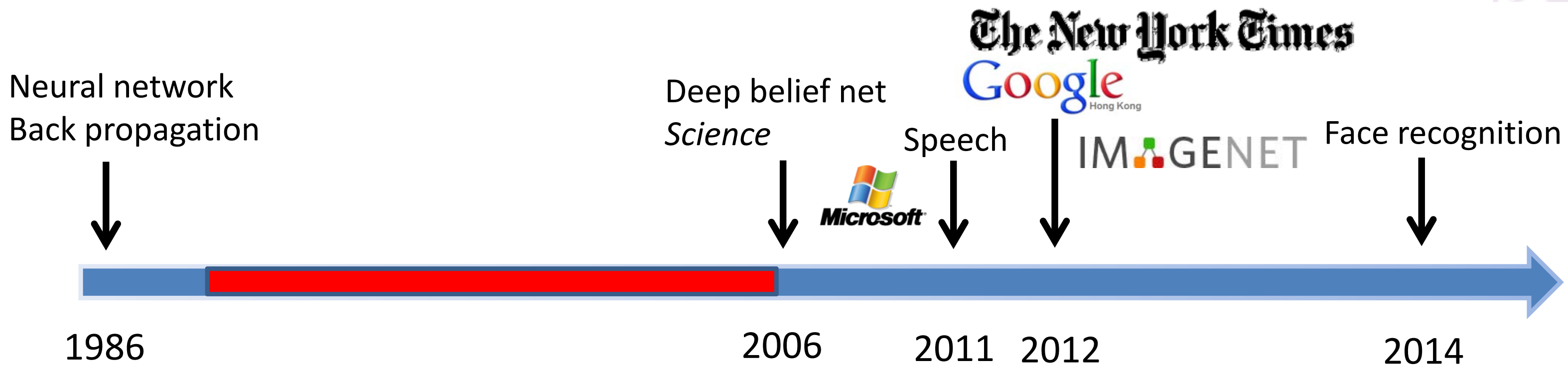
http://cs.nju.edu.cn/yuy/course_ai17.ashx

Historical review of deep learning



- Google and Baidu announced their deep learning based visual search engines (2013)
  - [Google]()
    - "on our test set we saw **double the average precision** when compared to other approaches we had tried. We acquired the rights to the technology and went full speed ahead adapting it to run at large scale on Google's computers. We took cutting edge research straight out of an academic research lab and launched it, in just a little over six months."
  - [Baidu]()

Historical review of DL (con't)



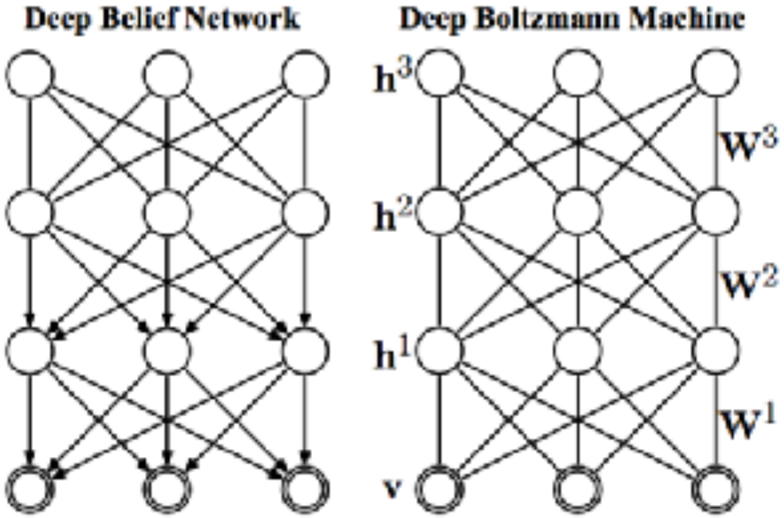
- Deep learning achieves 99.47% face verification accuracy on Labeled Faces in the Wild (LFW), higher than human performance

Y. Sun, X. Wang, and X. Tang. Deep Learning Face Representation by Joint Identification-Verification. NIPS, 2014.

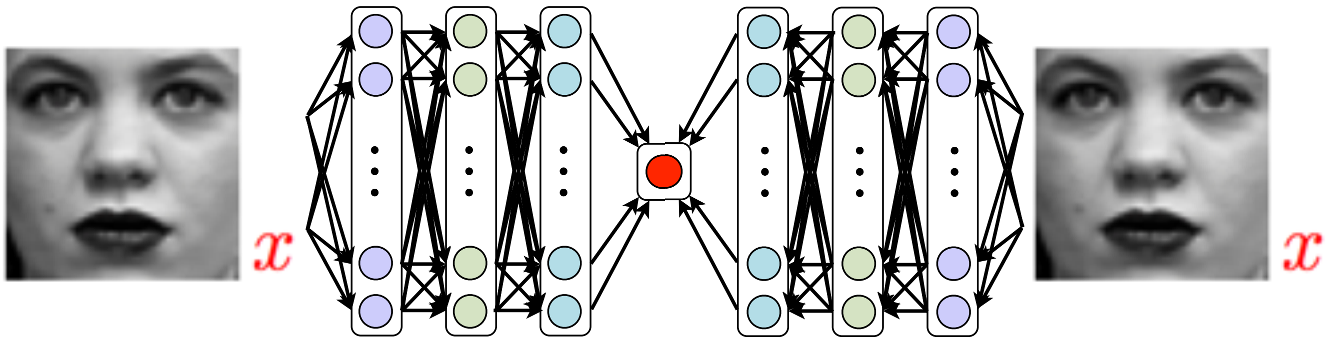
Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. CVPR, 2015.

Major types

Deep Boltzmann machine:

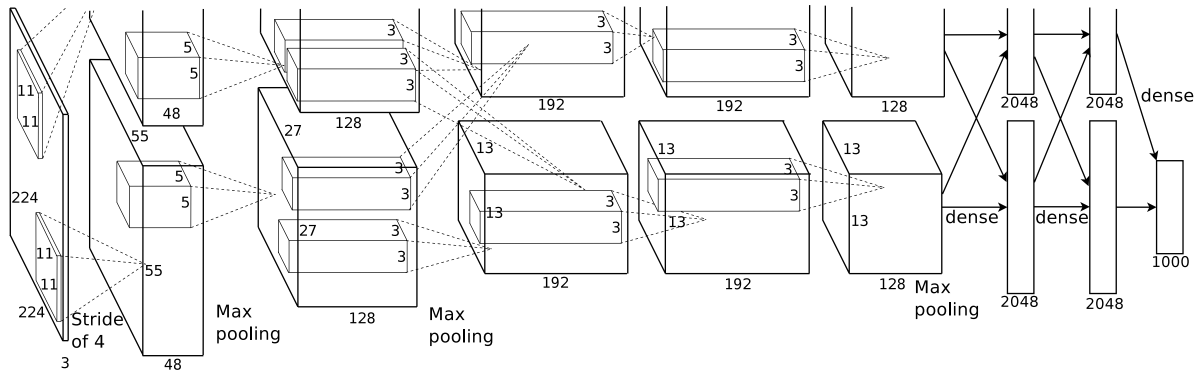


Auto-encoder:

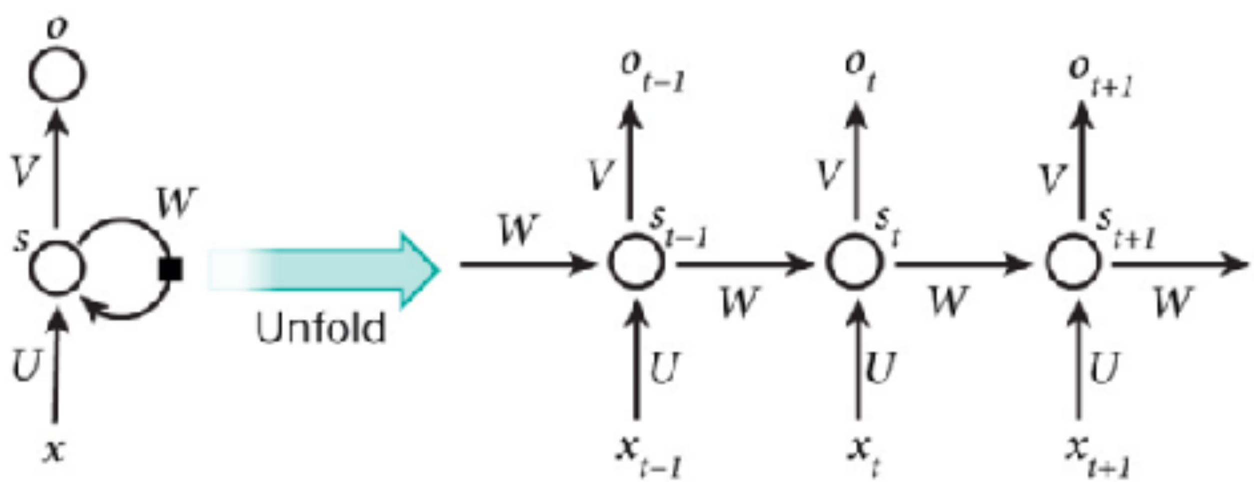


Major types

Convolutional neural networks:



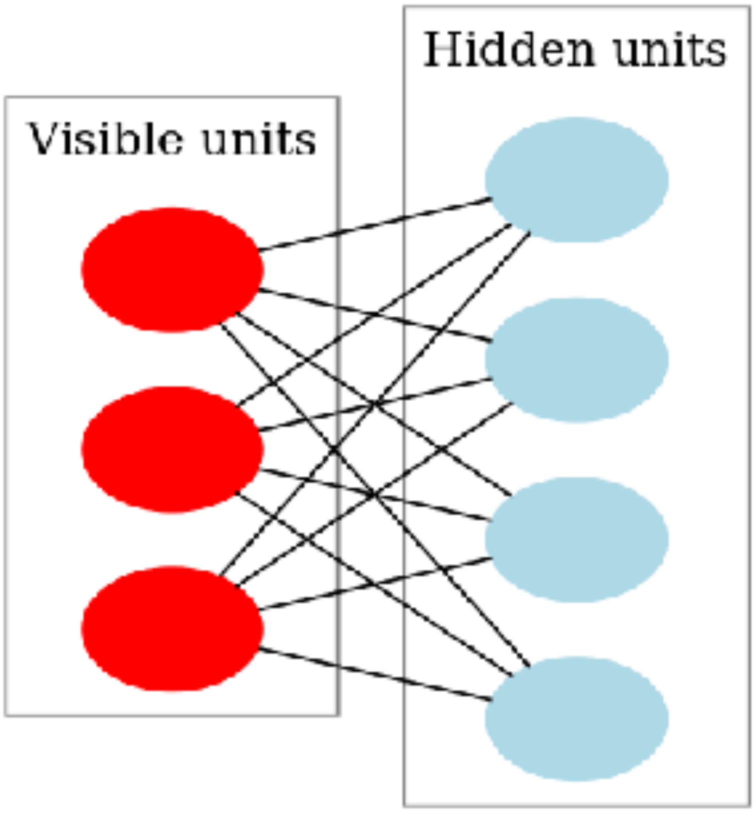
Recurrent neural networks:



Autoencoder

autoencoder

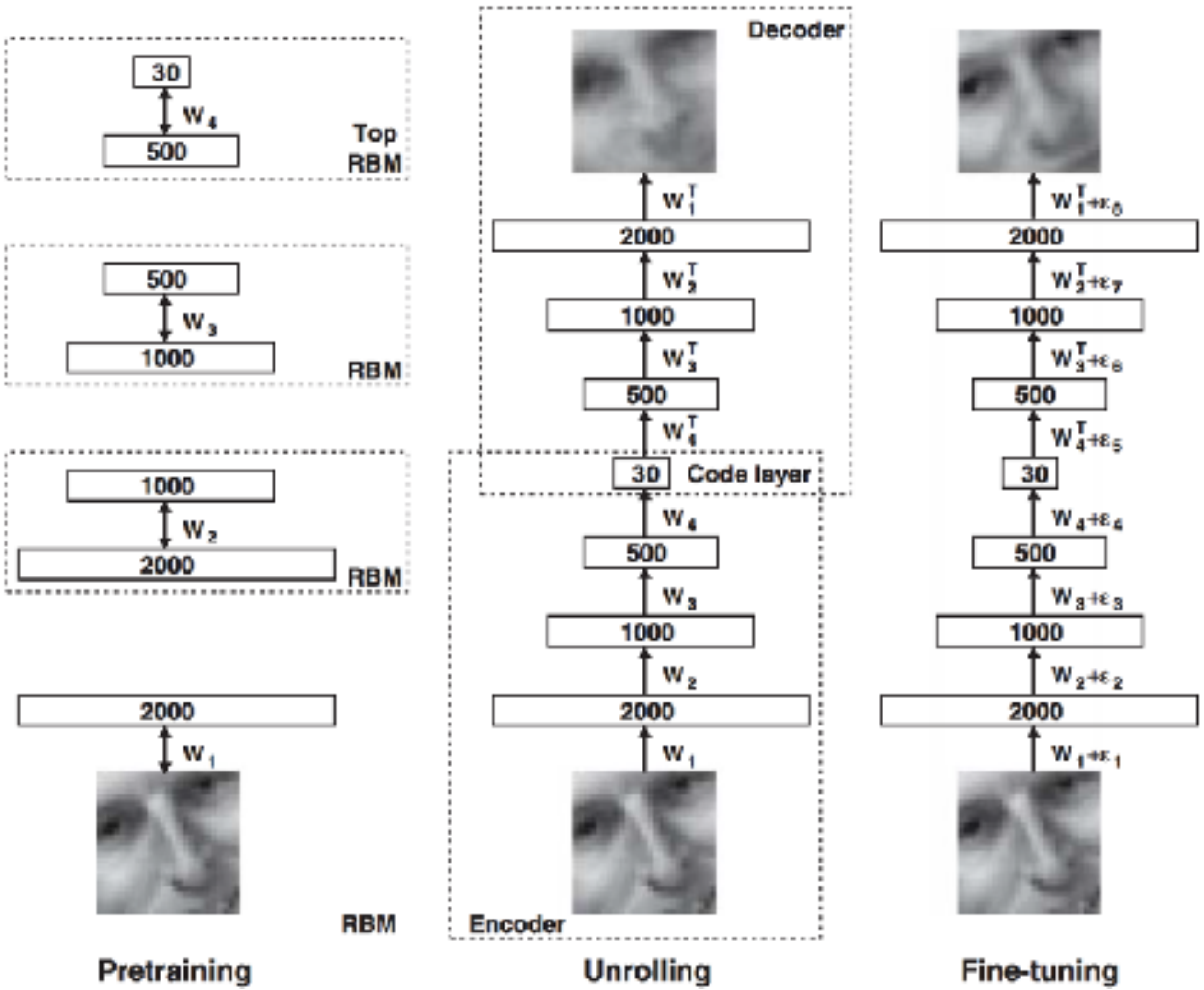
restricted Boltzmann machine
a type of associative memory network



[image from http://en.wikipedia.org/wiki/Restricted_Boltzmann_machine

Autoencoder

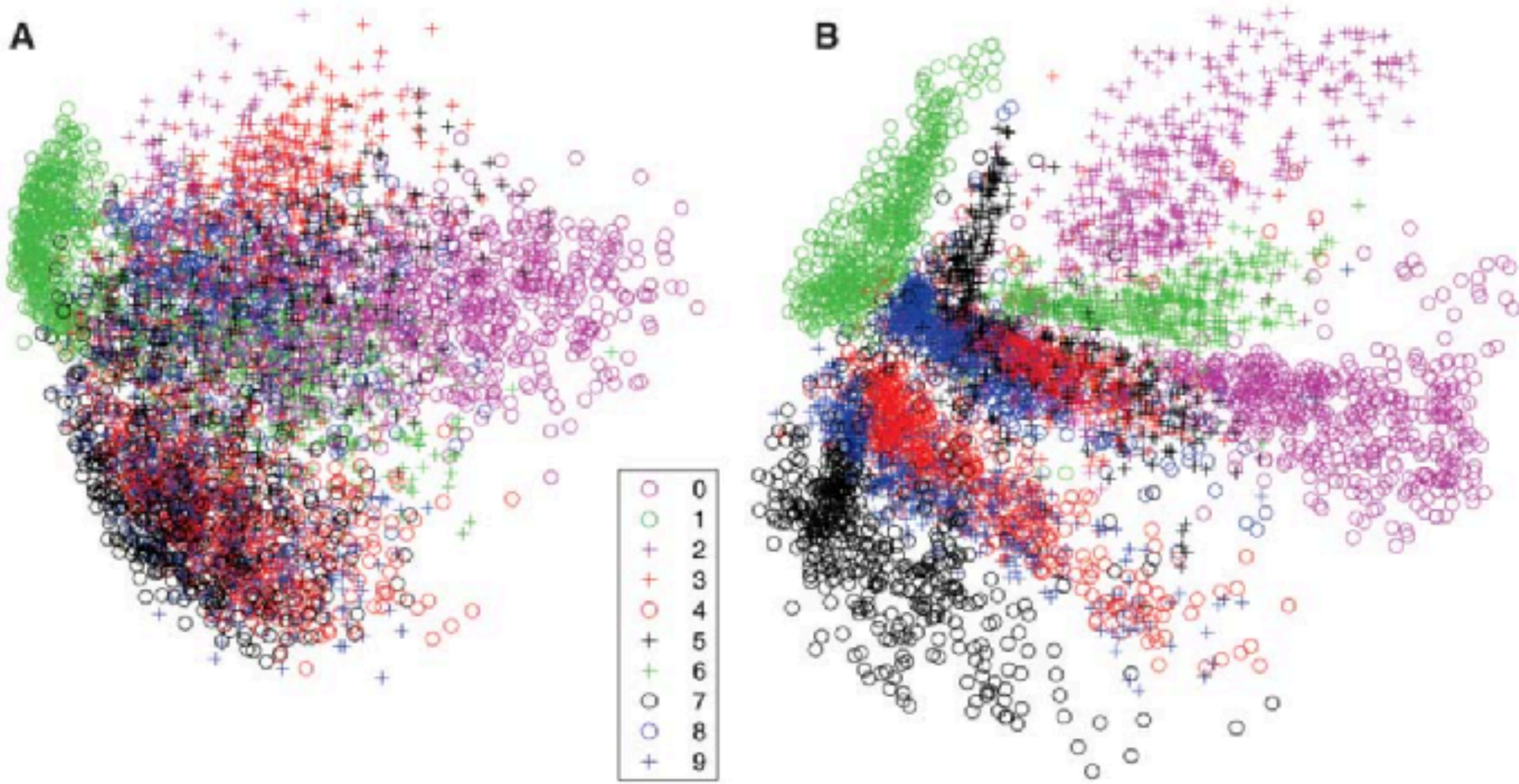
autoencoder



[image from [G. E. Hinton and R. R. Salakhutdinov, Science 2006]]

Autoencoder

autoencoder

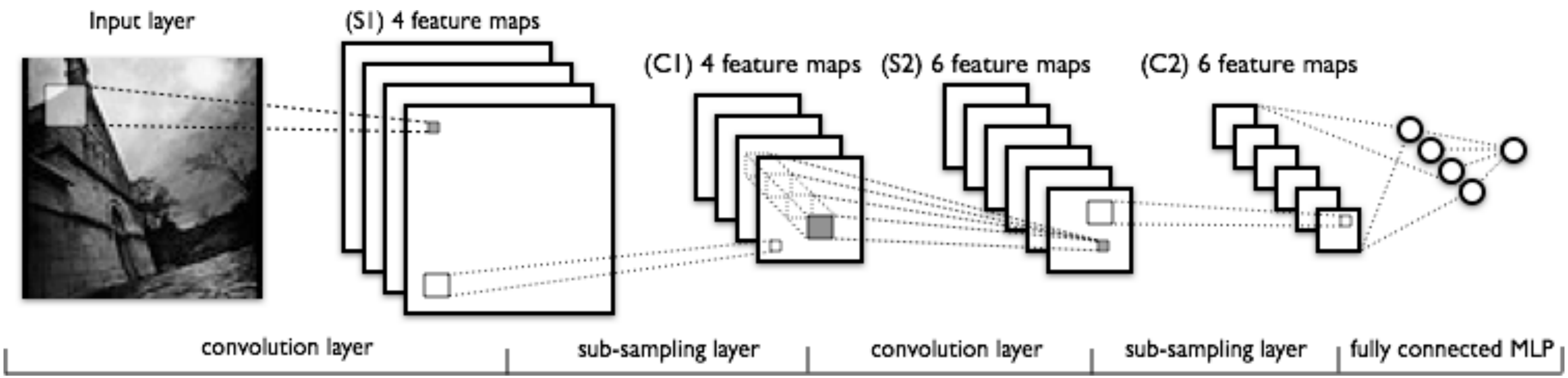


PCA autoencoder

[image from [G. E. Hinton and R. R. Salakhutdinov, Science 2006]]

CNN

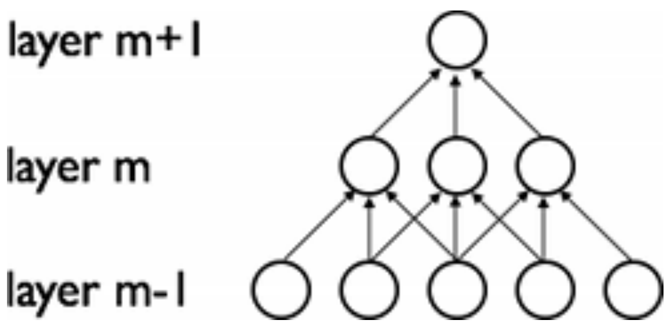
Convolutional Neural Networks (CNN/LeNet) for general image feature extraction

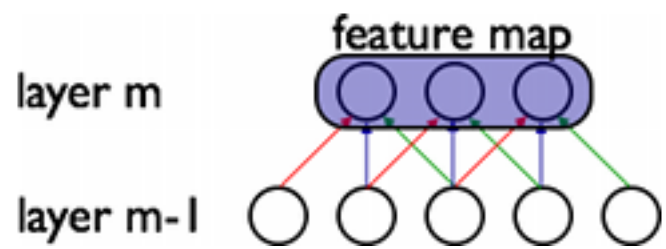


[image from http://deeplearning.net/tutorial/lenet.html]

CNN

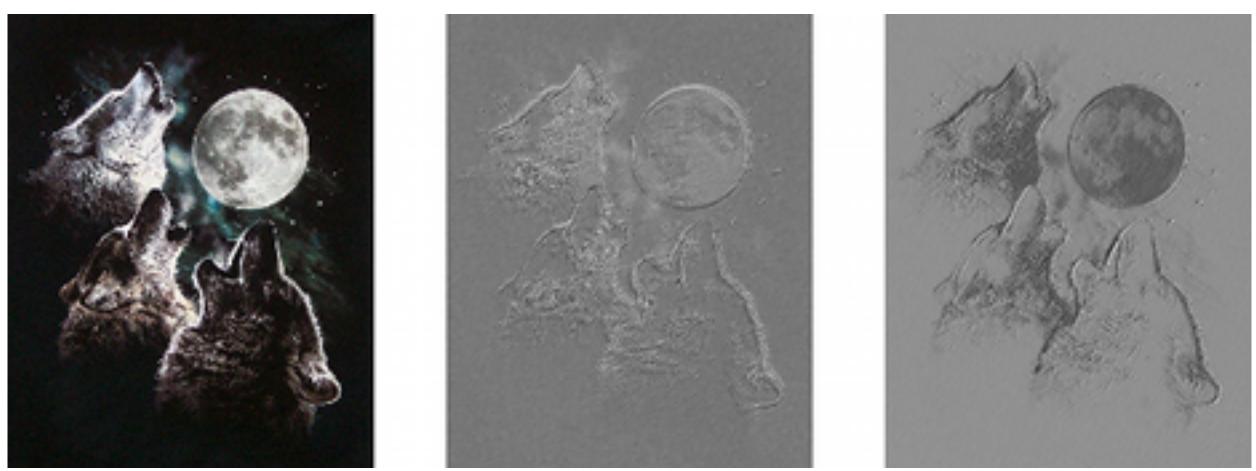
Convolution layer





sparse connectivity

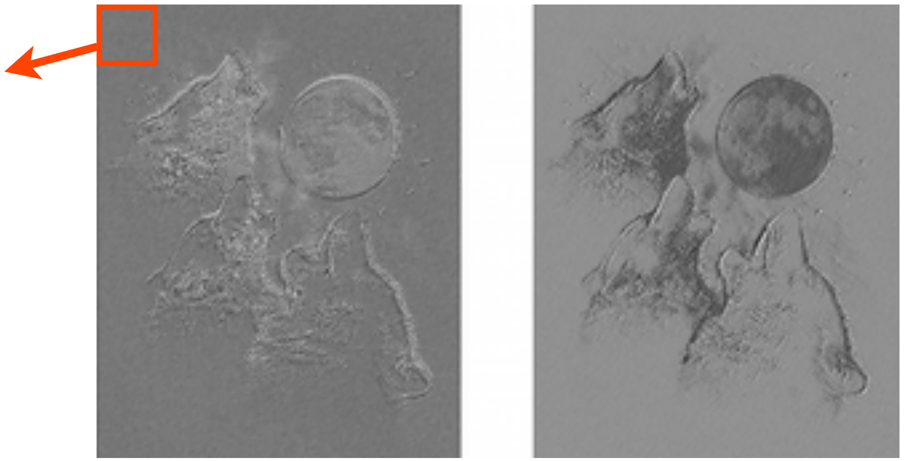
shared weights



[image from http://deeplearning.net/tutorial/lenet.html]

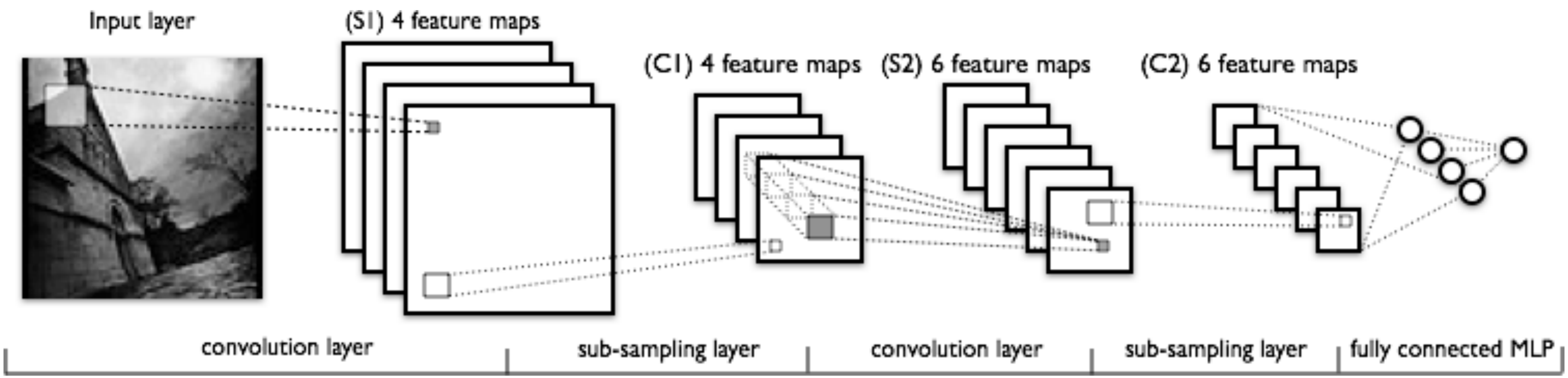
CNN

Subsampling layer



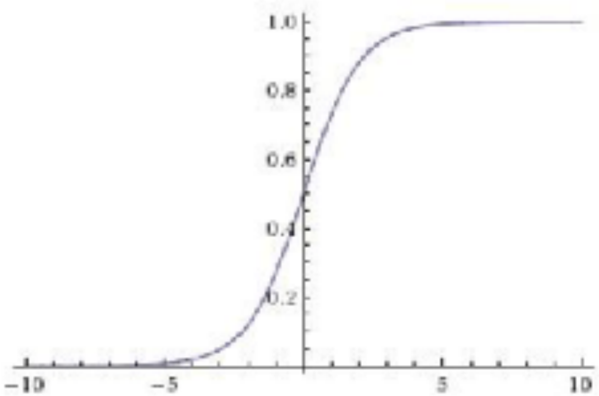
[image from http://deeplearning.net/tutorial/lenet.html]

Convolutional Neural Networks (CNN/LeNet) for general image feature extraction

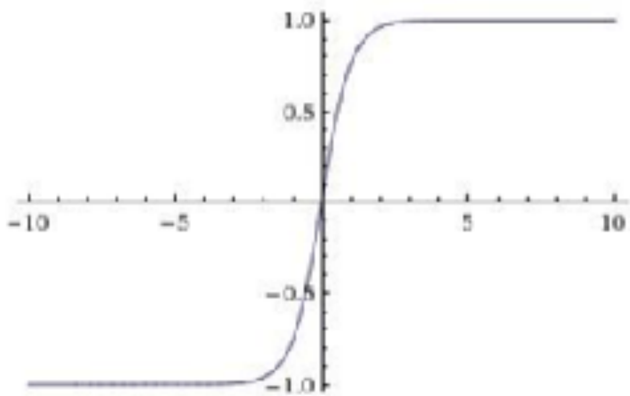


[image from http://deeplearning.net/tutorial/lenet.html]

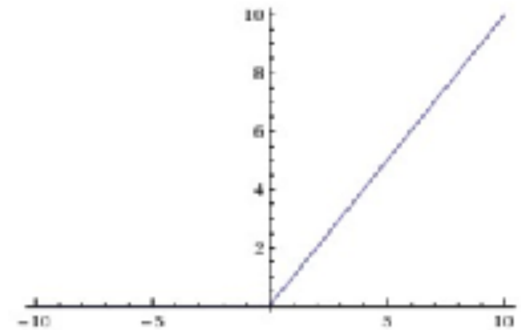
Activation functions (con't)



Sigmoid



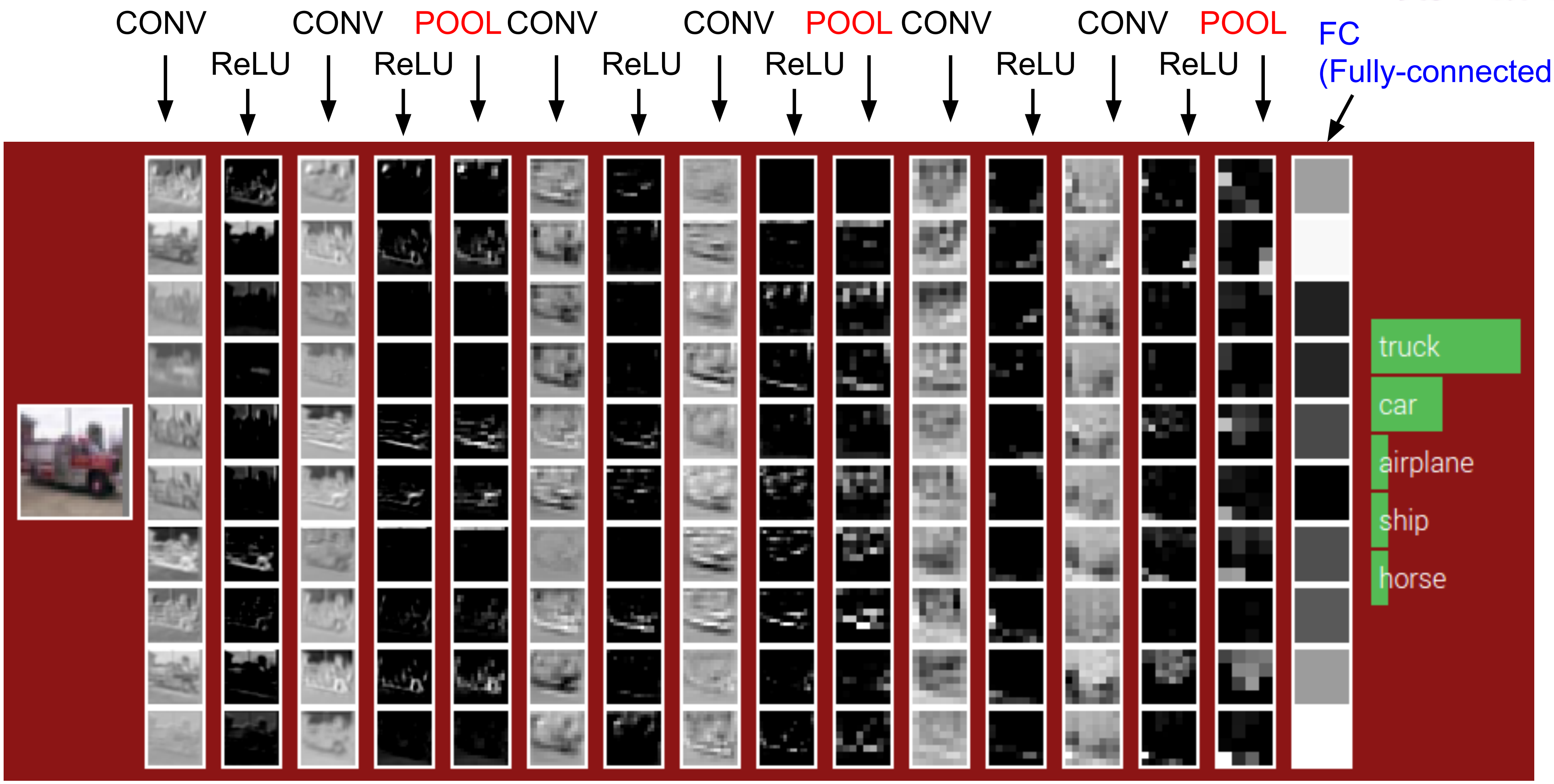
tanh(x)



ReLU
(Rectified Linear Unit)

And many more …

CNN


CONV
ReLU
CONV
ReLU
POOL
CONV
ReLU
CONV
ReLU
POOL
CONV
ReLU
CONV
ReLU
POOL
FC
(Fully-connected)
truck
car
airplane
ship
horse

CNN Tricks

Must Know Tips/Tricks in Deep Neural Networks (by Xiu-Shen Wei)

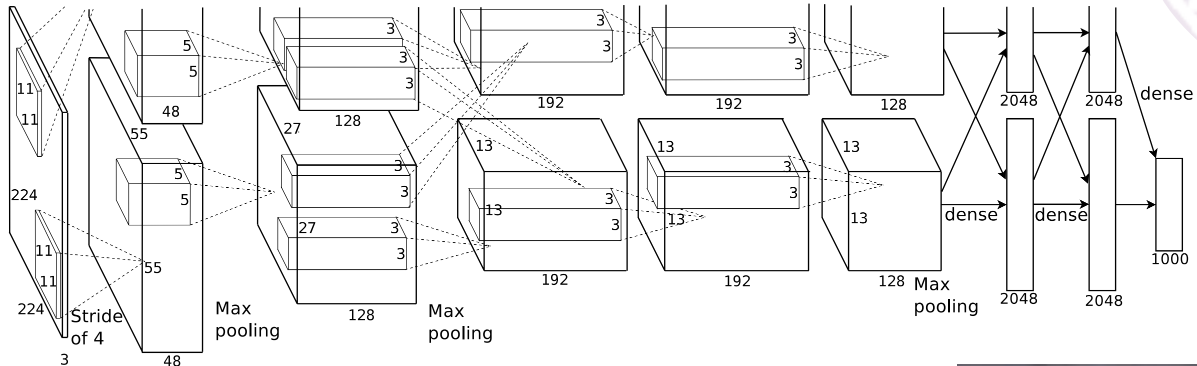
Deep Neural Networks, especially ***Convolutional Neural Networks*** (*CNN*), allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These methods have dramatically improved the state-of-the-arts in visual object recognition, object detection, text recognition and many other domains such as drug discovery and genomics.

In addition, many solid papers have been published in this topic, and some high quality open source CNN software packages have been made available. There are also well-written CNN tutorials or CNN software manuals. However, it might lack a recent and comprehensive summary about the details of how to implement an excellent deep convolutional neural networks from scratch. Thus, we collected and concluded many implementation details for DCNNs. **Here we will introduce these extensive implementation details, i.e., *tricks* or *tips*, for building and training your own deep networks.**

- ☑ Data augmentation
- ☑ Pre-processing
- ☑ Initializations
- ☑ During training
- ☑ Activation functions
- ☑ Regularizations
- ☑ Insights from figures
- ☑ Ensemble

http://lamda.nju.edu.cn/weixs/project/CNNTricks/CNNTricks.html

CNN



4.94% (DL) vs 5.1% (human)

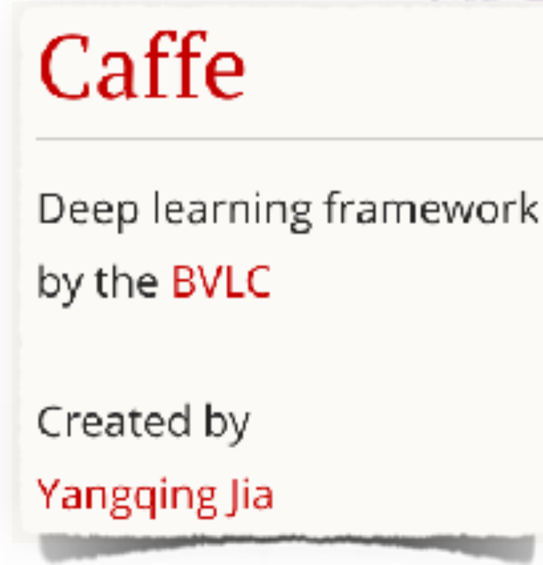
Geoffrey E. Hinton
University of Toronto

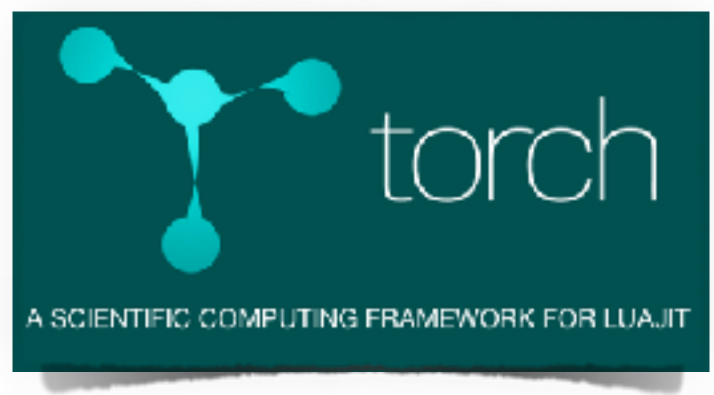
IMAGENET

Fei-Fei Li
Stanford University

CNN toolbox

- MatConvNet (Oxford University)
- Caffe (UC Berkeley)
- Torch (Facebook & NYU)
- …



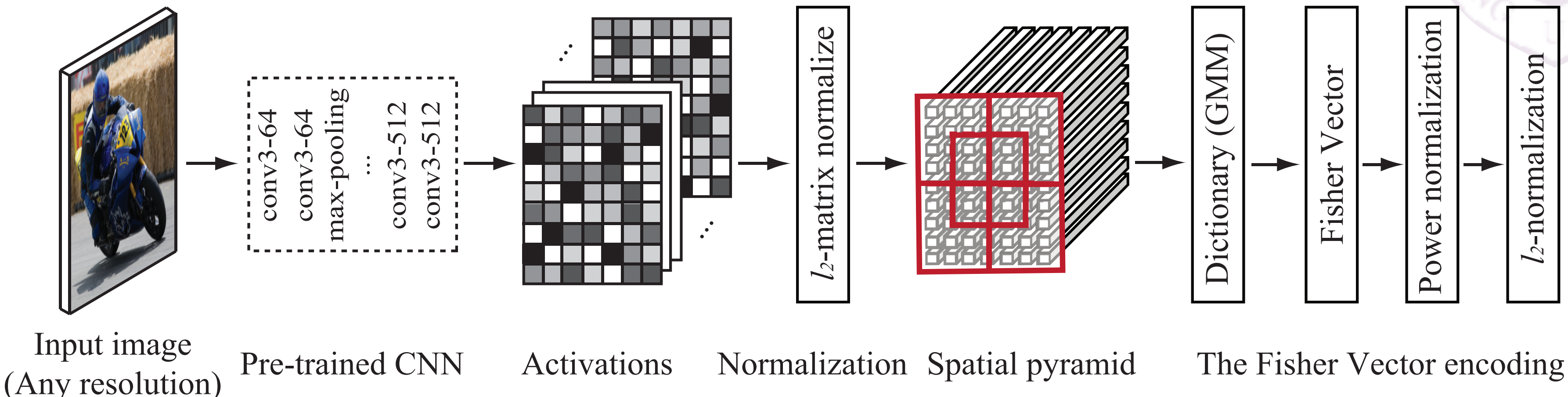


deeplearning.net

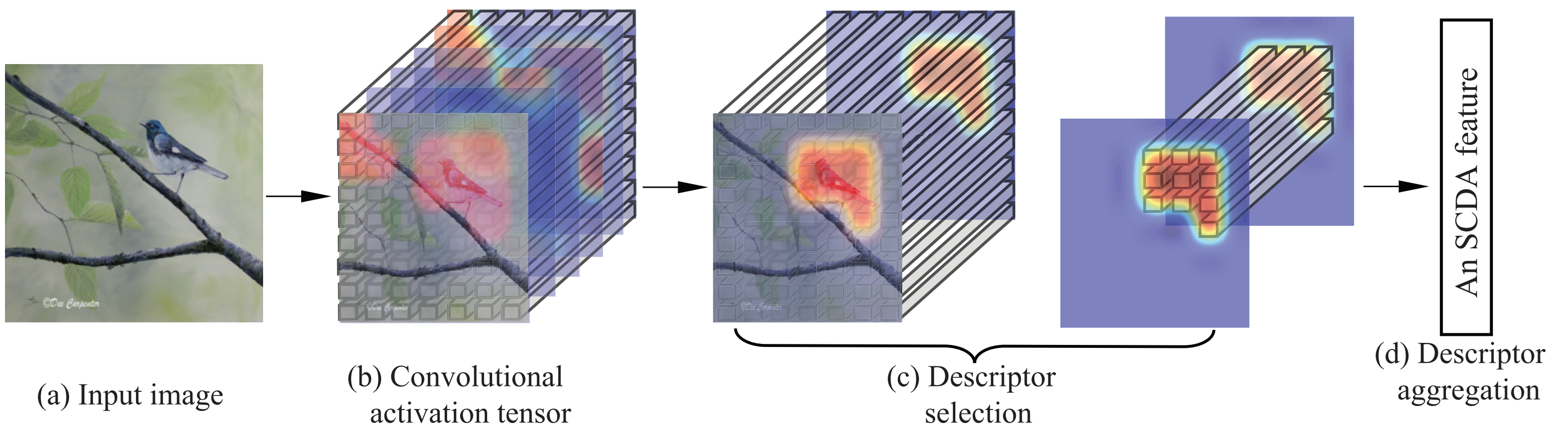
NVIDIA-GPUs


DEEP LEARNING
NVIDIA Home > Products > NVIDIA DGX-1
Subscribe
NVIDIA DGX-1
Deep Learning System
THE WORLD'S FIRST DEEP LEARNING
SUPERCOMPUTER IN A BOX

Some Applications

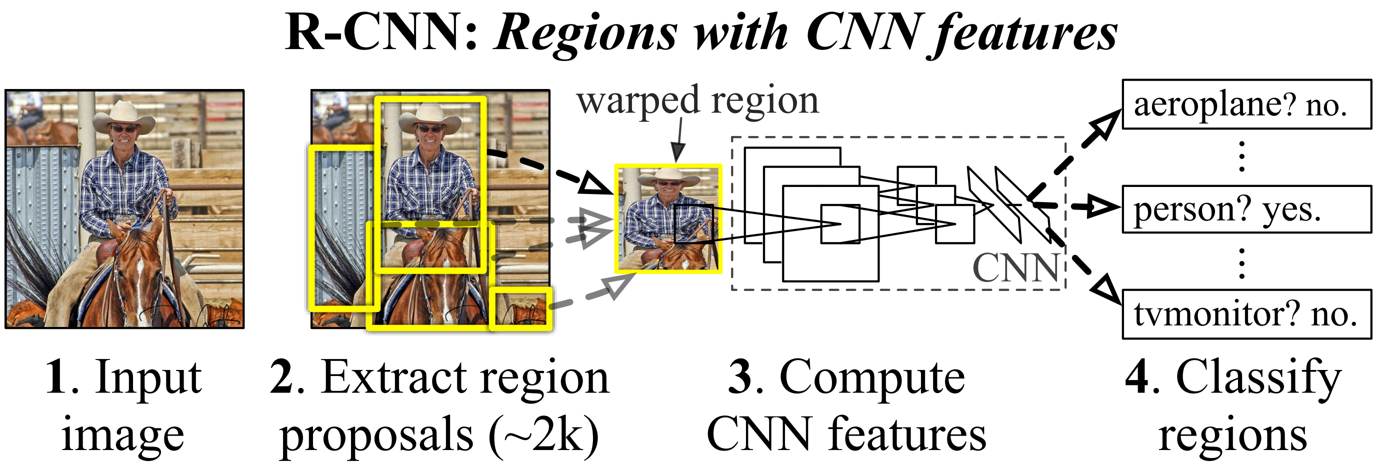


Pre-trained model as feature extractor

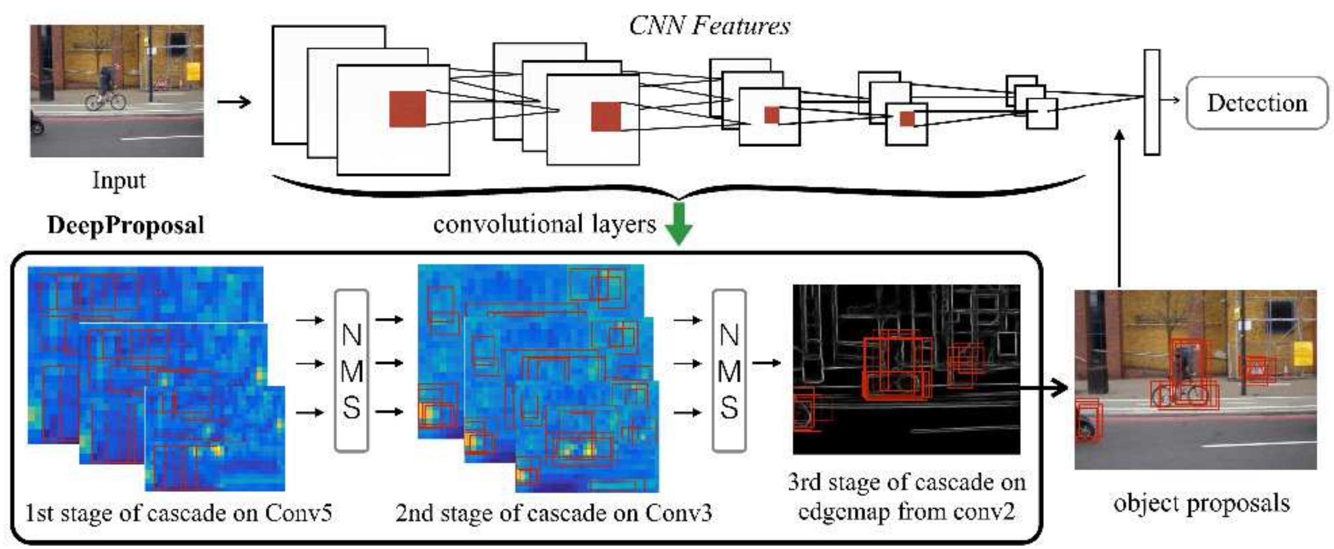


Fine-grained image retrieval

Some Applications

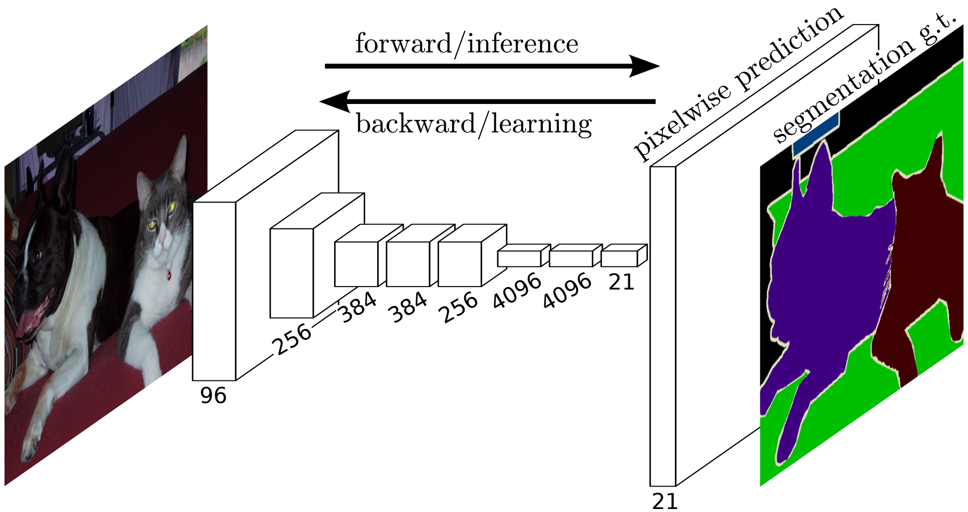


Object detection

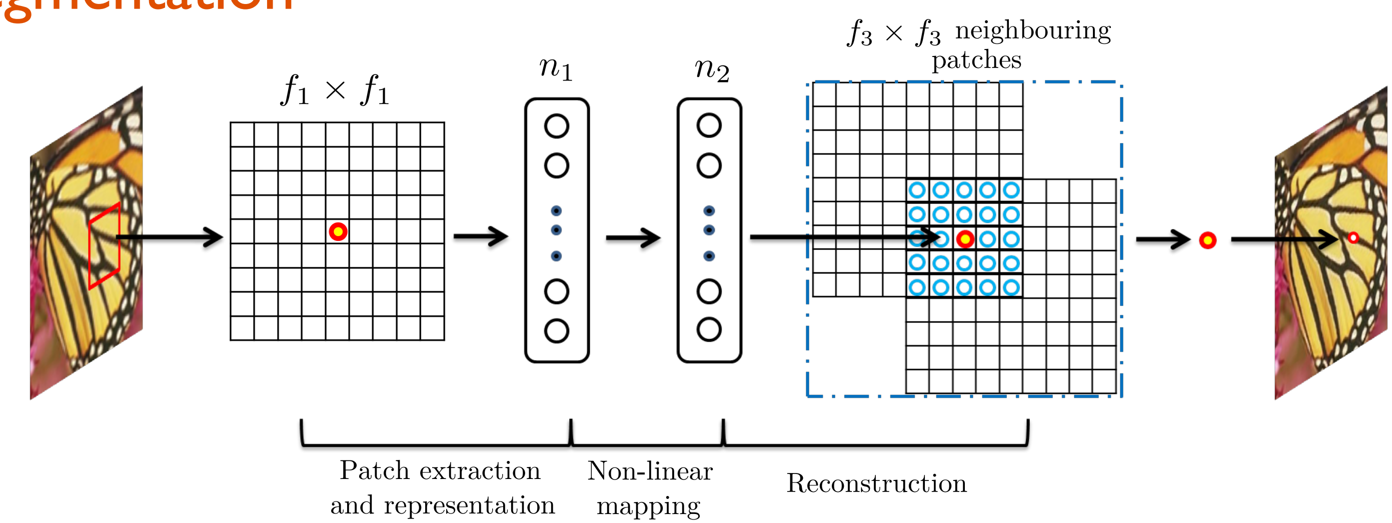


DeepProposal

Some Applications

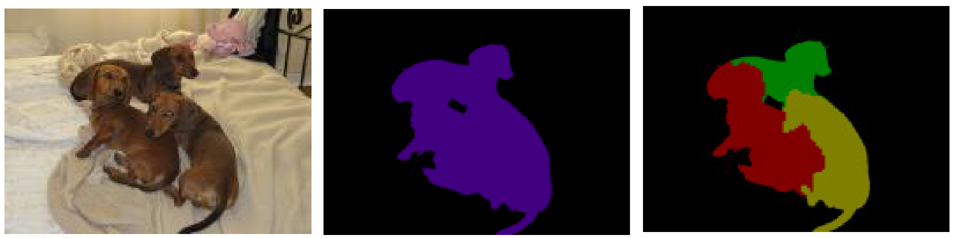


Semantic segmentation



Super-resolution

Some Applications

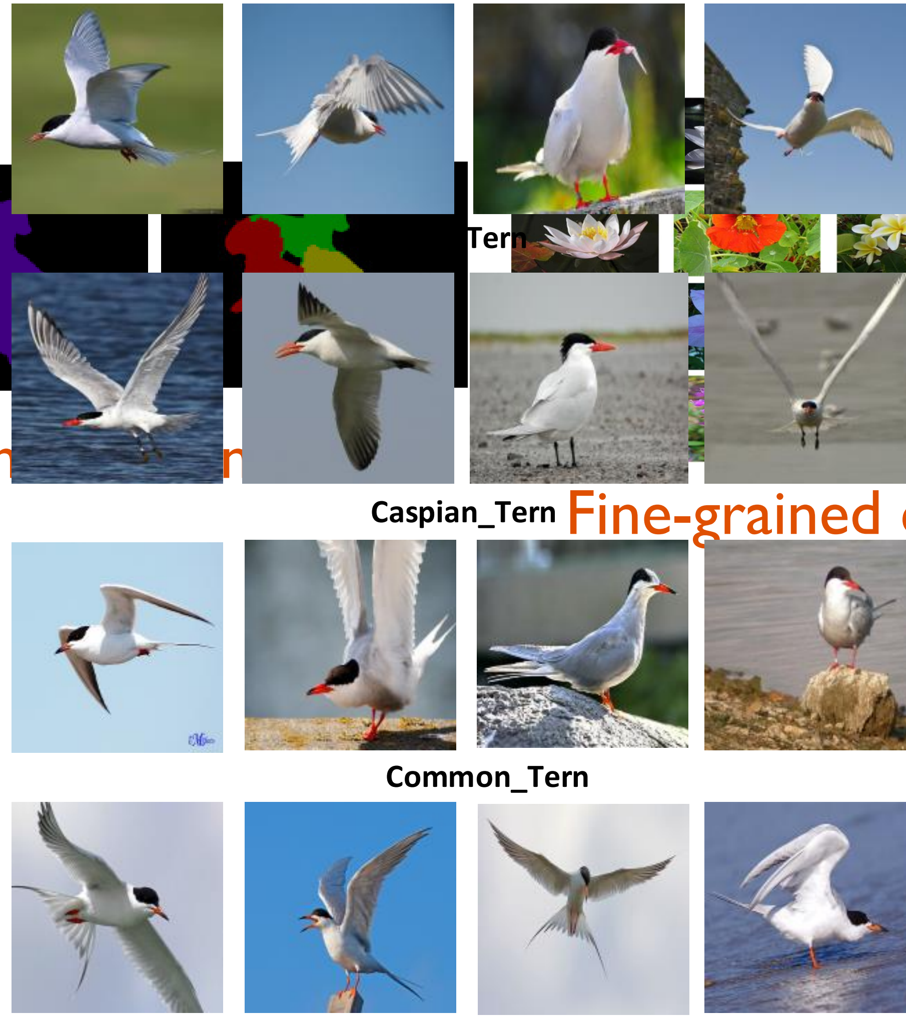


Object segmentation

Fine-grained classification

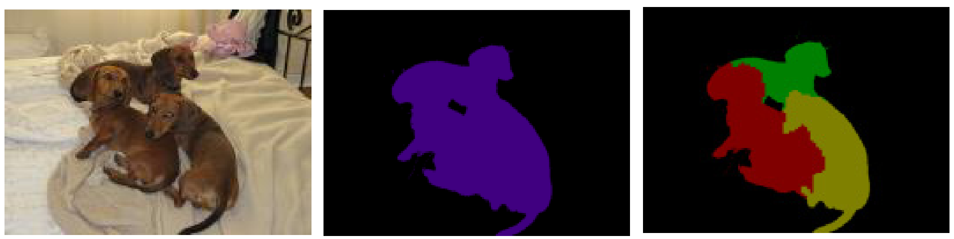
Some Applications

Object segm[...]



Fine-grained classification

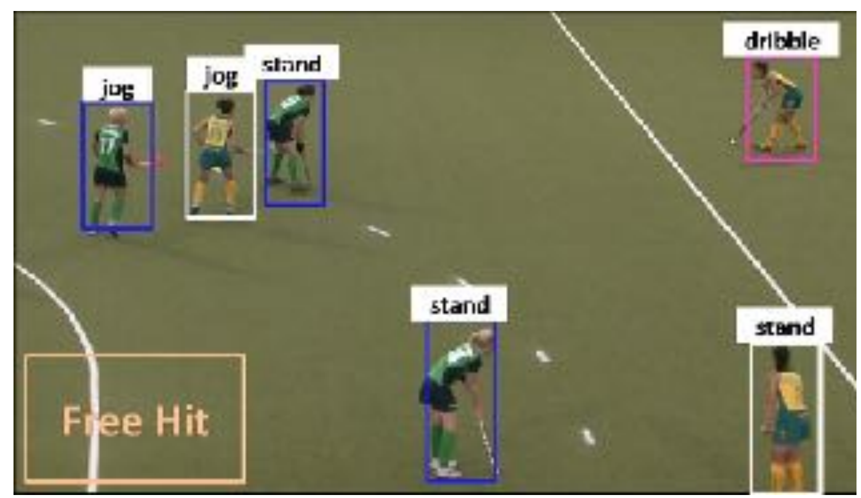
Some Applications



Object segmentation

Fine-grained classification

Face recognition



Action recognition

Some Applications

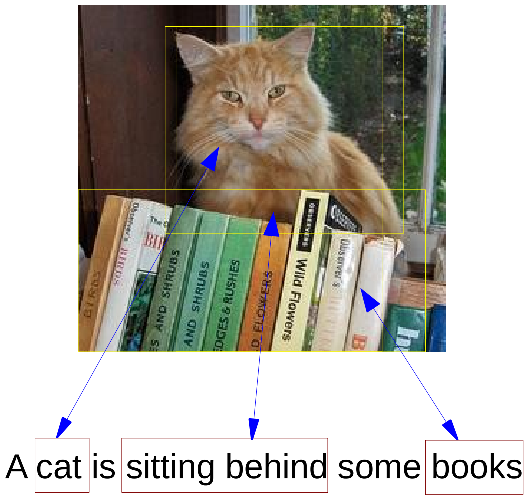


Image caption

Automatic driving

Cultural event recognition

Some Applications

Multimodal Linguistic Regularities

[Kiros et al., TACL 2015]

Some Applications

Multimodal Linguistic Regularities

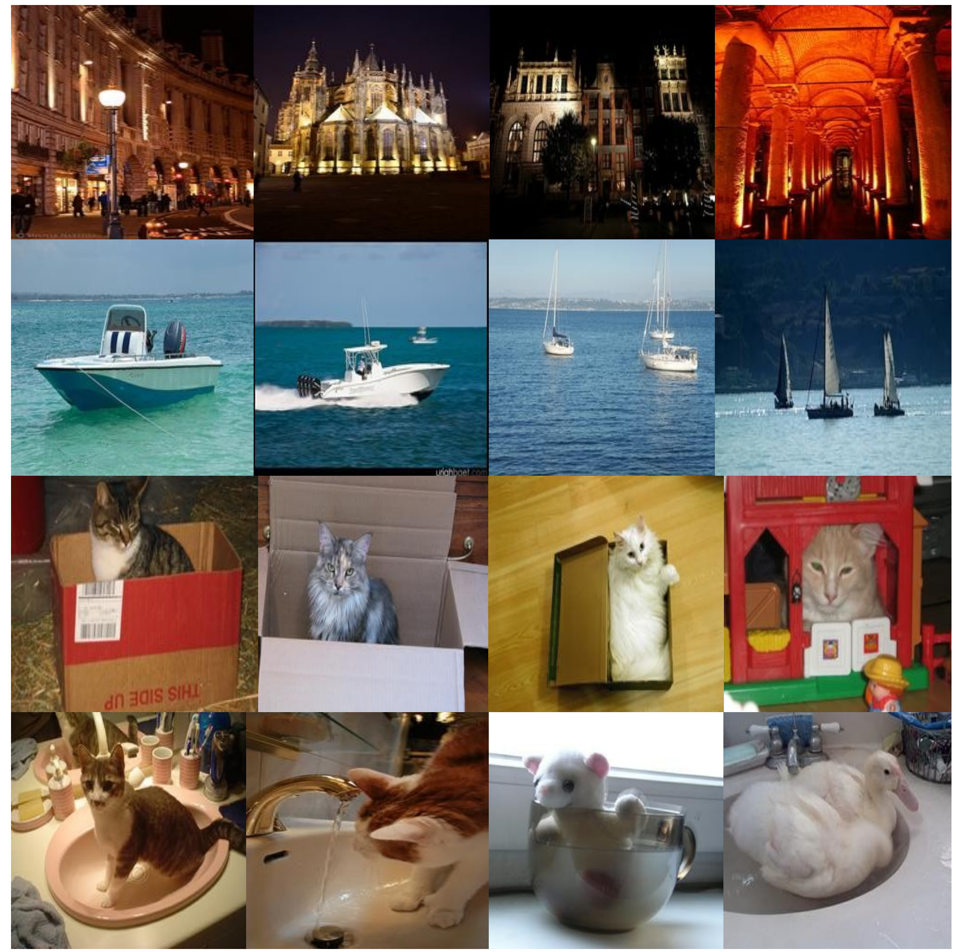
- day + night =

- flying + sailing =

- bowl + box =

- box + bowl =

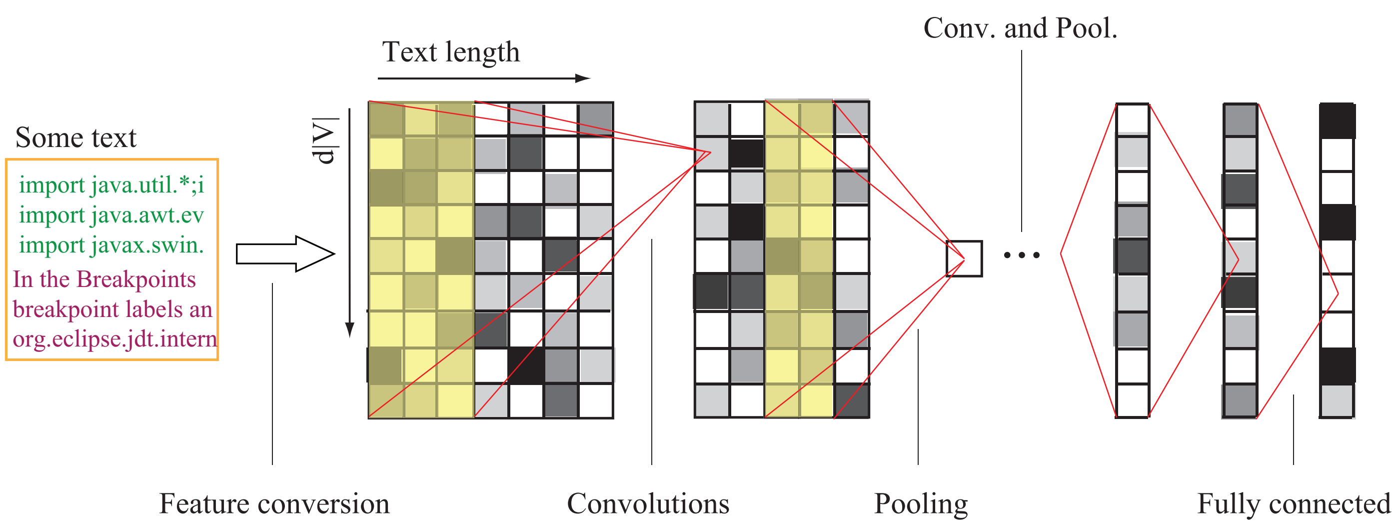
Nearest images



[Kiros et al., TACL 2015]

Some Applications：NLP

How does CNN apply to NLP?



Effective Use of Word Order for Text Categorization with Convolutional Neural Networks

Rie Johnson
RJ Research Consulting
Tarrytown, NY, USA
riejohnson@gmail.com

Tong Zhang
Baidu Inc., Beijing, China
Rutgers University, Piscataway, NJ, USA
tzhang@stat.rutgers.edu

Character-level Convolutional Networks for Text Classification*

Xiang Zhang **Junbo Zhao** **Yann LeCun**
Courant Institute of Mathematical Sciences, New York University
719 Broadway, 12th Floor, New York, NY 10003
{xiang, junbo.zhao, yann}@cs.nyu.edu

Transformation for text

One-hot encoding

"I love it." $\longrightarrow$ $\mathbf{x} = [\,0\;0\;1\;0\;0\,|\,0\;0\;0\;0\;1\,|\,0\;0\;0\;1\;0\,]^\top$

V={"don't", "hate", "I", "it", "love"}

Transformation for text

Seq-CNN for text

"I love it." $\longrightarrow$ $$\mathbf{r}_0(\mathbf{x}) = \left[\begin{array}{c} 0 \\ 0 \\ \mathbf{1} \\ 0 \\ 0 \\ \text{—} \\ 0 \\ 0 \\ 0 \\ 0 \\ \mathbf{1} \end{array}\right] \begin{array}{c} \text{don't} \\ \text{hate} \\ \textbf{I} \\ \text{it} \\ \text{love} \\ \\ \text{don't} \\ \text{hate} \\ \text{I} \\ \text{it} \\ \textbf{love} \end{array} \qquad \mathbf{r}_1(\mathbf{x}) = \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ \mathbf{1} \\ \text{—} \\ 0 \\ 0 \\ 0 \\ \mathbf{1} \\ 0 \end{array}\right] \begin{array}{c} \text{don't} \\ \text{hate} \\ \text{I} \\ \text{it} \\ \textbf{love} \\ \\ \text{don't} \\ \text{hate} \\ \text{I} \\ \textbf{it} \\ \text{love} \end{array}$$

V={"don't", "hate", "I", "it", "love"}

Transformation for text

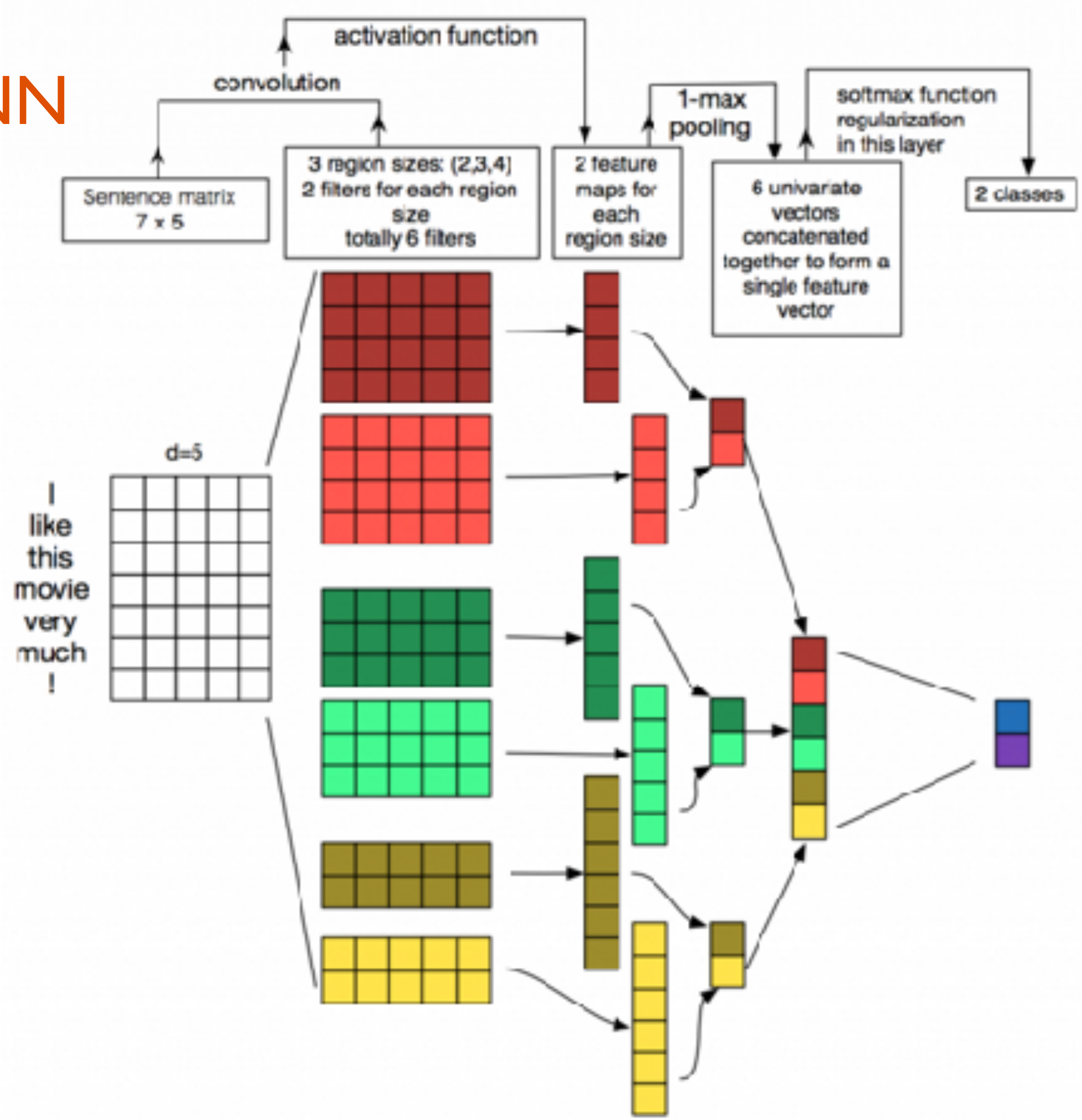
bow-CNN for text

"I love it." $\longrightarrow$ $\mathbf{r}_0(\mathbf{x}) = \begin{array}{cc} \begin{bmatrix} 0 \\ 0 \\ \mathbf{1} \\ 0 \\ \mathbf{1} \end{bmatrix} & \begin{array}{c} \text{don't} \\ \text{hate} \\ \mathbf{I} \\ \text{it} \\ \textbf{love} \end{array} \end{array}$ $\mathbf{r}_1(\mathbf{x}) = \begin{array}{cc} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \mathbf{1} \\ \mathbf{1} \end{bmatrix} & \begin{array}{c} \text{don't} \\ \text{hate} \\ \text{I} \\ \textbf{it} \\ \textbf{love} \end{array} \end{array}$

V={"don't", "hate", "I", "it", "love"}

More for text

Shallow-CNN



More for text

Deep-CNN

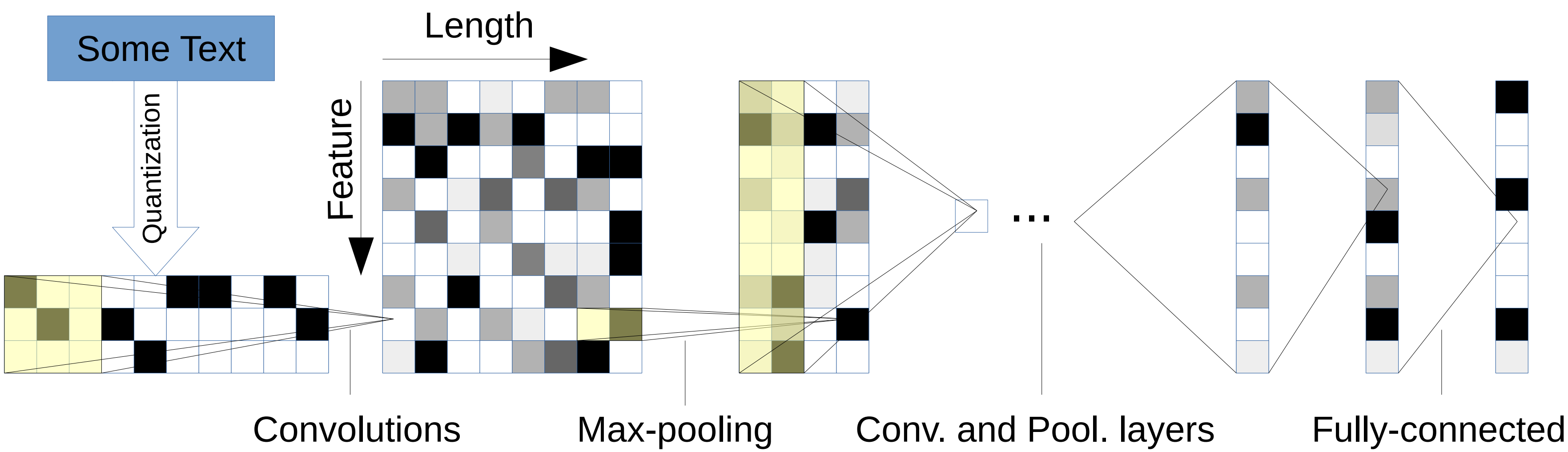


Figure courtesy of [Xiang Zhang et. al, NIPS' 15]