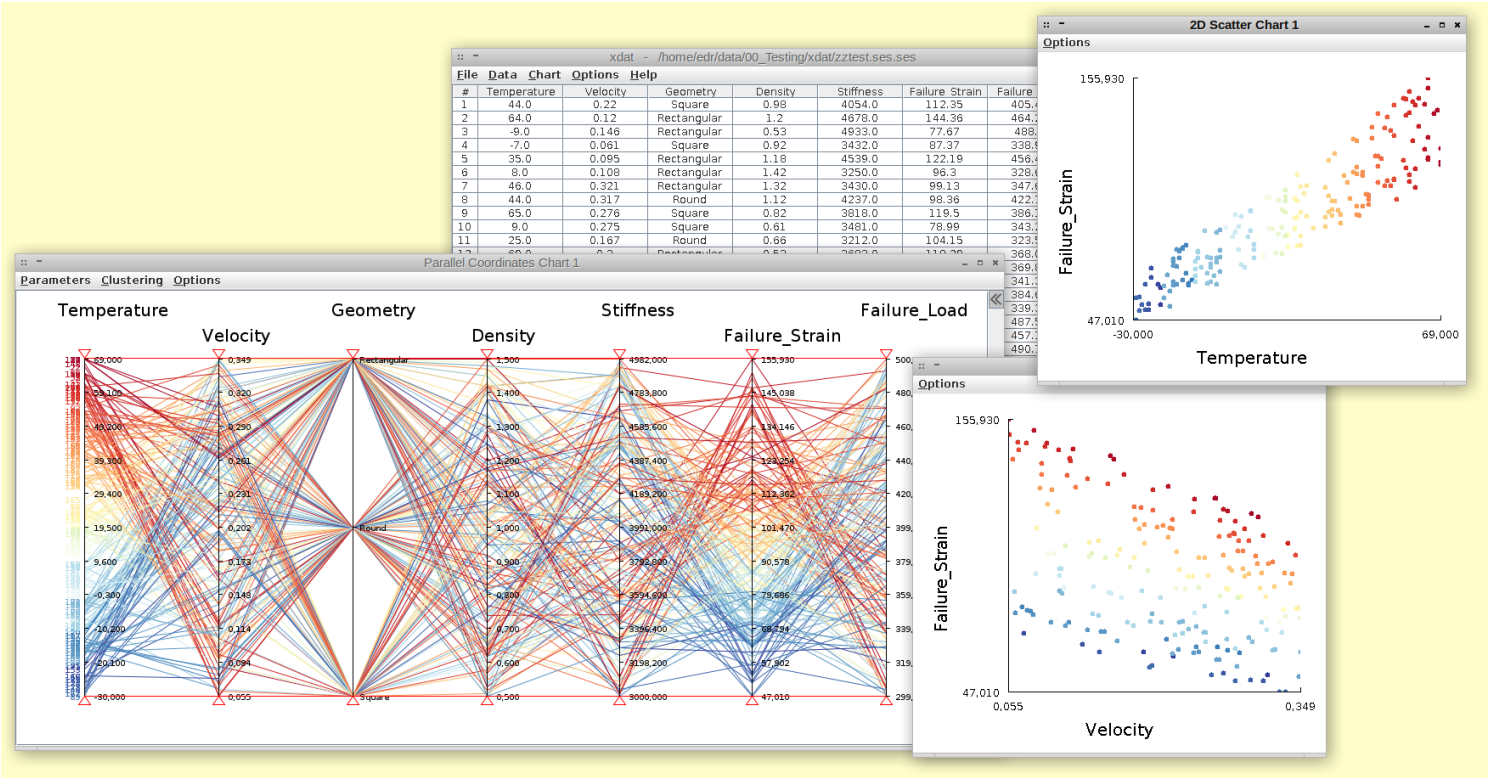
Introduction to Deep Learning

Alvaro Soto

Computer Science Department, PUC

Ingredient 1: Feature learning

Feature space



Handcrafted Features

deeper neural networks enable us to approximate functions of certain classes with fewer parameters (

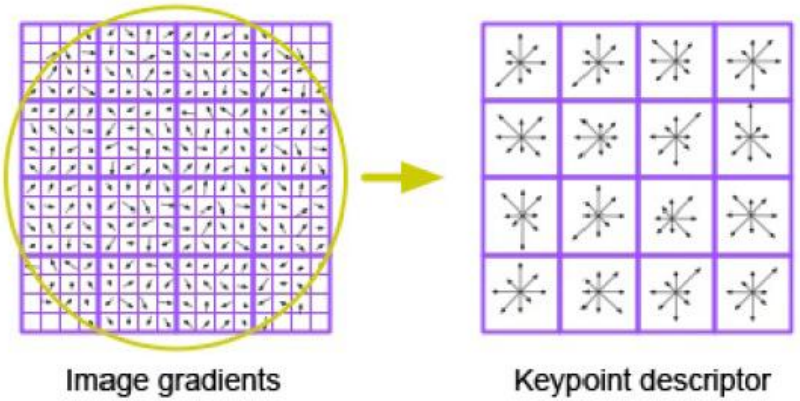
d1 : Mary loves Movies, Cinema and Art
d2 : John went to the Football game

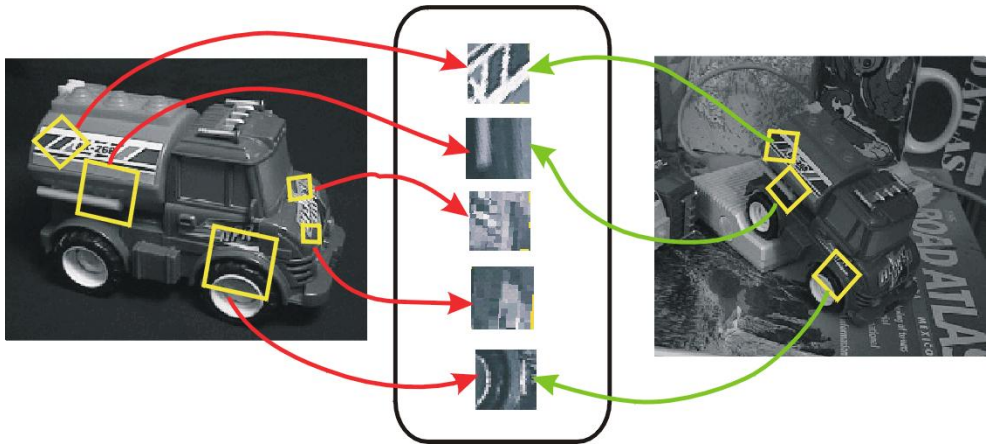
d3 : Robert went for the Movie Delicatessen

Class 1 : Arts
Class 2 : Sports

Class : Arts

	Mary	Loves	Movies	Cinema	Art	John	Went	to	the	Delicatessen	Robert	Football	Game	and	for
d1	1	1	1	1	1									1	
d2						1	1	1	1			1	1		
d3			1				1		1		1				1

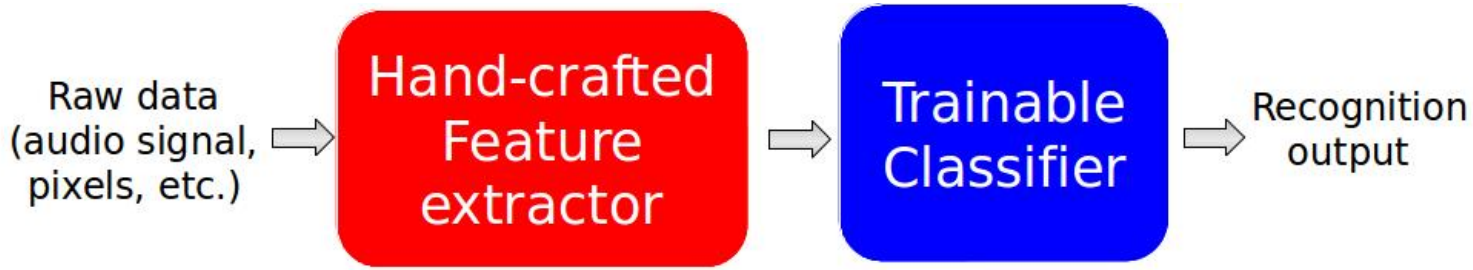




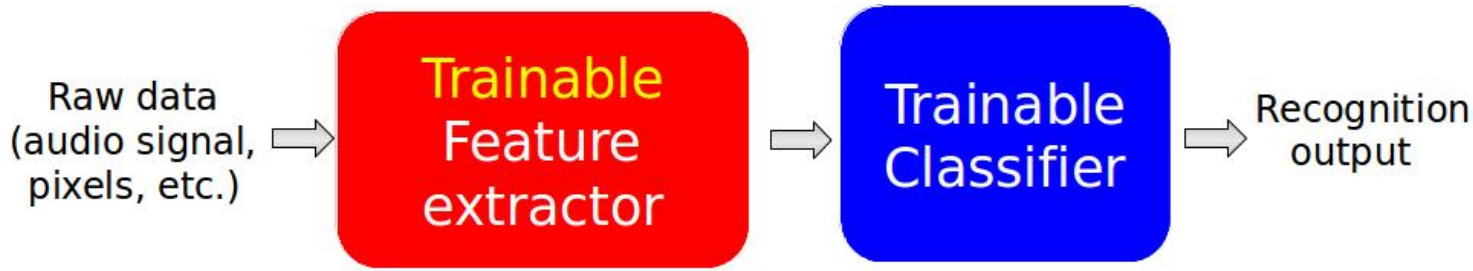
Feature Learning

Discriminative vs semantic feats

- Traditional pattern recognition approach since the 50's



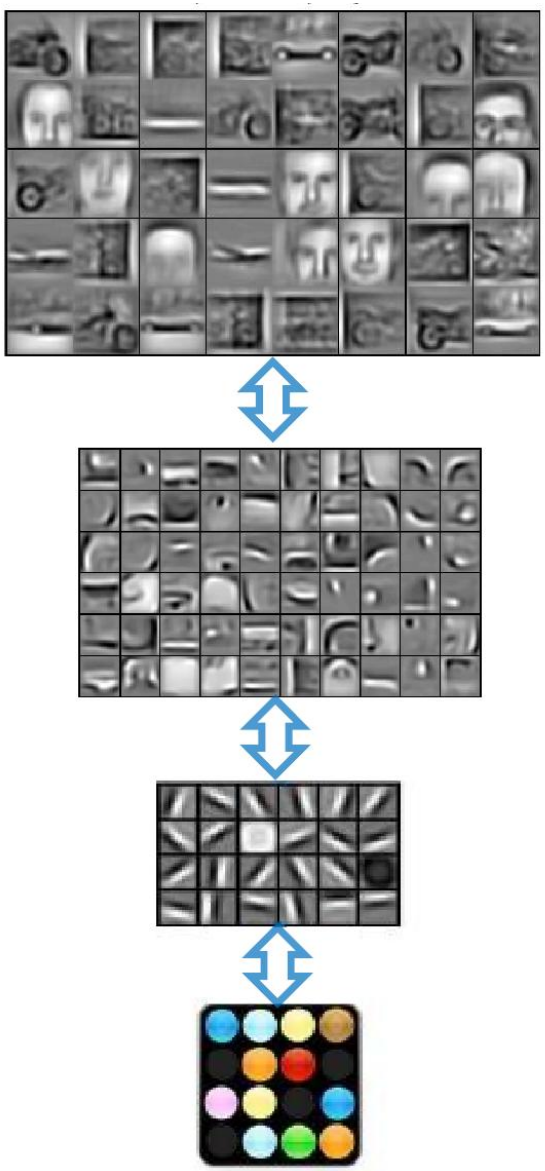
- Feature learning approach



Ingredient 2: A hierarchical compositional representation

Hierarchical and Composable Feature Representations

- Hierarchical representations organized from low to high levels of abstraction are common in nature.
- Compositionality is key. Language is compositional, vision is compositional, etc.
- Examples:
  - Sounds $\rightarrow$ phonemes $\rightarrow$ syllables $\rightarrow$ words.
  - Pixels $\rightarrow$ edges $\rightarrow$ parts $\rightarrow$ objects
  - Characters $\rightarrow$ words $\rightarrow$ word groups $\rightarrow$ clauses $\rightarrow$ sentences.
  - Nucleotides $\rightarrow$ genes $\rightarrow$ proteins $\rightarrow$ cells $\rightarrow$ organs.



3rd layer
“Objects”

2nd layer
“Object parts”

1st layer
“edges”

Input

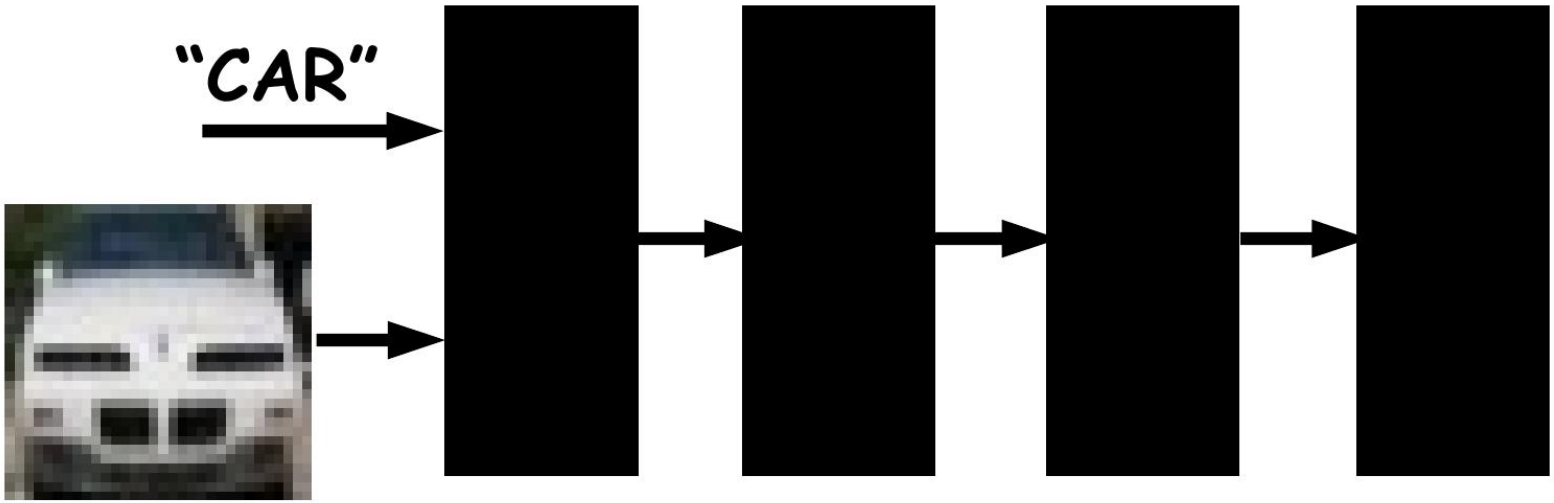
Now our honor guest:
Deep Learning

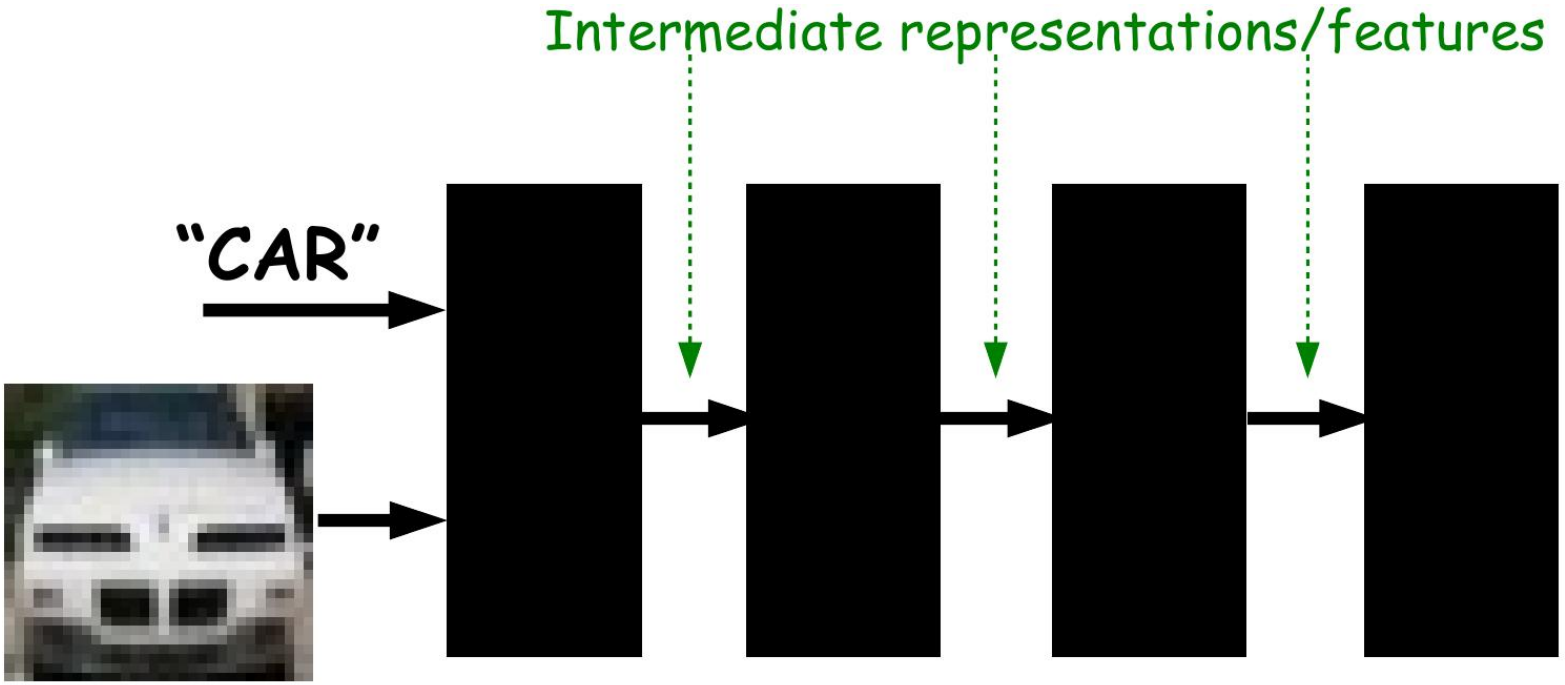
Deep Learning (DL)

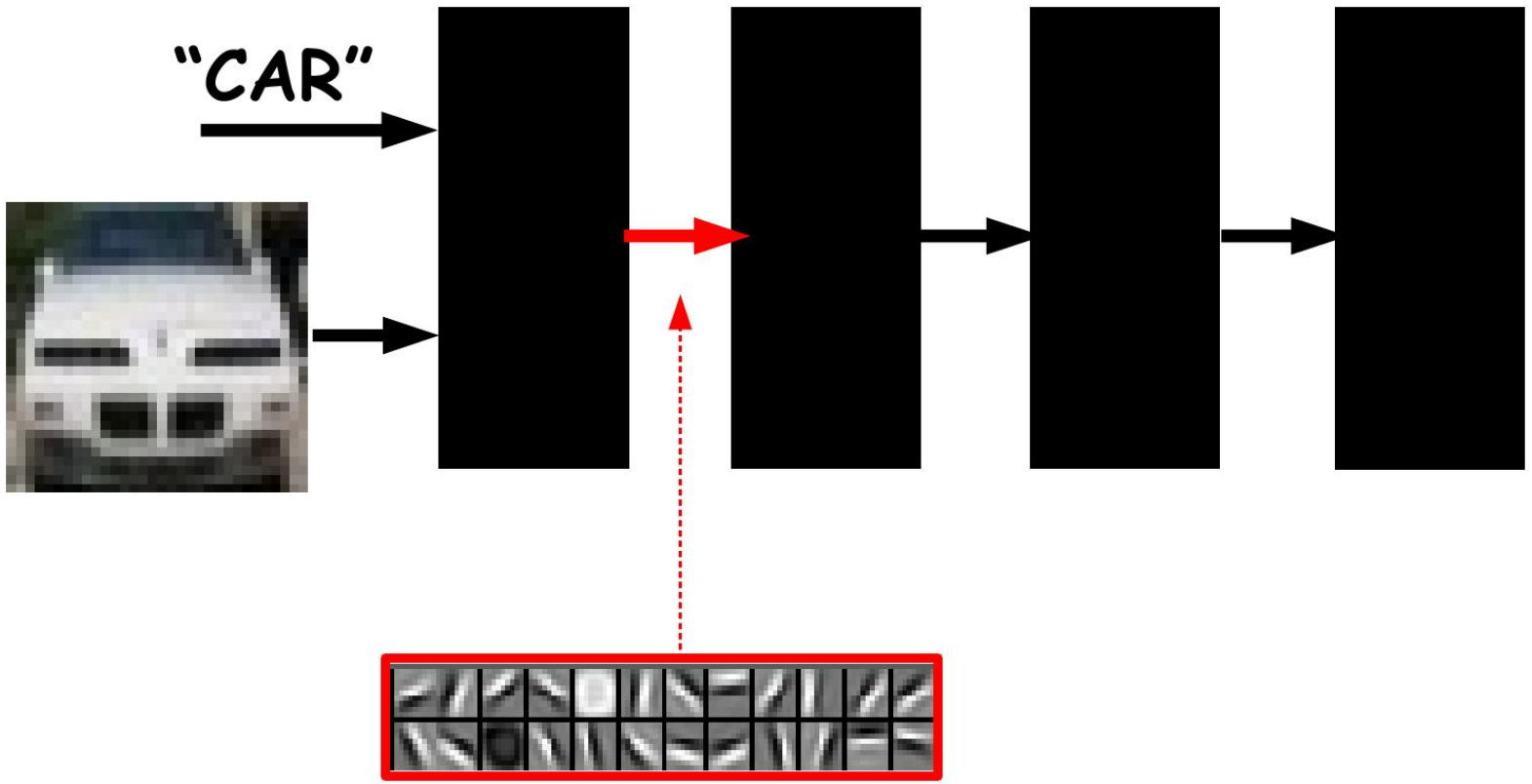
- Deep learning: Neural network (NN) models that learn hierarchical compositional representations.
- In general, NNs with several hidden layers.
- Today, the most successful learning framework. They hold records for best recognition performance on several difficult tasks, such as, voice, handwritting, and object recognition.
- What is the secret?

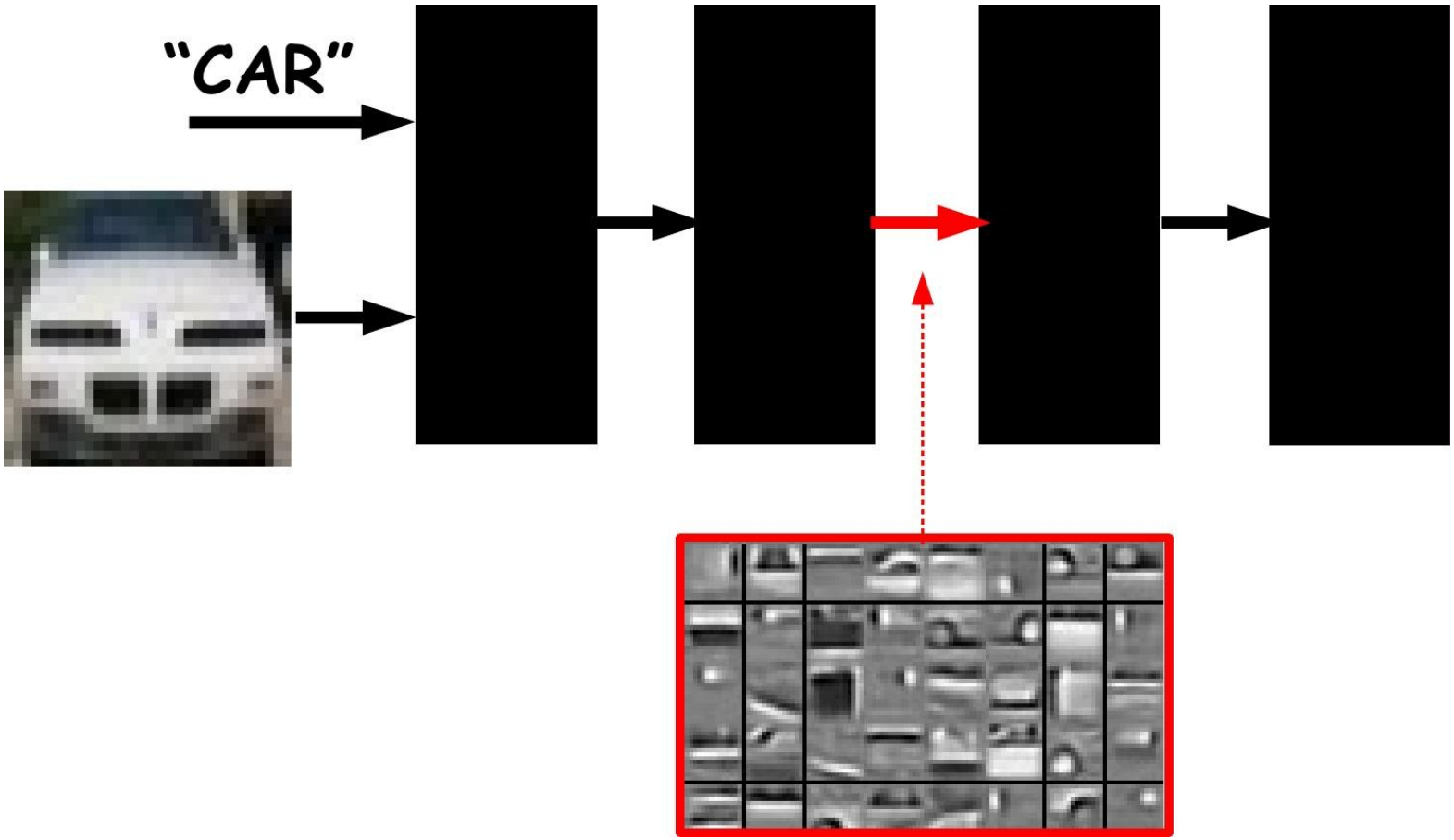
The magic is learning suitable hierarchical and composable feature representations

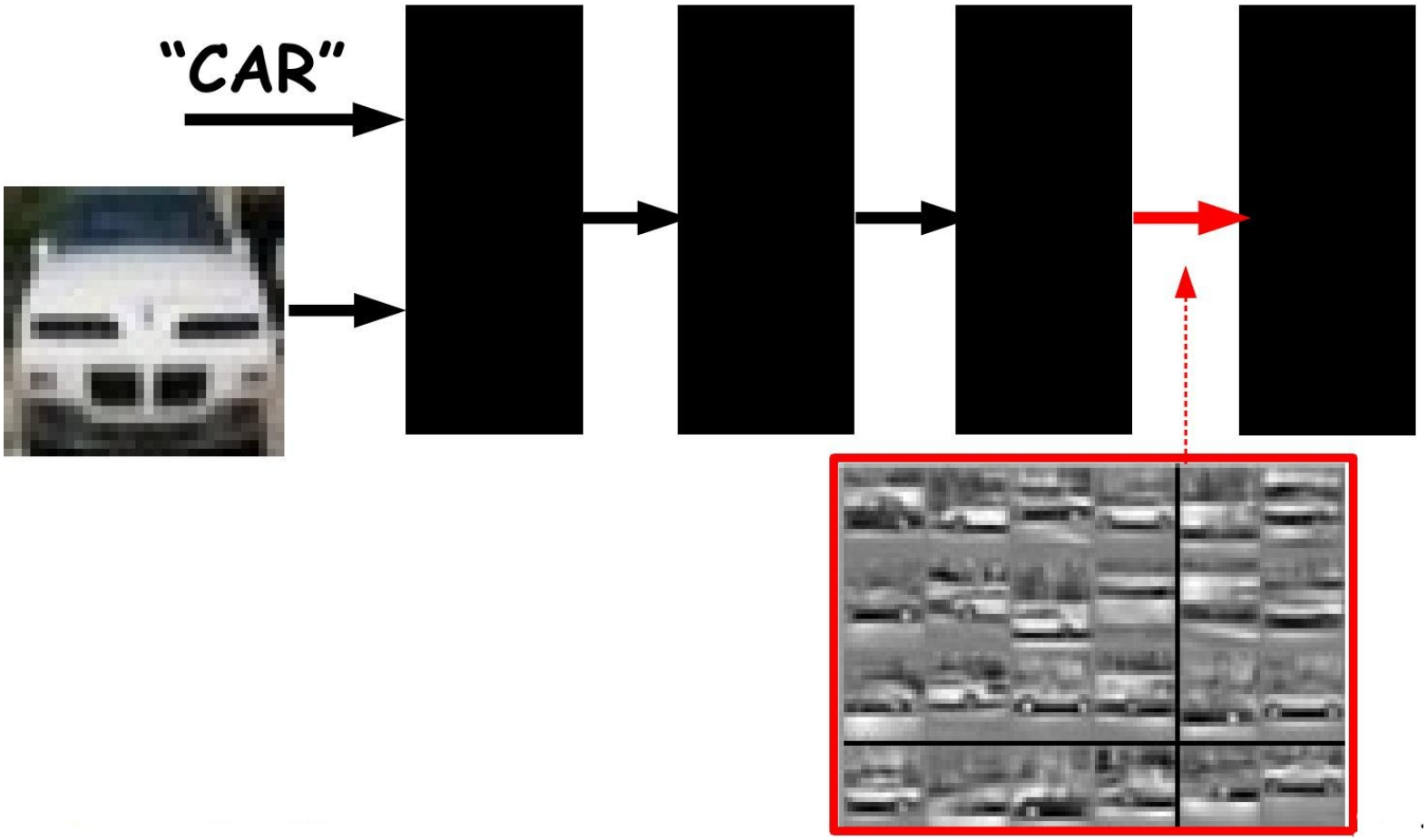

"CAR"


"CAR"

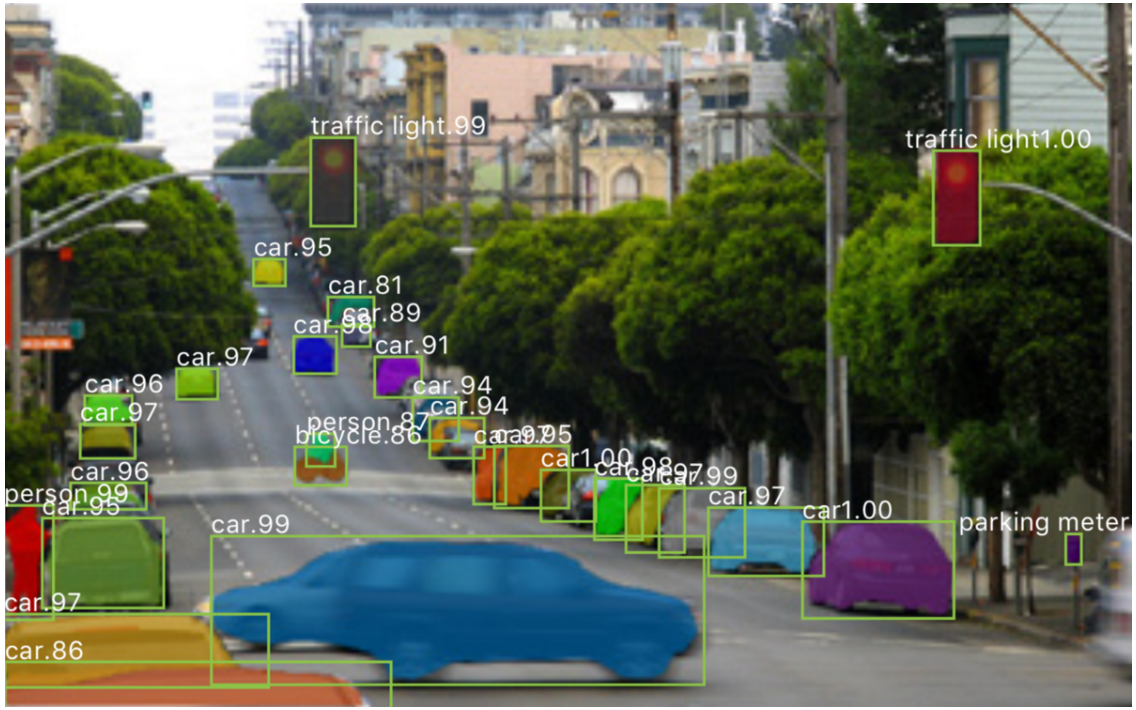

Intermediate representations/features
"CAR"


"CAR"


"CAR"

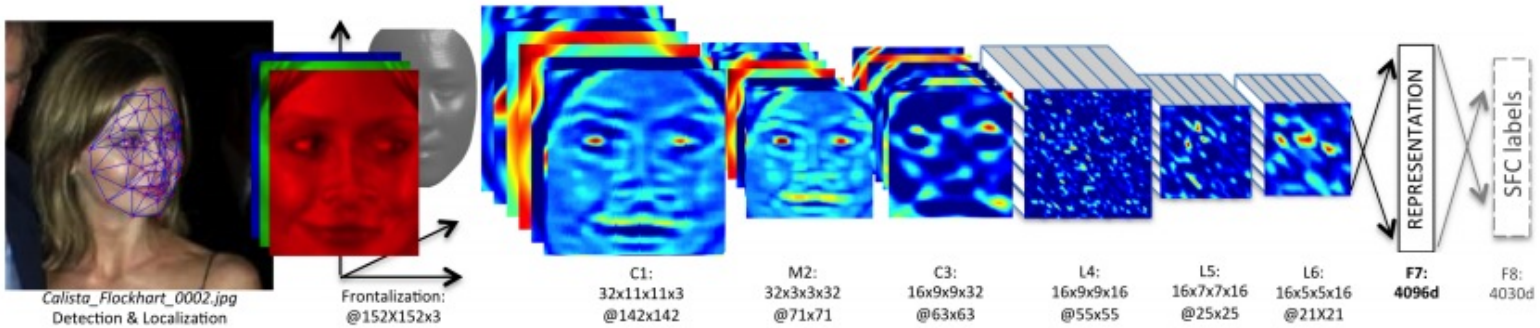

"CAR"

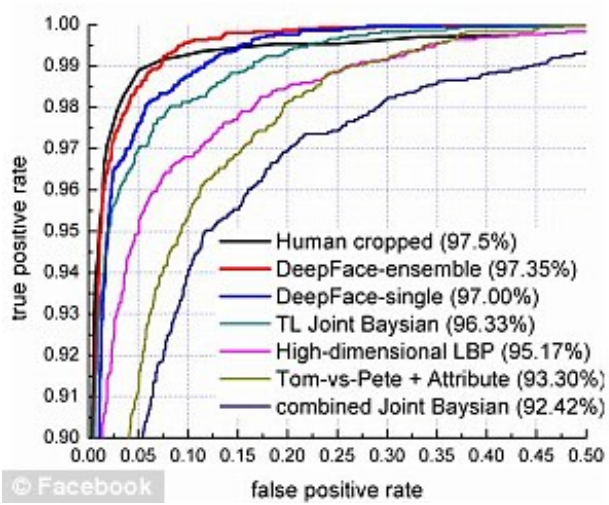
DL in Action



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.

DL records


Calista_Flockhart_0002.jpg
Detection & Localization
Frontalization:
@152X152x3
C1:
32x11x11x3
@142x142
M2:
32x3x3x32
@71x71
C3:
16x9x9x32
@63x63
L4:
16x9x9x16
@55x55
L5:
16x7x7x16
@25x25
L6:
16x5x5x16
@21X21
F7:
4096d
F8:
4030d
REPRESENTATION
SFC labels


true positive rate
false positive rate
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.92
0.91
0.90
0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50
Human cropped (97.5%)
DeepFace-ensemble (97.35%)
DeepFace-single (97.00%)
TL Joint Baysian (96.33%)
High-dimensional LBP (95.17%)
Tom-vs-Pete + Attribute (93.30%)
combined Joint Baysian (92.42%)
© Facebook

Deep Learning (DL)

- What is the secret?

The magic is learning suitable hierarchical and composable feature representations

Why Deep?

Theoretician's dilemma:

We can approximate any function as close as we want using a shallow architecture. Why do we need deep ones ?.

$$y = F(< W^1, F(< W^0, X >) >) \tag{1}$$

V/S

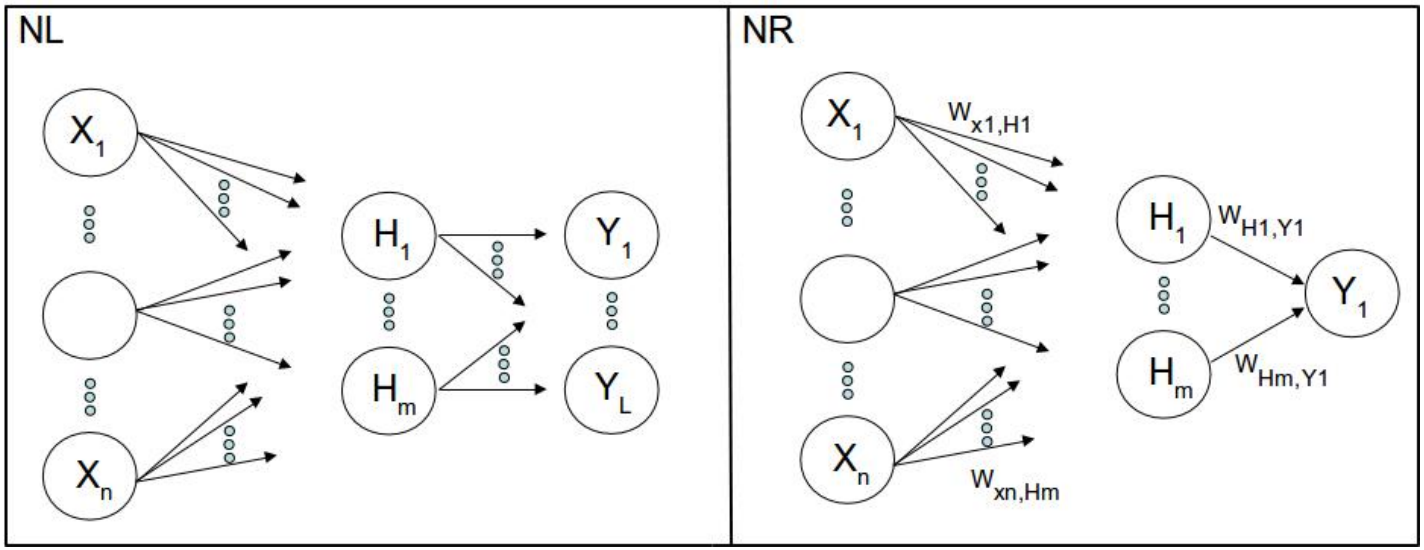
$$y = F(< W^K, F(W^{K-1} \dots F(< W^1, F(< W^0, X >) >))) \tag{2}$$

Solution: Learning Efficiency

A hierachical compositional architecture provides an efficient learning framework, where lower level layers are shared by higher level layers.

A Shared Low Level Representation

Consider the following two configurations of a NN.



- Suppose to want to learn function to map $X \in \Re^n$ to $Y \in \Re^L$.
- Can we use the previous configurations?,
- What is the main difference?

Main Challenges

- How to adjust so many parameters without overfitting.
- We require lot of training data, even worse: labeled data (supervised learning).
- Tons of labeled data, How?
  - Use a creative, cheap, and/or fast way to label data.
  - In some cases it is possible to use unlabeled data (unsupervised learning/ autoencoders/Gans).
  - Transfer learning.
  - Apply training tricks (add noise, synthetic data, whitening, drop-out, etc.).
- Difficult optimization and scalability: High performance computing (GPUs).