Applied Deep Learning

Introduction

Sep 22nd, 2016

YUN-NUNG (VIVIAN) CHEN

WWW.CSIE.NTU.EDU.TW/~YVCHEN/F105-ADL

National Taiwan University

Slide credit from Prof. Hung-Yi Lee

What is Machine Learning?

What Computers Can Do?

Programs can do the things you ask them to do

Program for Solving Tasks

Task: predicting positive or negative given a product review

"I love this product!" → program.py → +
if input contains "love", "like", etc.
output = positive

"It claims too much." → program.py → -
if input contains "too much", "bad", etc.
output = negative

"It's a little expensive." → program.py → ?

"台灣第一波上市!" → program.py → 推

"規格好雞肋..." → program.py → 噓

"樓下買了我才考慮" → program.py → ?

Some tasks are complex, and we don't know how to write a program to solve them.

Learning ≈ Looking for a Function

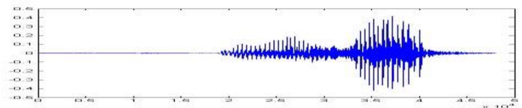
Task: predicting positive or negative given a product review

“I love this product!”	“It claims too much.”	“It’s a little expensive.”
↓ *f*	↓ *f*	↓ *f*
+	**-**	**?**
“台灣第一波上市!”	“規格好雞肋...”	“樓下買了我才考慮”
↓ *f*	↓ *f*	↓ *f*
推	**噓**	**?**

Given a large amount of data, the machine learns what the function ***f*** should be.

Learning ≈ Looking for a Function

Speech Recognition

$f($  $) =$ “你好”

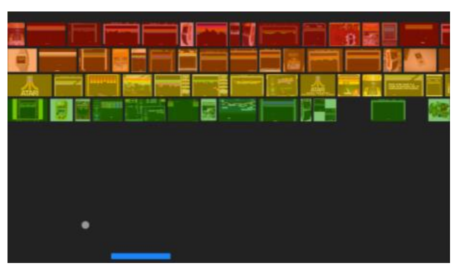
Handwritten Recognition

$f($  $) =$ “2”

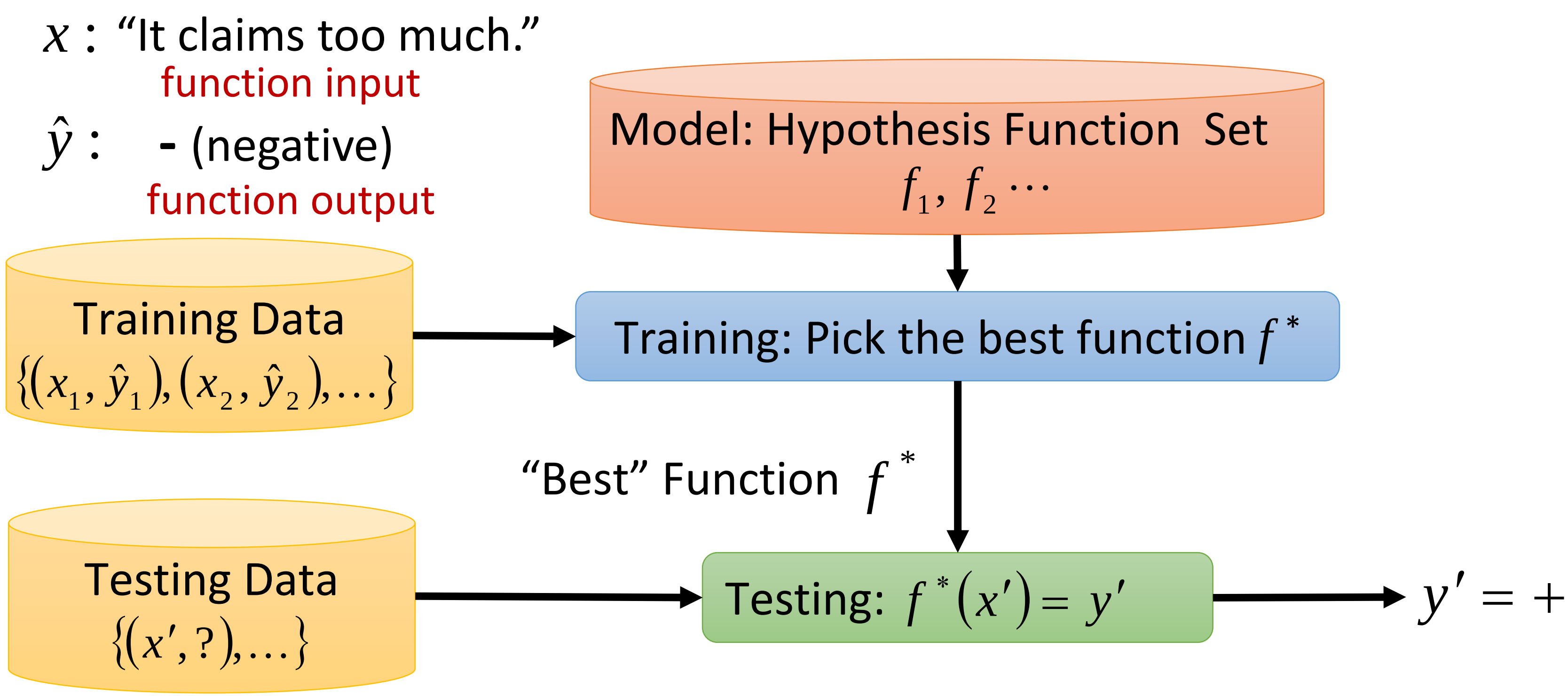
Weather forecast

$f($ Thursday $) =$ “  Saturday”

Play video games

$f($  $) =$ “move left”

Machine Learning Framework



Training is to pick the best function given the observed data

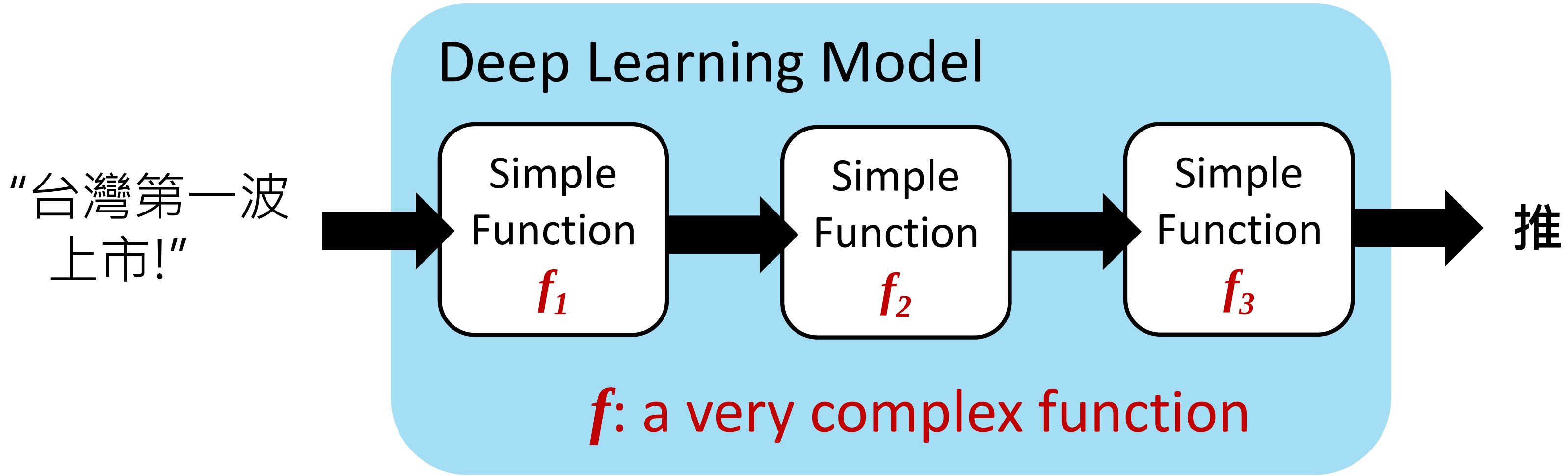
Testing is to predict the label using the learned function

What is Deep Learning?

A subfield of machine learning

Stacked Functions Learned by Machine

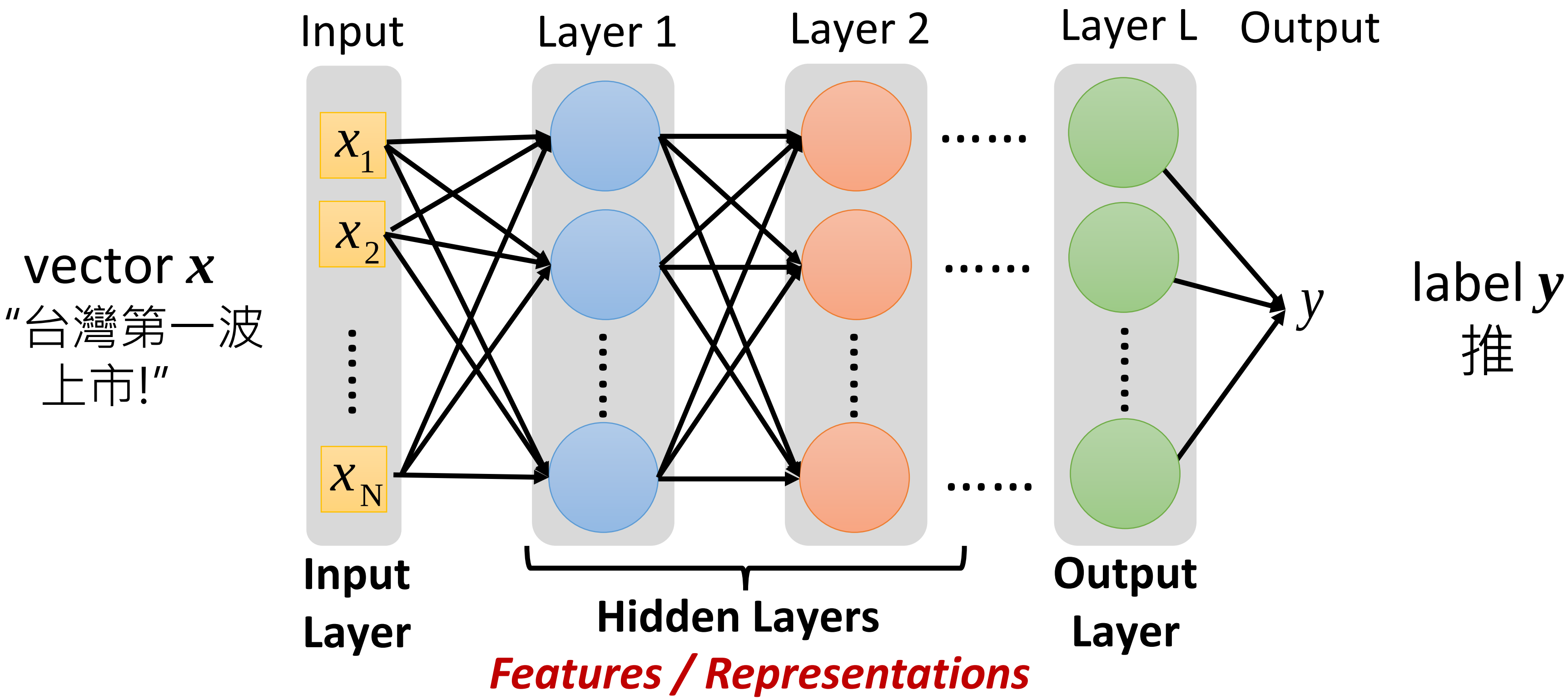
Production line (生產線)



End-to-end training: what each function should do is learned automatically

Deep learning usually refers to *neural network* based model

Stacked Functions Learned by Machine

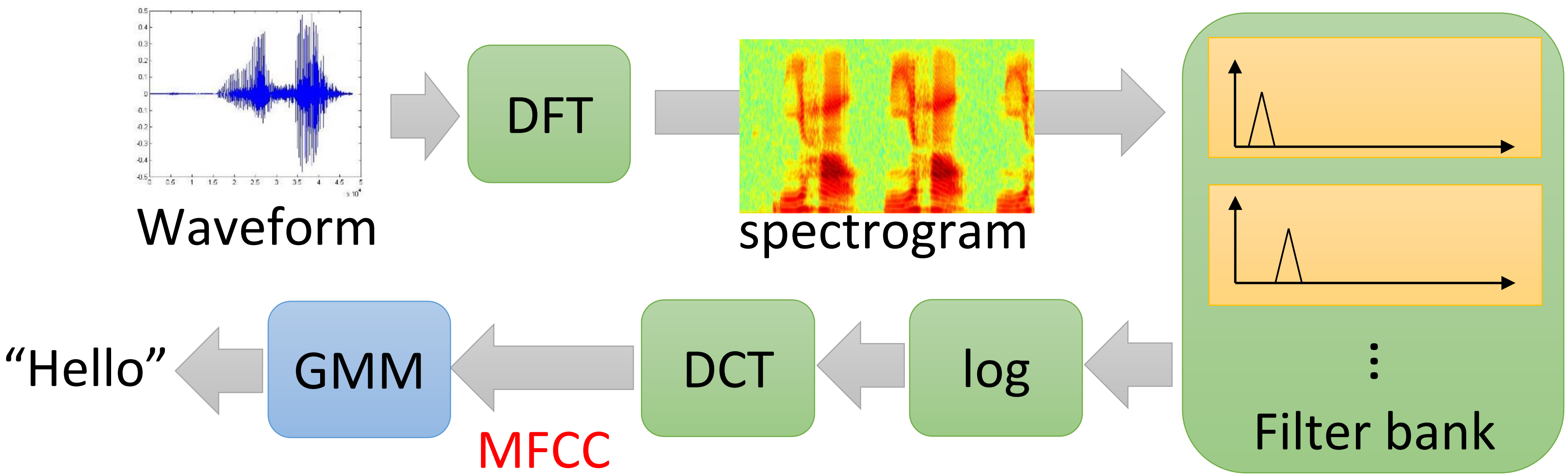


Representation Learning attempts to learn good features/representations

Deep Learning attempts to learn (multiple levels of) representations and an output

Deep v.s. Shallow – Speech Recognition

Shallow Model

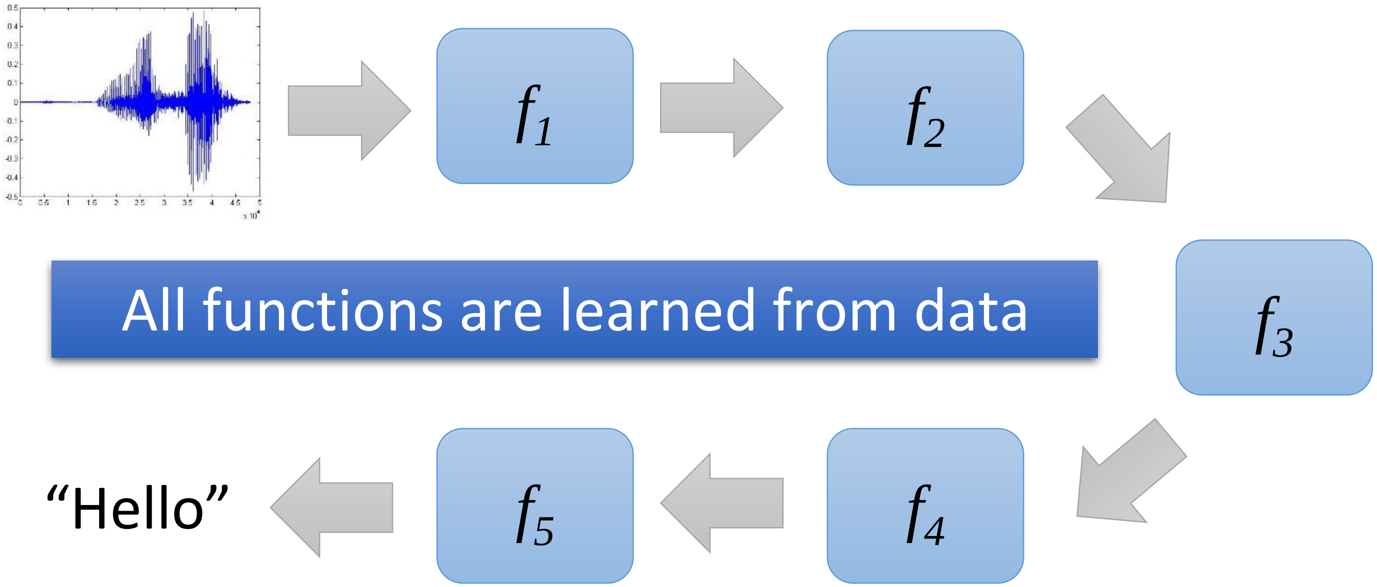


Each box is a simple function in the production line:

:hand-crafted :learned from data

Deep v.s. Shallow – Speech Recognition

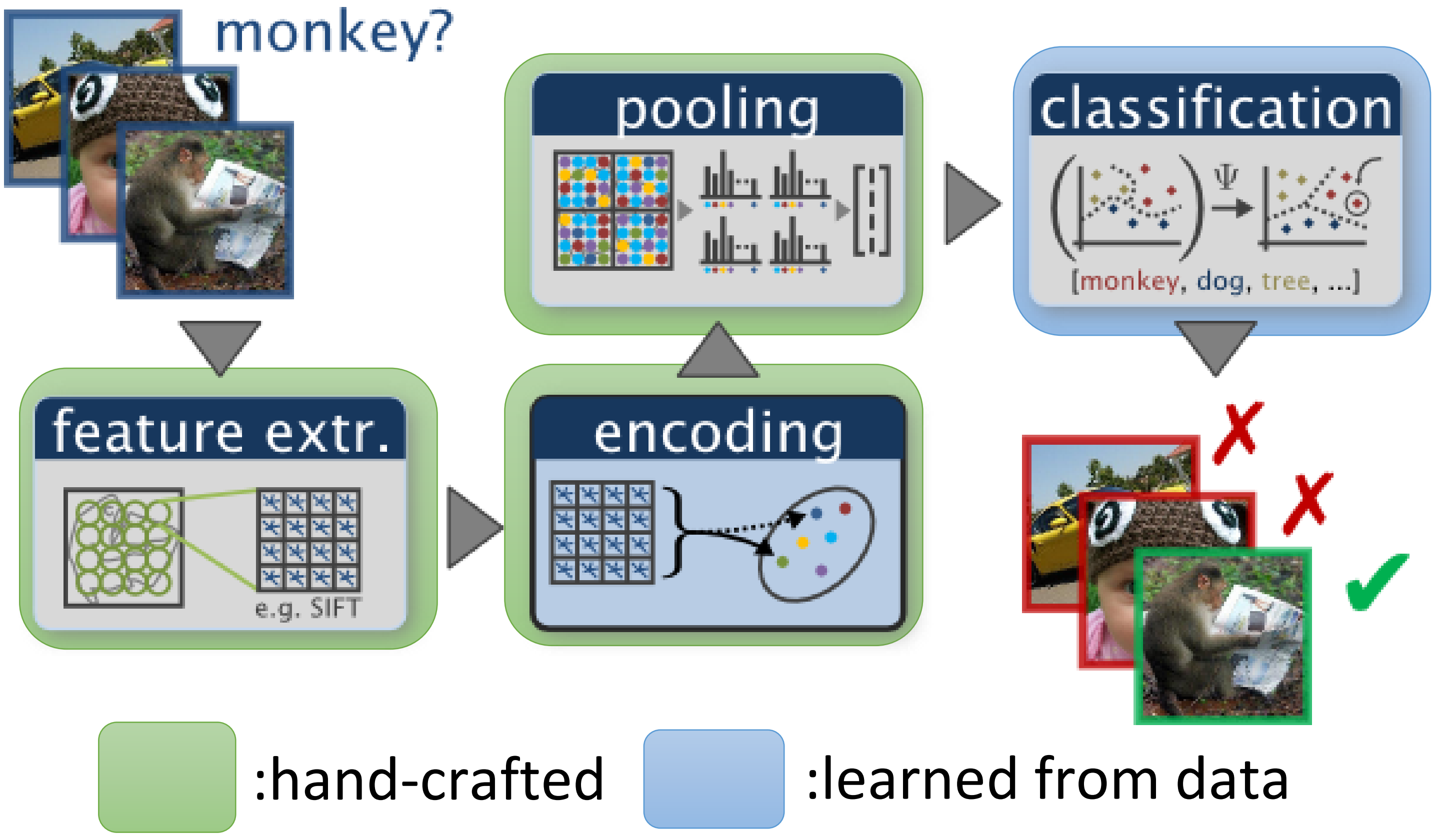
Deep Model



Less engineering labor, but machine learns more

Deep v.s. Shallow – Image Recognition

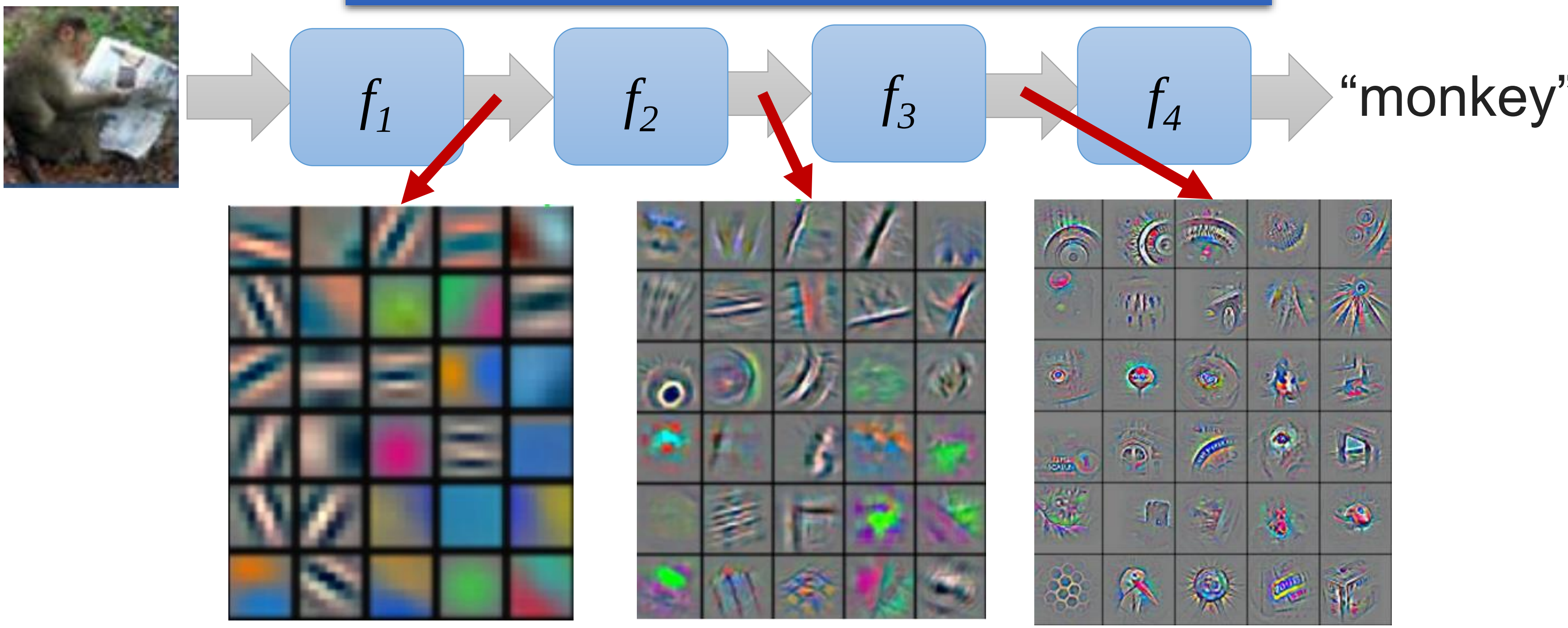
Shallow Model



Deep v.s. Shallow – Image Recognition

Deep Model

All functions are learned from data



Features / Representations

Machine Learning v.s. Deep Learning

Machine Learning

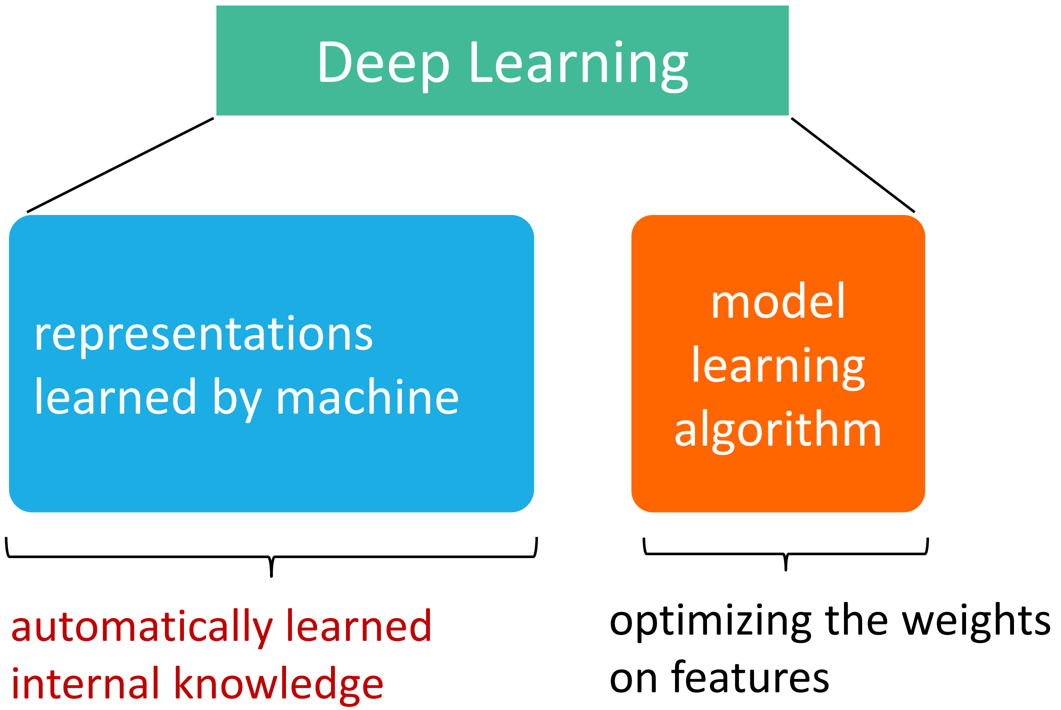
describing your data with features a computer can understand

model learning algorithm

hand-crafted domain-specific knowledge

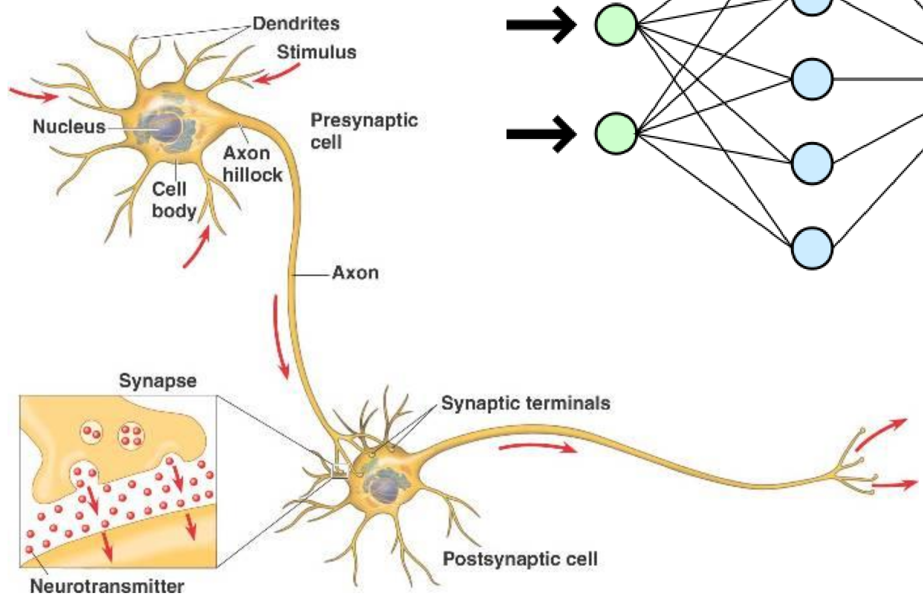
optimizing the weights on features

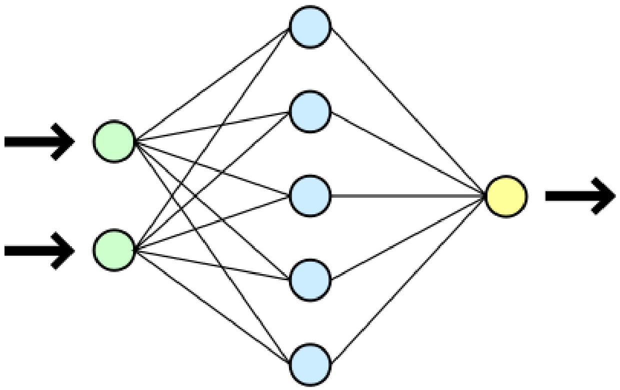
Machine Learning v.s. Deep Learning



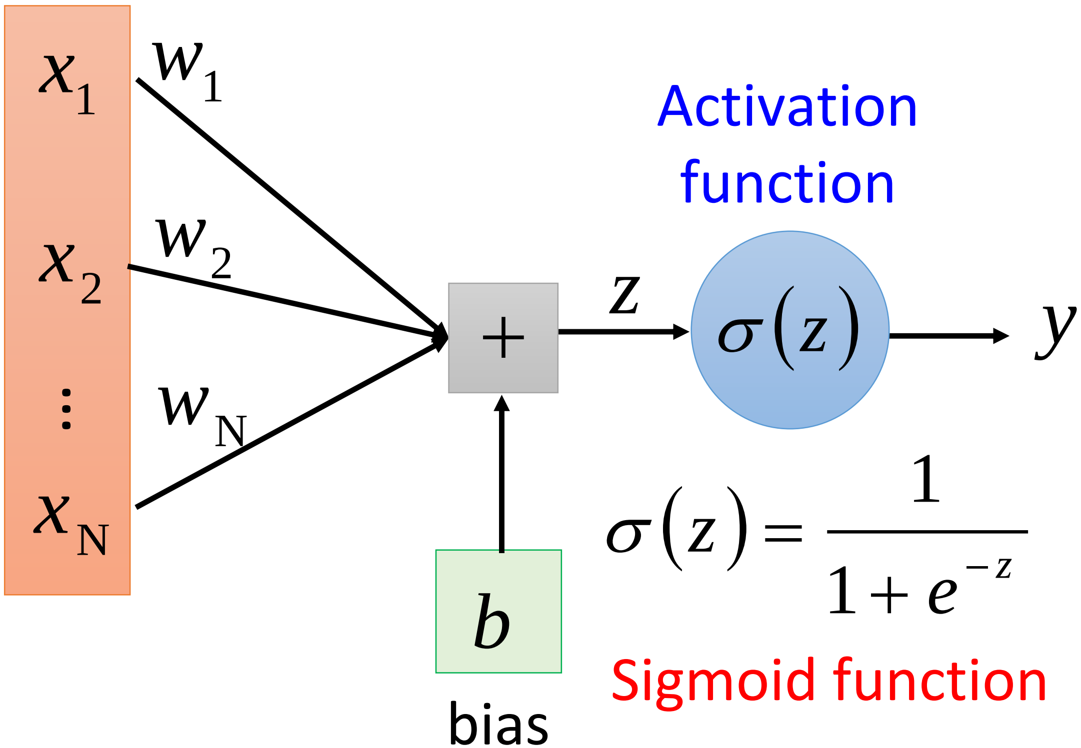
Deep learning usually refers to *neural network* based model

Inspired by Human Brain



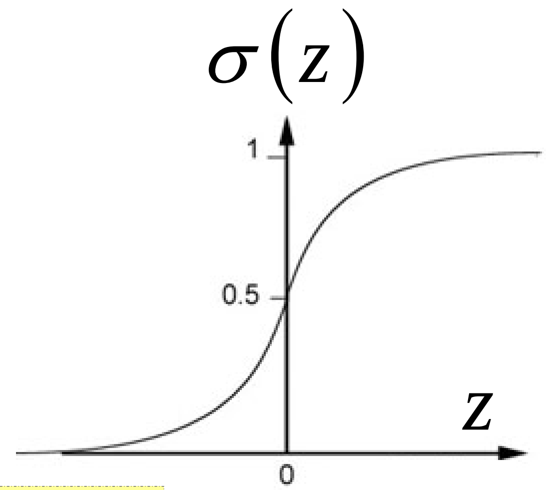


A Single Neuron



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

Sigmoid function



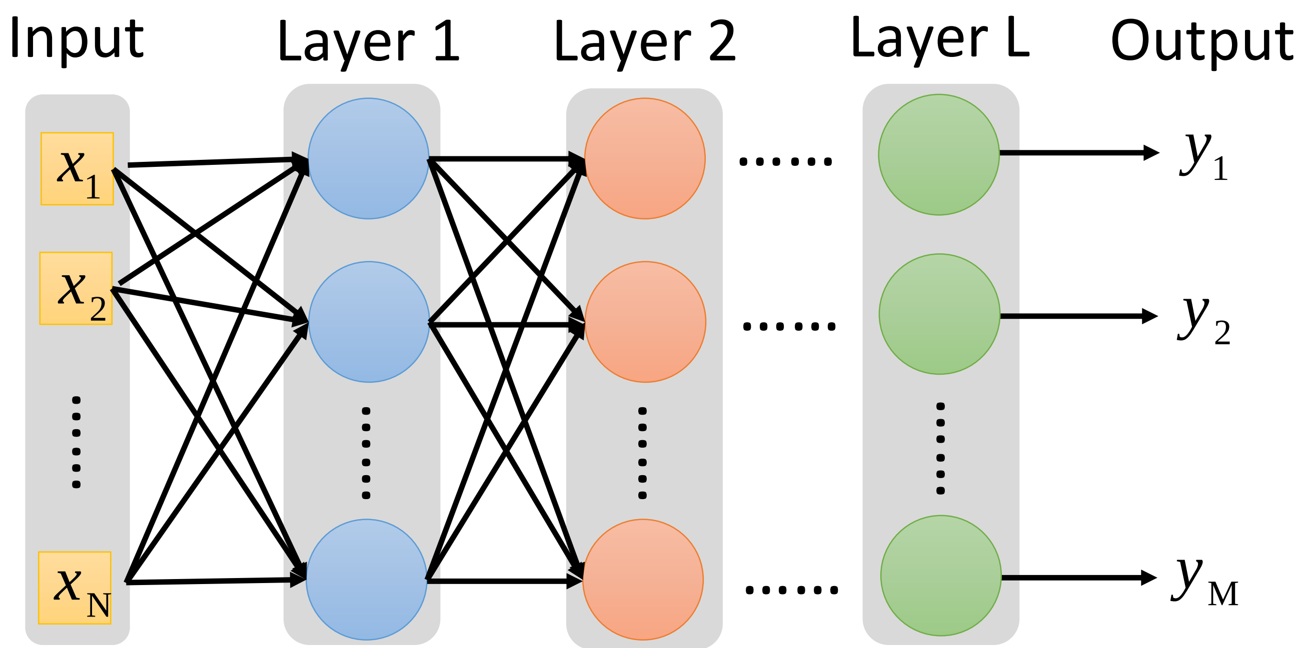
Each neuron is a very simple function

Deep Neural Network

A neural network is a complex function:

$$f : R^N \to R^M$$

Cascading the neurons to form a neural network



Each layer is a simple function in the production line

History of Deep Learning

1960s: Perceptron (single layer neural network)

1969: Perceptron has limitation

1980s: Multi-layer perceptron

1986: Backpropagation

1989: 1 hidden layer is "good enough", why deep?

2006: RBM initialization (breakthrough)

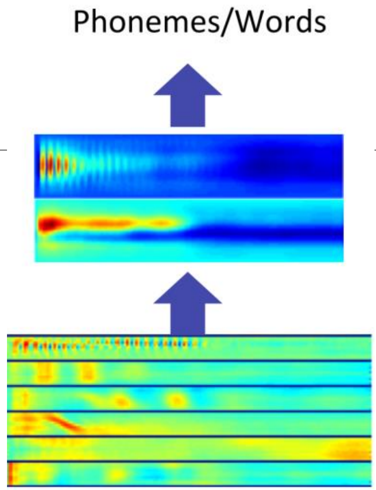
2009: GPU

2010: breakthrough in Speech Recognition (Dahl et al., 2010)

2012: breakthrough in ImageNet (Krizhevsky et al. 2012)

2015: "superhuman" results in Image and Speech Recognition

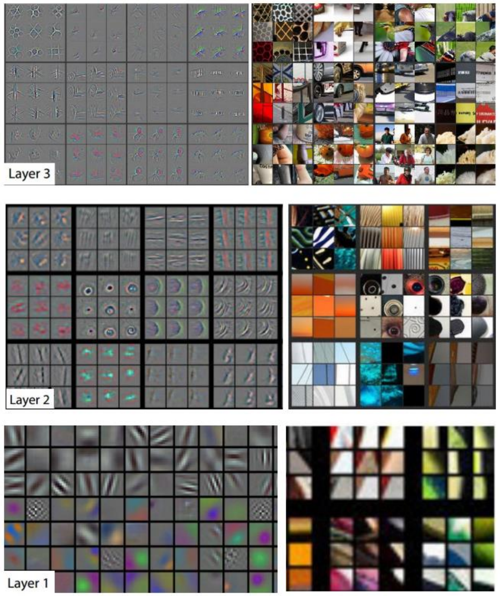
Deep Learning Breakthrough



First: Speech Recognition

Acoustic Model	WER on RT03S FSH	WER on Hub5 SWB
Traditional Features	27.4%	23.6%
Deep Learning	18.5% (-33%)	16.1% (-32%)

Second: Computer Vision



History of Deep Learning

1960s: Perceptron (single layer neural network)

1969: Perceptron has limitation

1980s: Multi-layer perceptron

1986: Backpropagation

1989: 1 hidden layer is “good enough”, why deep?

2006: RBM initialization (breakthrough)

2009: GPU

2010: breakthrough in Speech Recognition (Dahl et al., 2010)

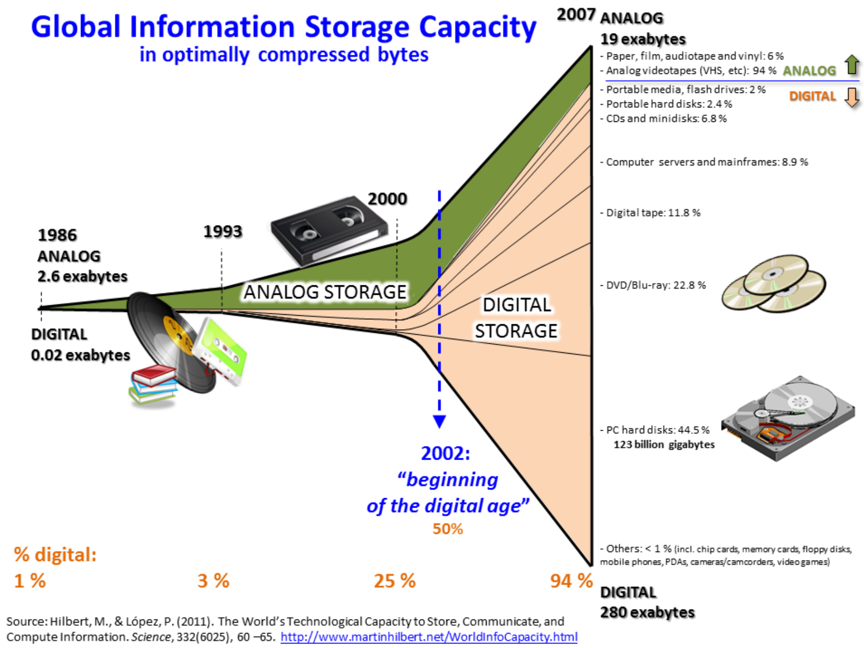
2012: breakthrough in ImageNet (Krizhevsky et al. 2012)

2015: “superhuman” results in Image and Speech Recognition

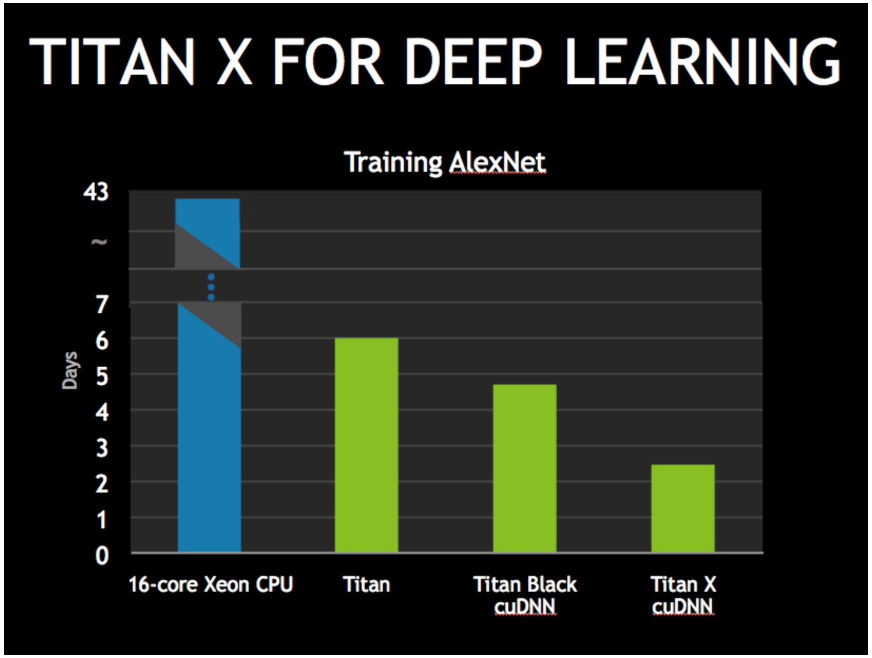
Why does deep learning show breakthrough in applications after 2010?

Reasons why Deep Learning works

Big Data



GPU



Why to Adopt GPU for Deep Learning?

GPU is like a brain

Human brains create *graphical imagination* for *mental thinking*

台灣好吃牛肉麵

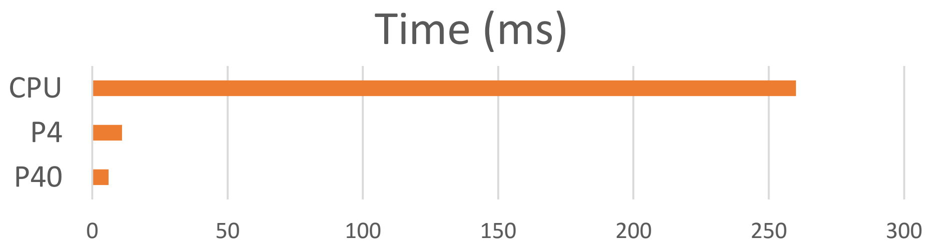
Why Speed Matters?

Training time

- Big data increases the training time
- Too long training time is not practical

Inference time

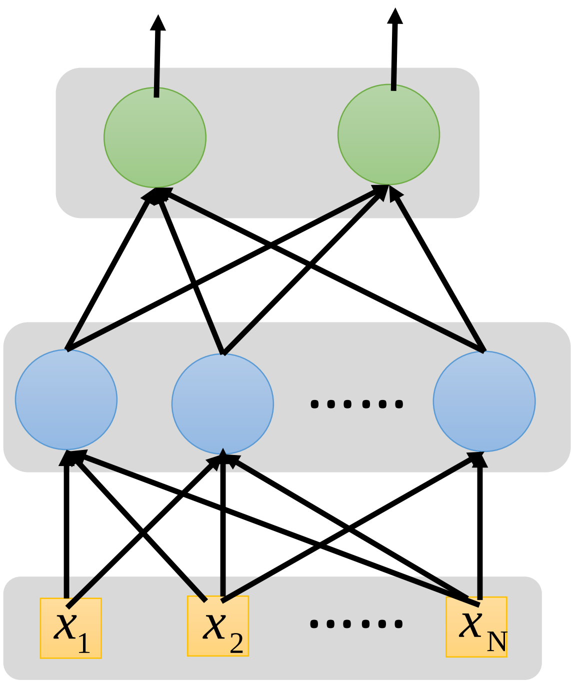
- Users are not patient to wait for the responses



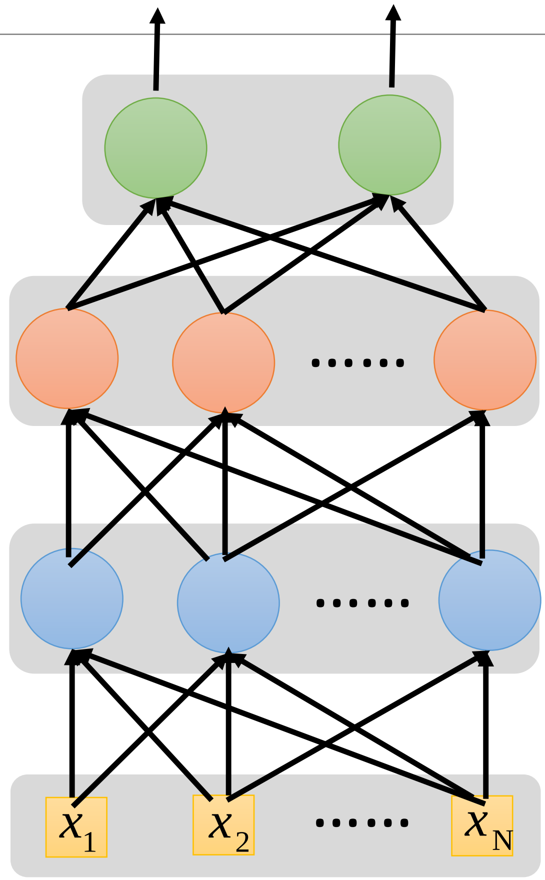
GPU enables the real-world applications using the computational power

Why Deeper is Better?

Deeper → More parameters



Shallow



Deep

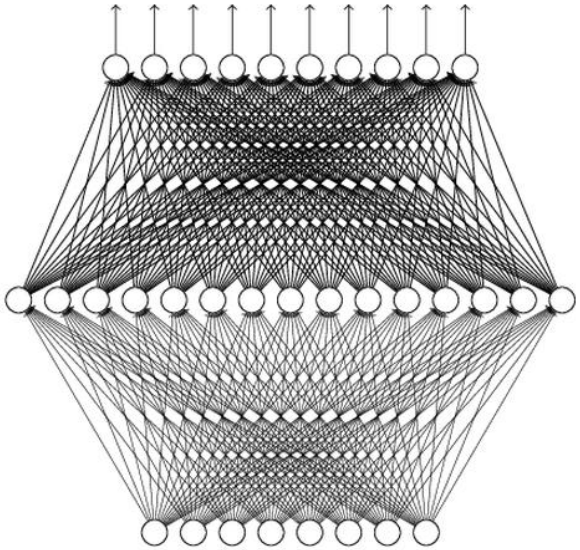
Universality Theorem

Any continuous function f

$$f : R^N \rightarrow R^\mathrm{M}$$

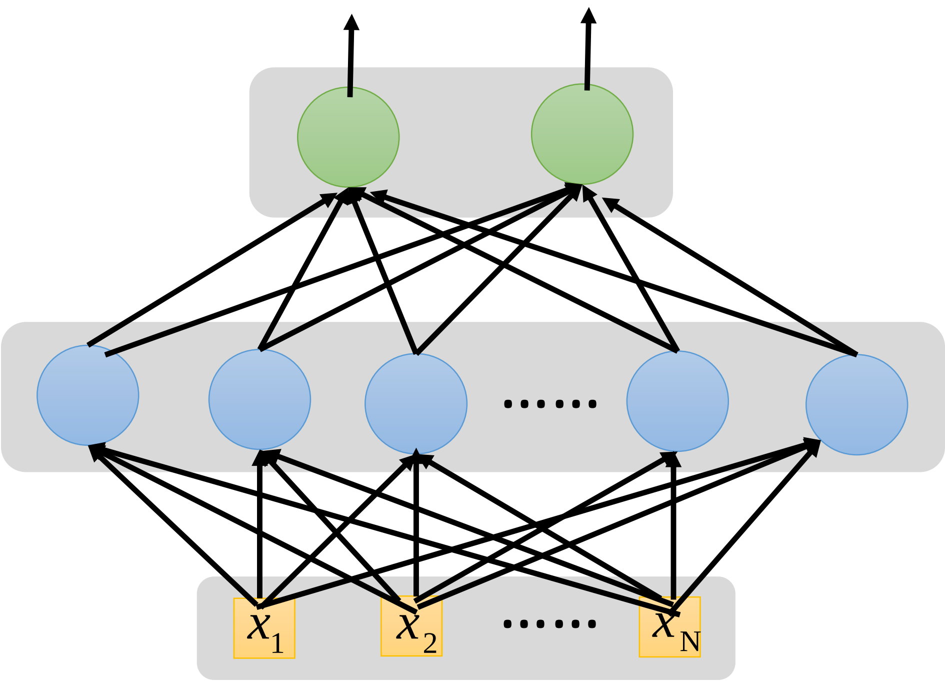
can be realized by a network with only hidden layer

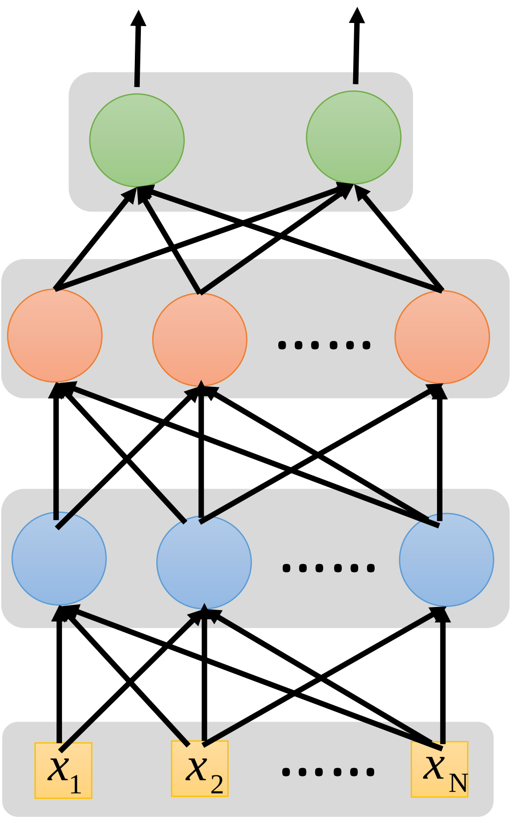
Why "deep" not "fat"?



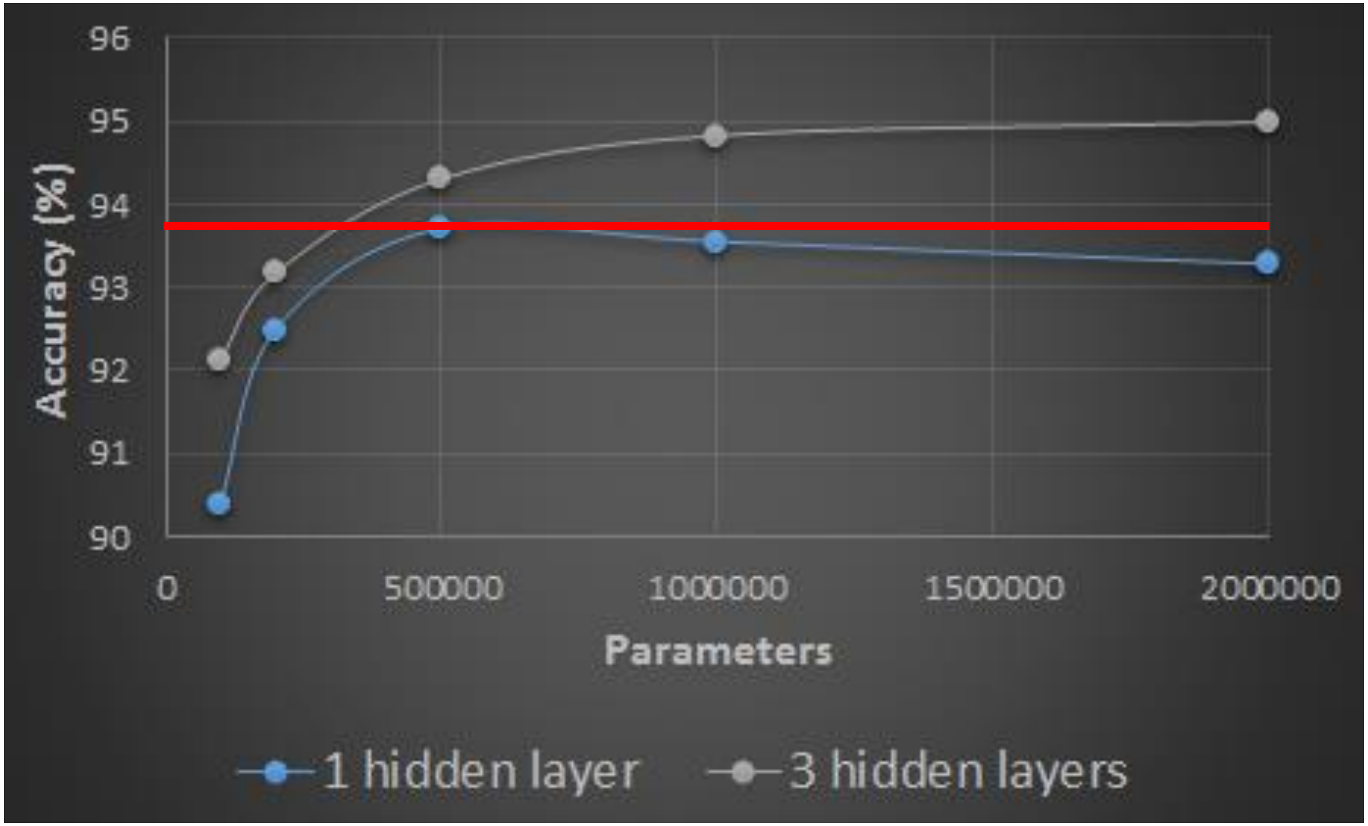
Fat + Shallow v.s. Thin + Deep

Two networks with the same number of parameters





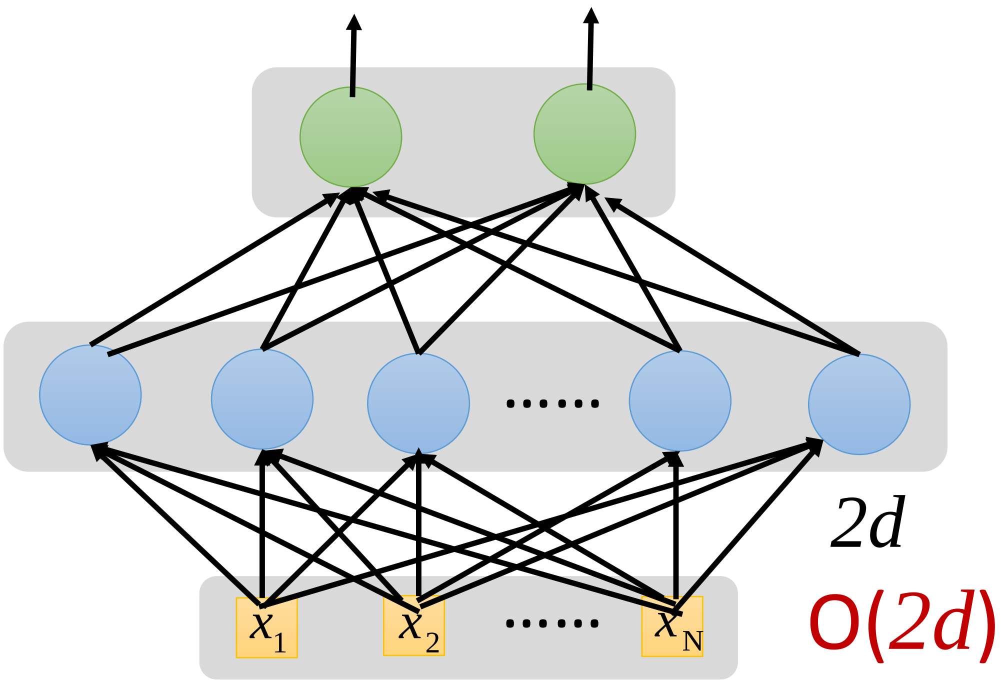
Fat + Shallow v.s. Thin + Deep
Hand-Written Digit Classification

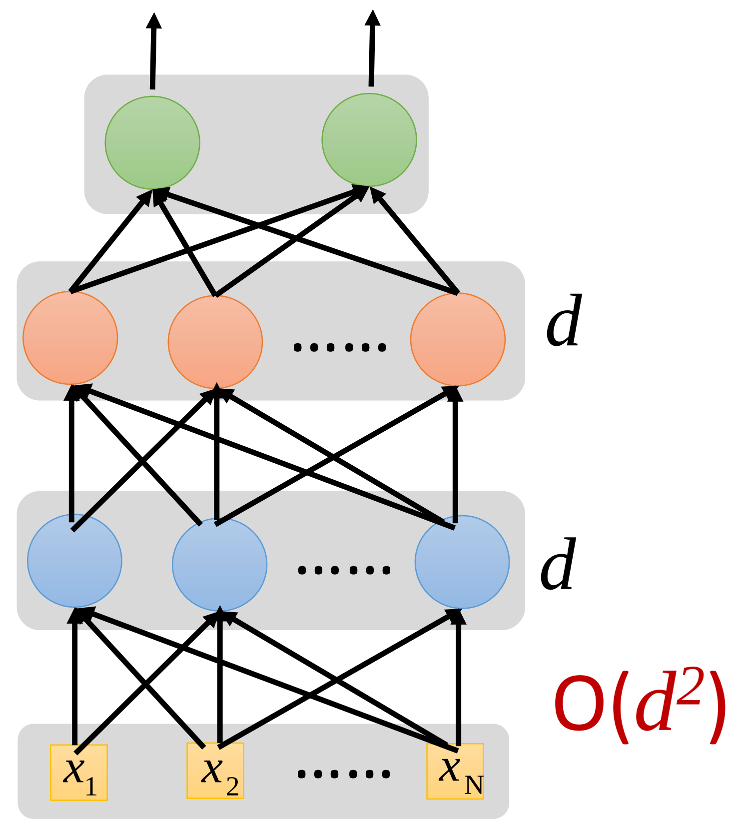


The deeper model uses less parameters to achieve the same performance

Fat + Shallow v.s. Thin + Deep

Two networks with the same number of parameters





How to Apply?

How to Frame the Learning Problem?

The learning algorithm f is to map the input domain X into the output domain Y

$$f : X \rightarrow Y$$

Input domain: word, word sequence, audio signal, click logs

Output domain: single label, sequence tags, tree structure, probability distribution

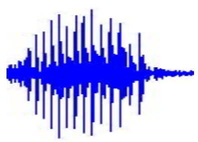
Output Domain – Classification

Sentiment Analysis

“這規格有誠意!” → +

“太爛了吧~” → -

Speech Phoneme Recognition

 → /h/

Handwritten Recognition

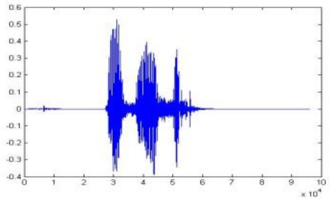
 → 2

Output Domain – Sequence Prediction

POS Tagging

"推薦我台大後門的餐廳" ⟶ 推薦/VV 我/PN 台大/NR 後門/NN 的/DEG 餐廳/NN

Speech Recognition



⟶ "大家好"

Machine Translation

"How are you doing today?" ⟶ "你好嗎?"

Learning tasks are decided by the output domains

Input Domain – How to Aggregate Information

Input: word sequence, image pixels, audio signal, click logs

Property: continuity, temporal, importance distribution

Example

- CNN (convolutional neural network): local connections, shared weights, pooling
  - AlexNet, VGGNet, etc.
- RNN (recurrent neural network): temporal information

Network architectures should consider the input domain properties

How to Frame the Learning Problem?

The learning algorithm f is to map the input domain X into the output domain Y

$$f : X \rightarrow Y$$

Input domain: word, word sequence, audio signal, click logs

Output domain: single label, sequence tags, tree structure, probability distribution

Network design should leverage input and output domain properties

"Applied" Deep Learning

Deep Learning

representations learned by machine

model learning algorithm

automatically learned internal knowledge

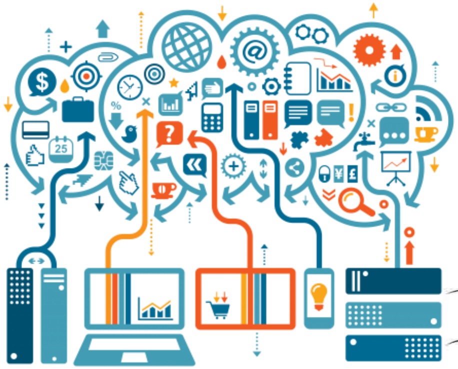
optimizing the weights on features

How to frame a task into a learning problem and design the corresponding model

Core Factors for Applied Deep Learning

1. Data: big data
2. Hardware: GPU computing
3. Talent: design algorithms to allow networks to work for the specific problems

Concluding Remarks



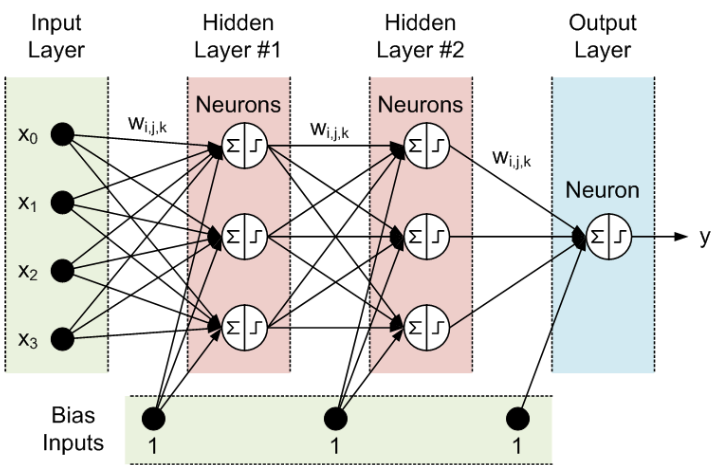
Training

Concluding Remarks

Inference

Concluding Remarks

Inference



Training

Main focus: how to apply deep learning to the real-world problems

Reference

Reading Materials

- Academic papers will be put in the website

Deep Learning

- Goodfellow, Bengio, and Courville, “Deep Learning,” 2016. http://www.deeplearningbook.org
- Michael Nielsen, “Neural Networks and Deep Learning” http://neuralnetworksanddeeplearning.com