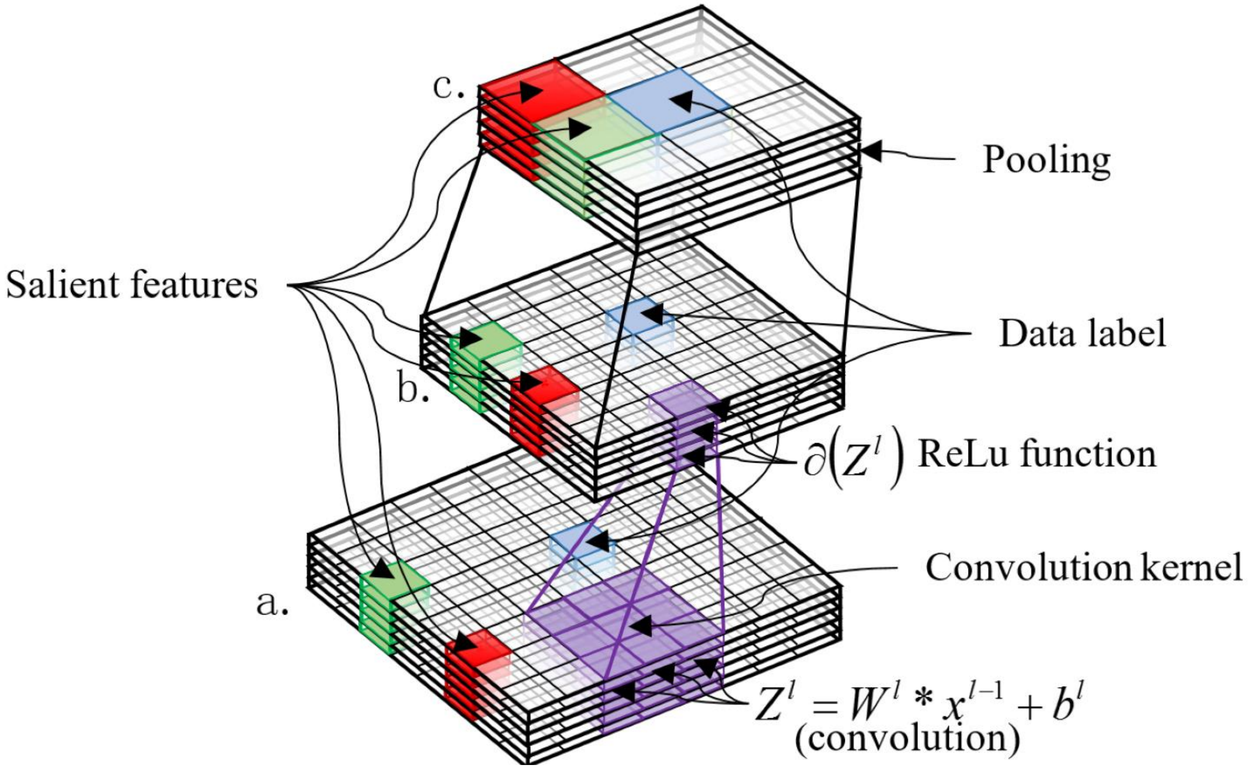
PyTorch Tutorial

01. Overview

Goal of this tutorial

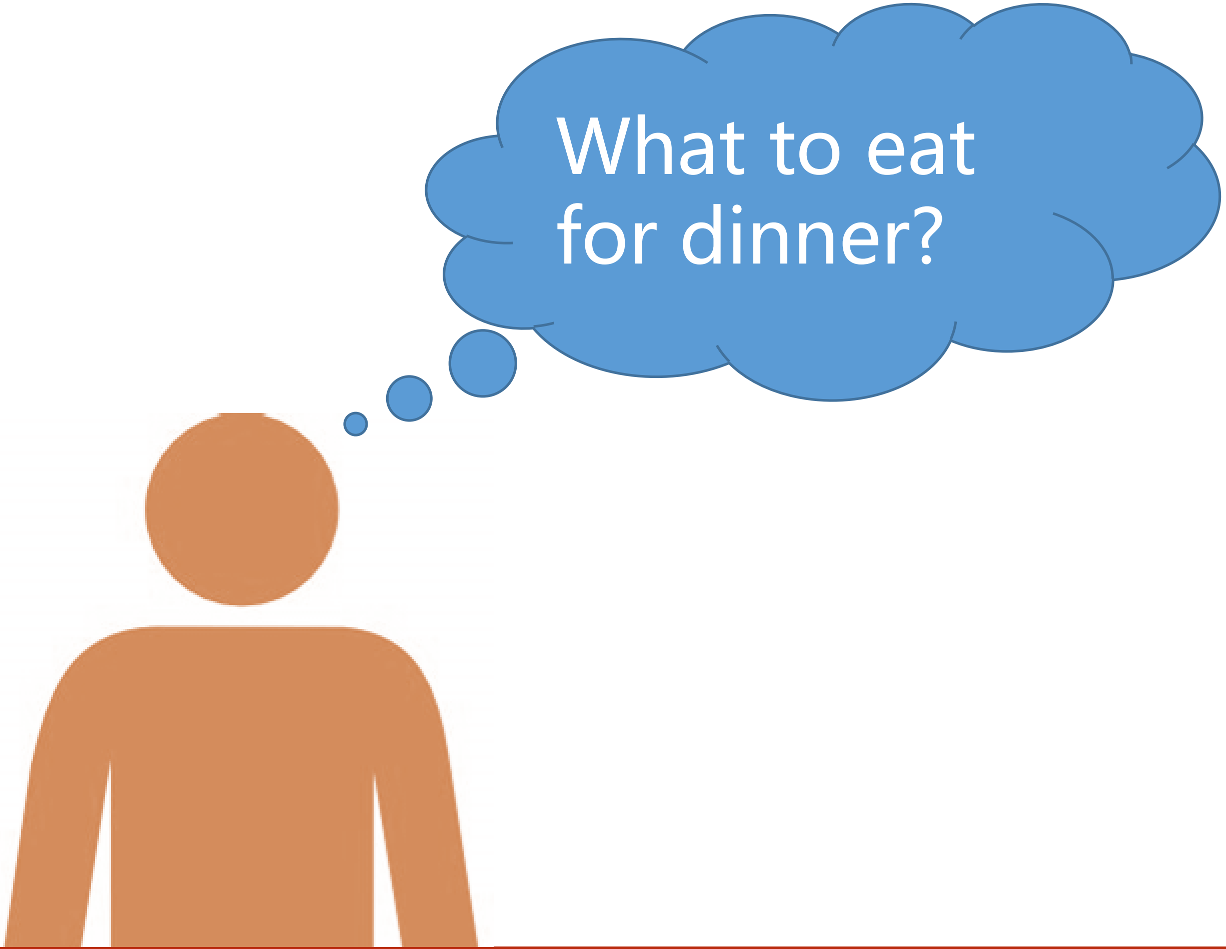
- How to implement learning system using PyTorch
- Understand the basic of neural networks / deep learning

- Requirements
  - Algebra + Probability
  - Python

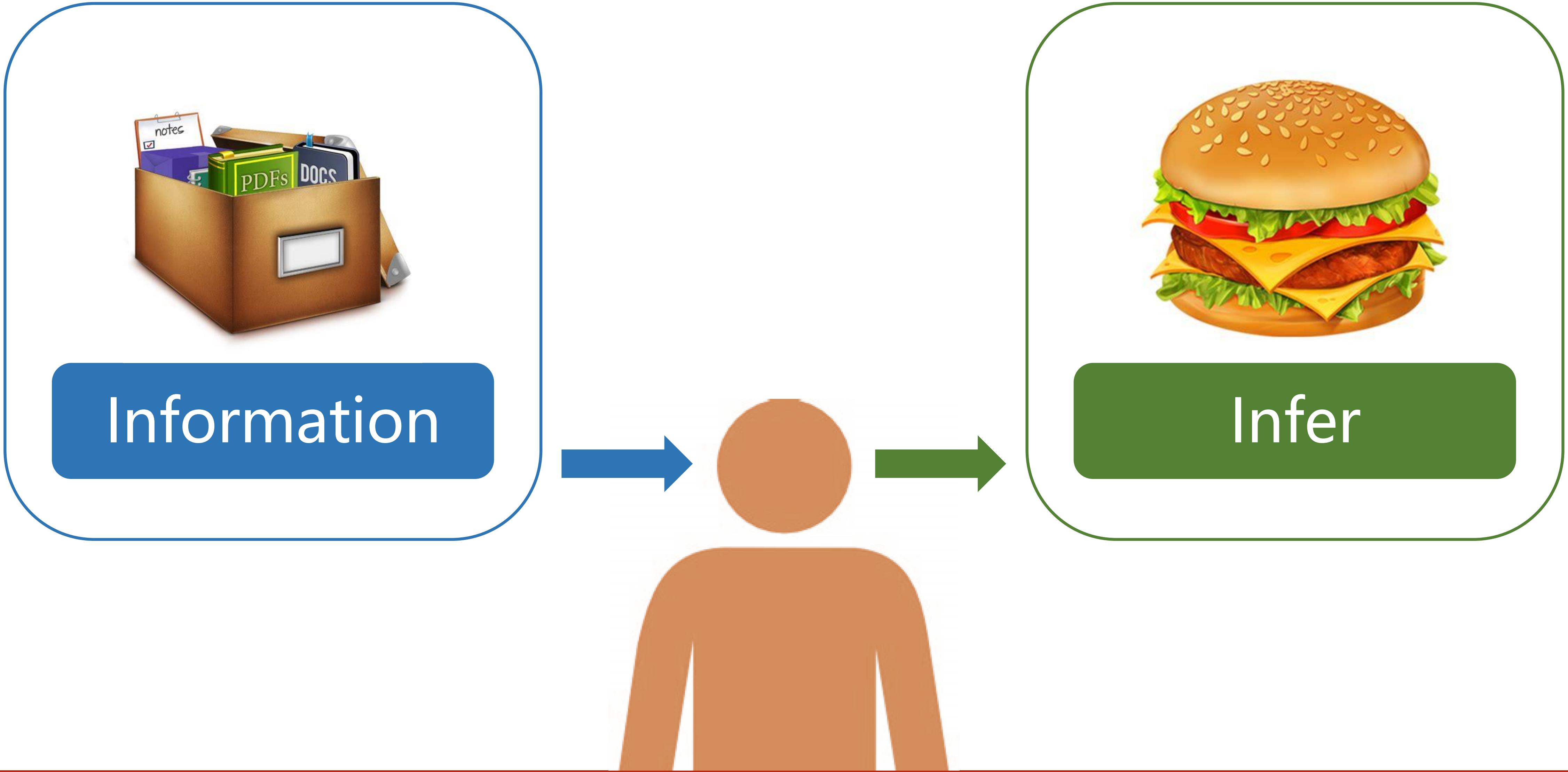


Gao F, Huang T, Wang J, et al. Dual-Branch Deep Convolution Neural Network for Polarimetric SAR Image Classification[J]. Applied Sciences, 2017, 7(5):447.

Human Intelligence

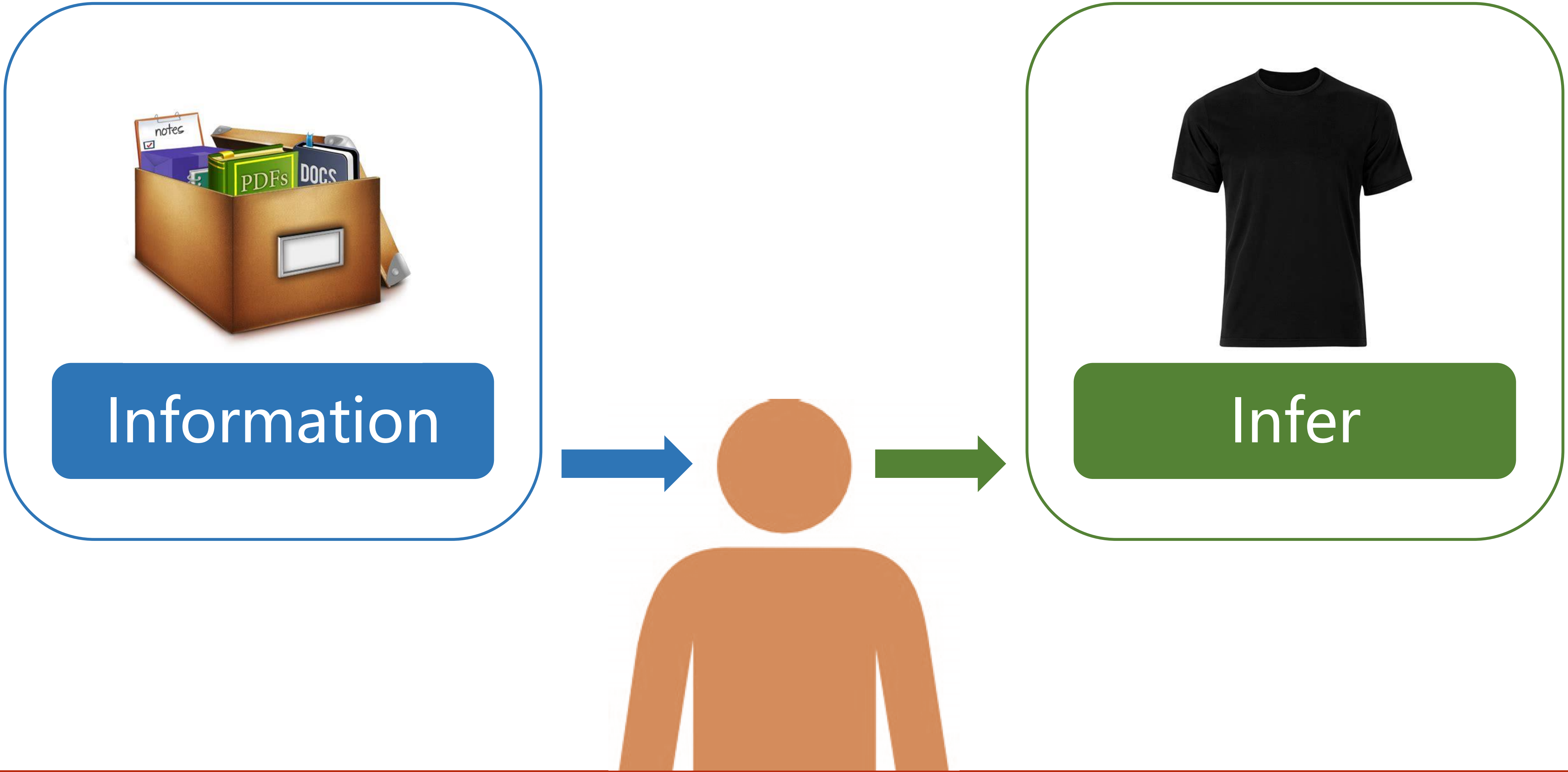


Human Intelligence

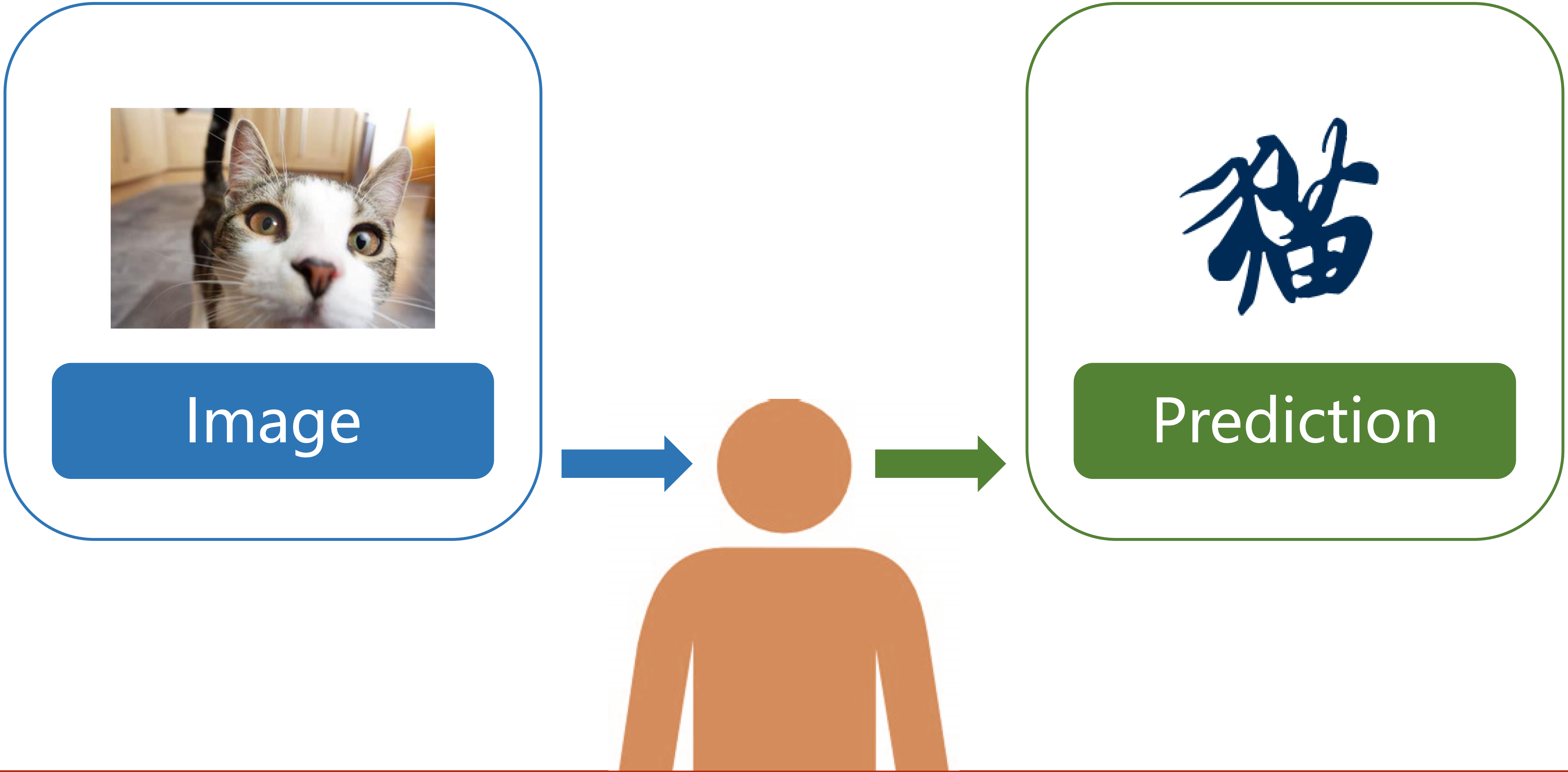


Human Intelligence

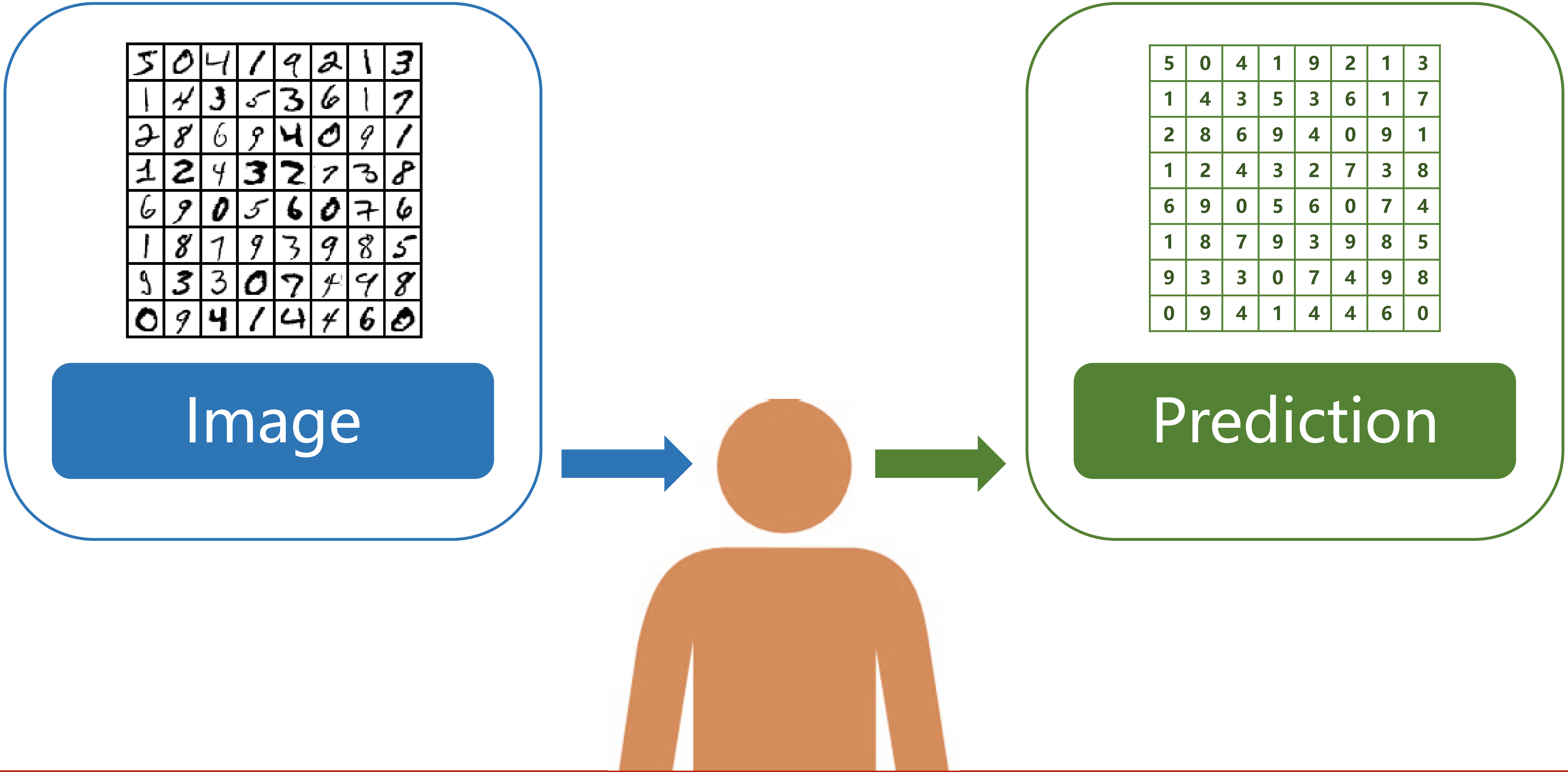
Human Intelligence



Human Intelligence



Human Intelligence

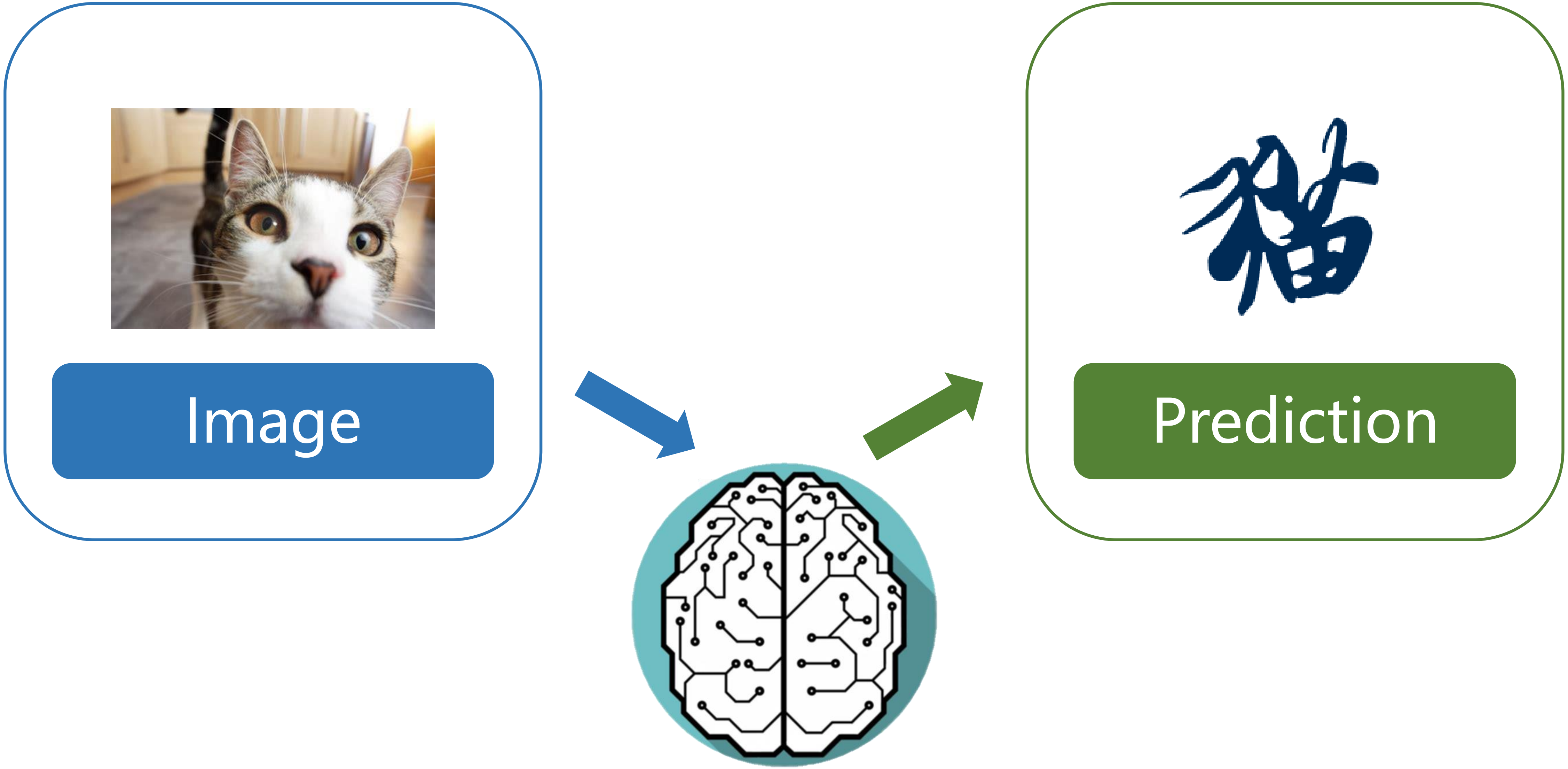


Machine learning

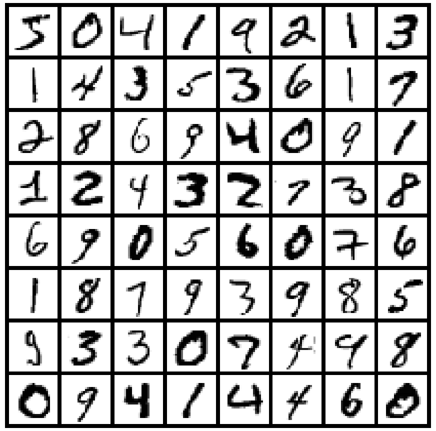
Information

Infer

Machine learning



Machine learning

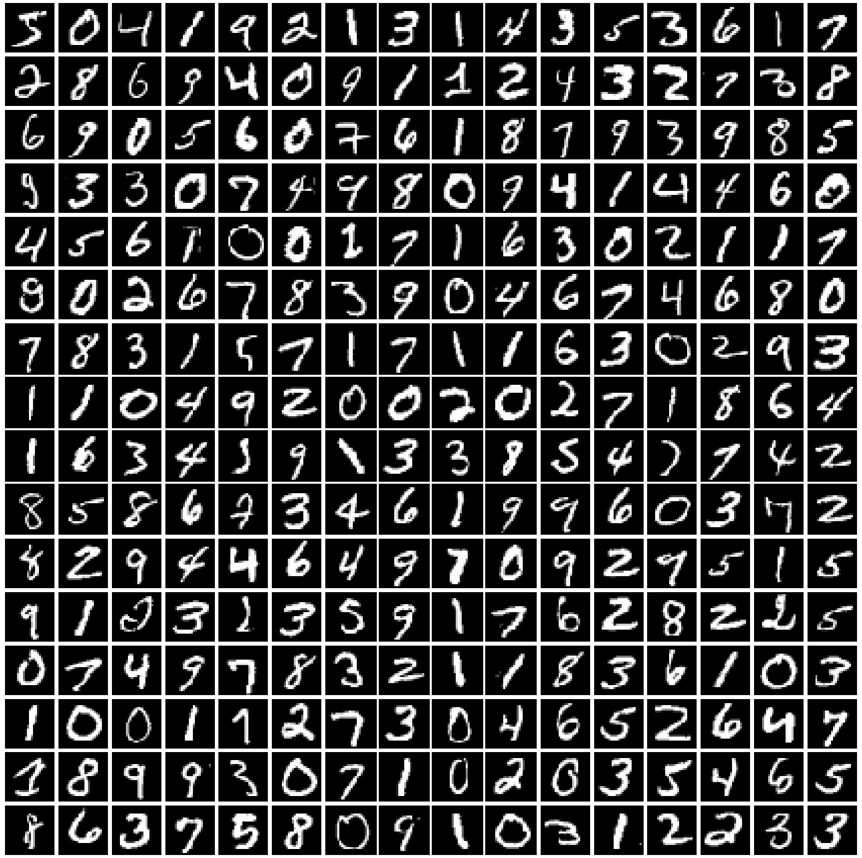


Image

5	0	4	1	9	2	1	3
1	4	3	5	3	6	1	7
2	8	6	9	4	0	9	1
1	2	4	3	2	7	3	8
6	9	0	5	6	0	7	4
1	8	7	9	3	9	8	5
9	3	3	0	7	4	9	8
0	9	4	1	4	4	6	0

Prediction

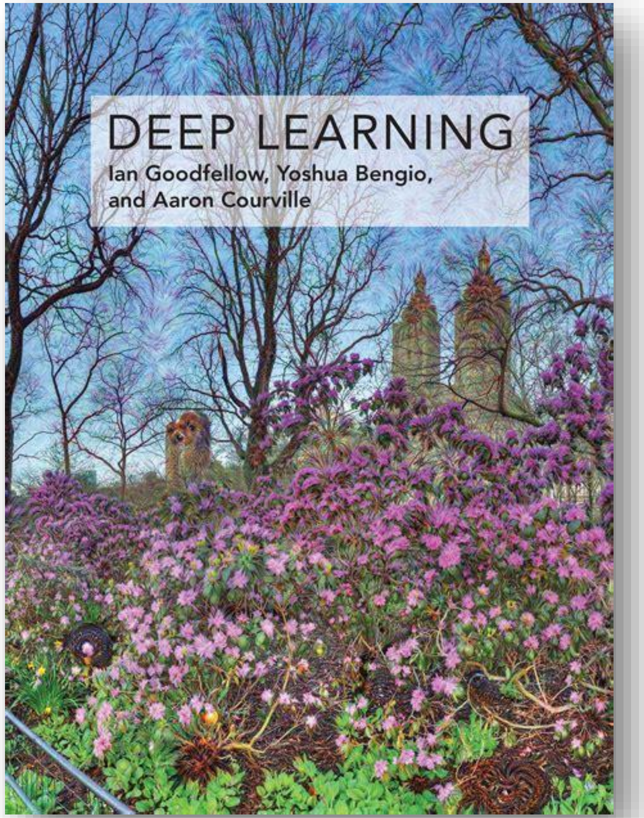
Machine learning



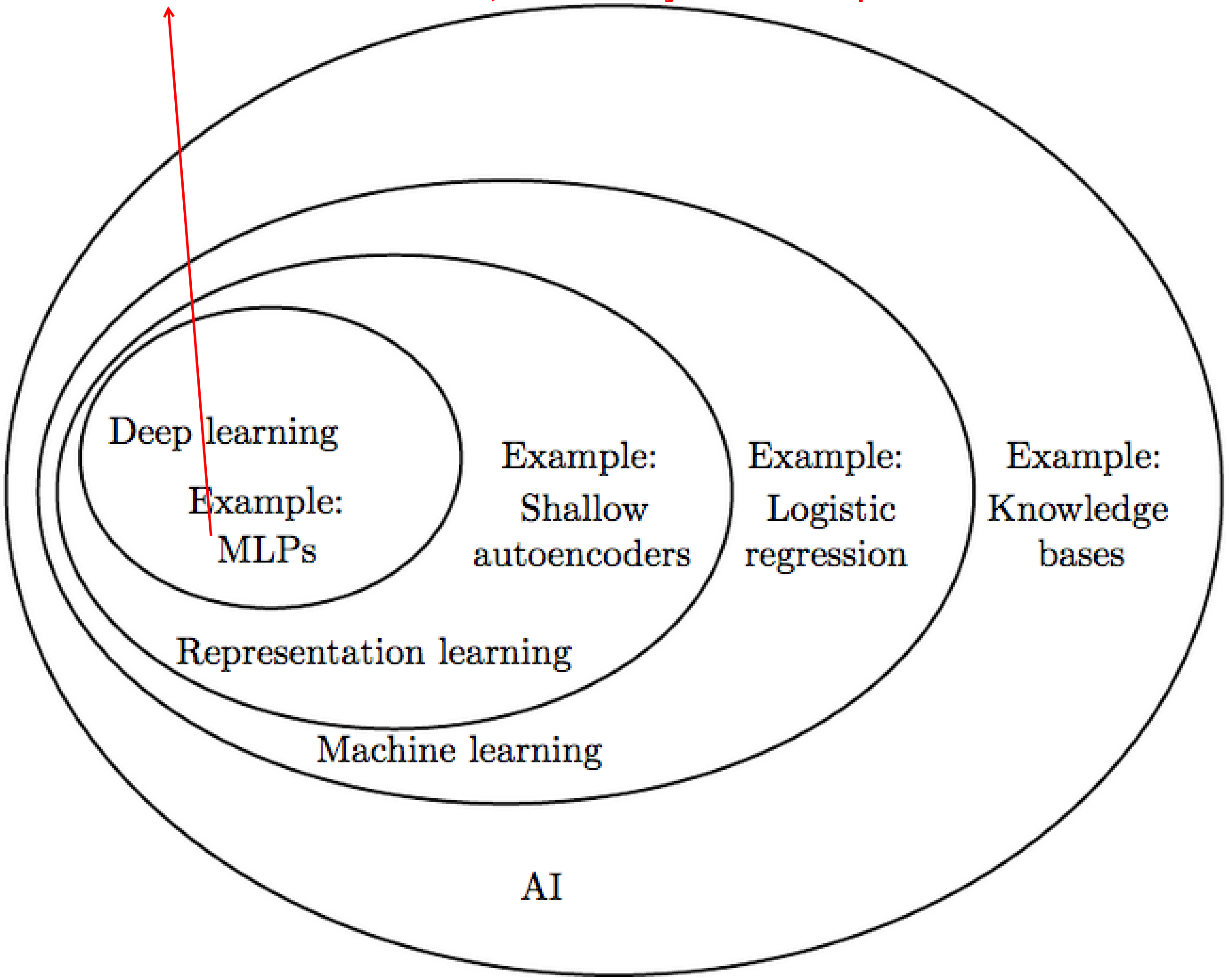
Labeled Dataset

Training

Machine learning



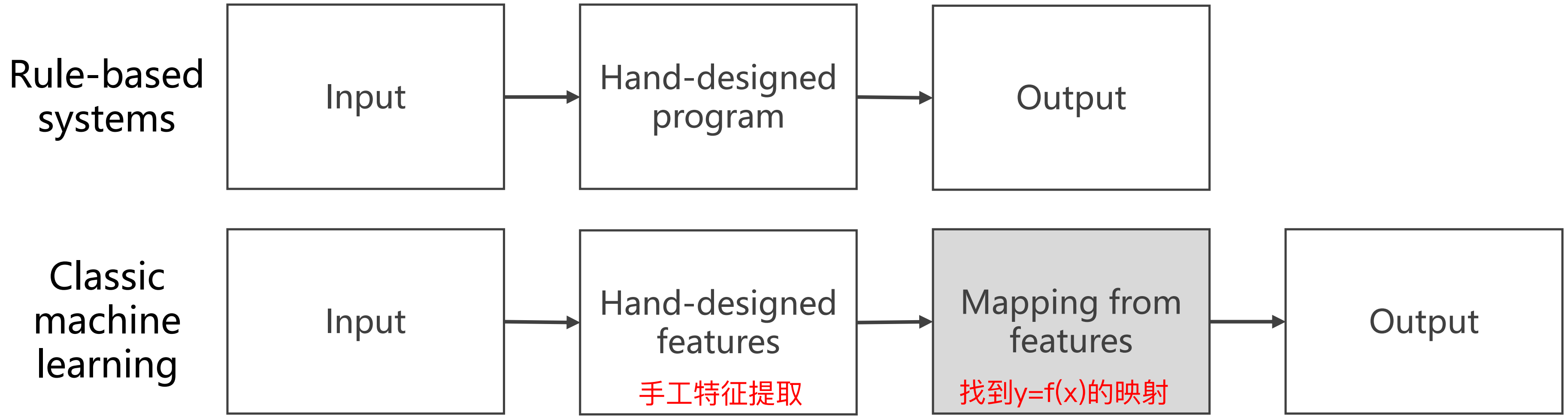
多层感知器（MLP，Multilayer Perceptron）



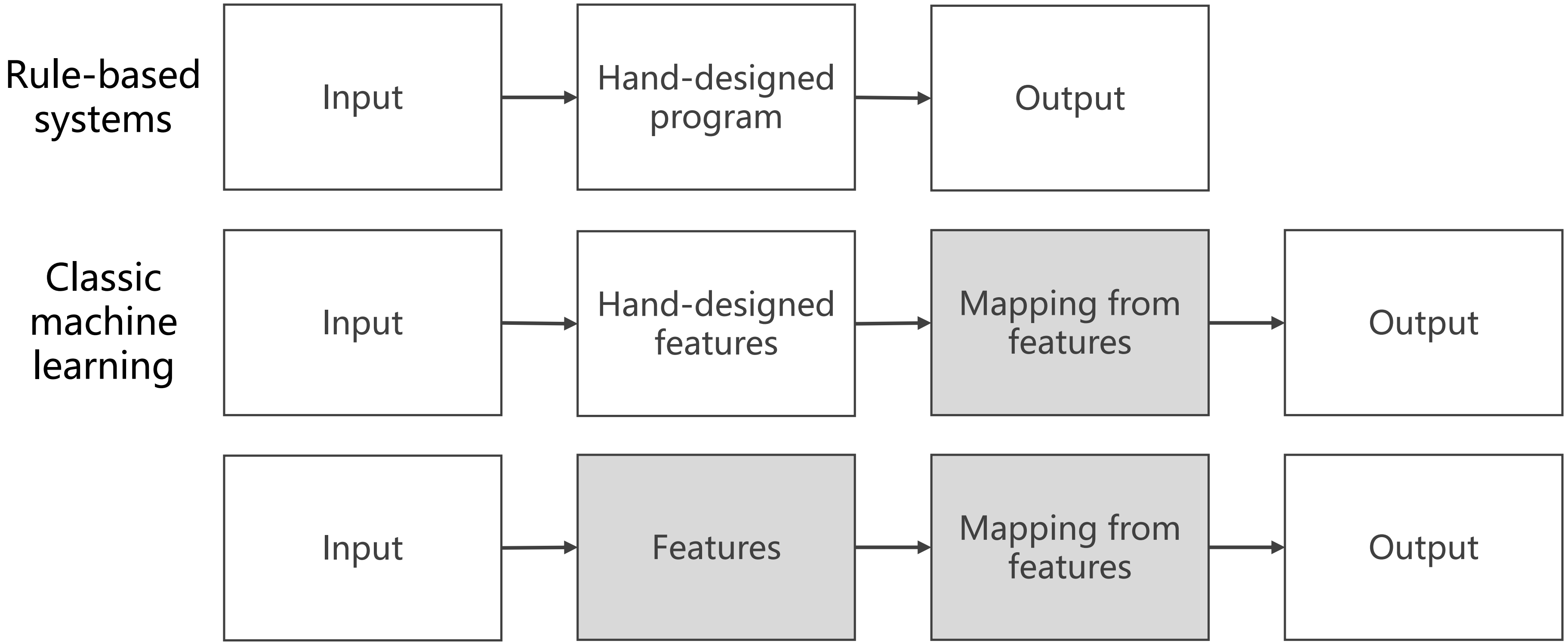
How to develop learning system?

Rule-based systems

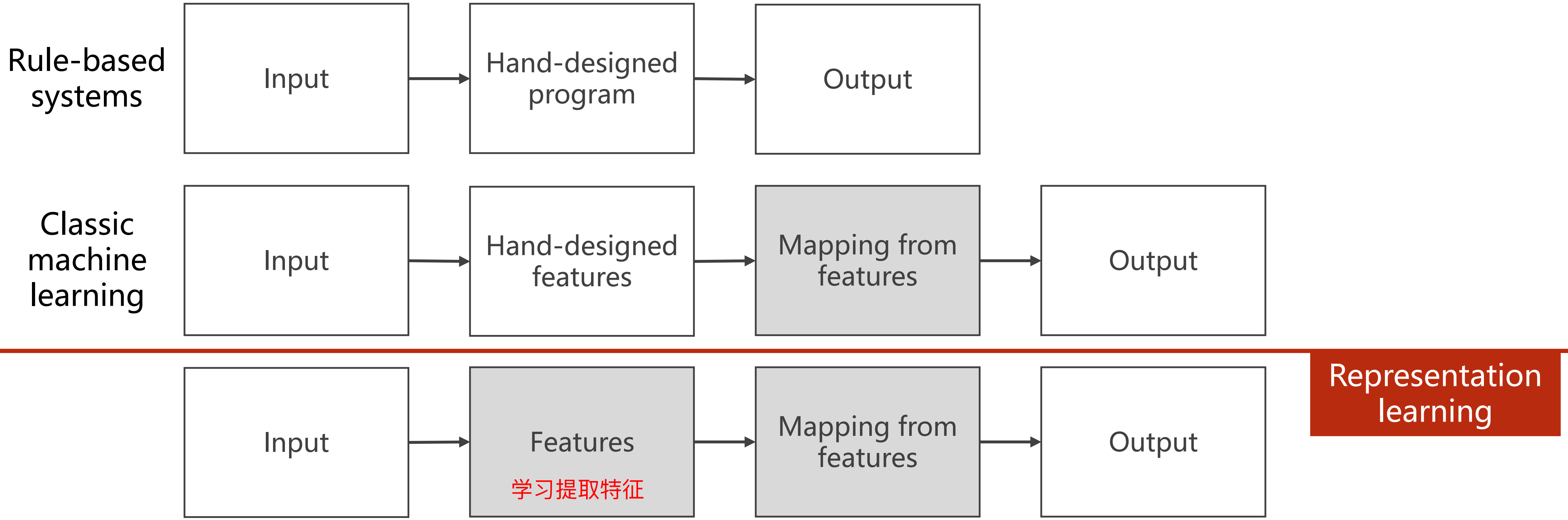
How to develop learning system?



How to develop learning system?



How to develop learning system?



相对于深度学习，传统机器学习的流程往往由多个独立的模块组成，比如在一个典型的自然语言处理（Natural Language Processing）问题中，包括分词、词性标注、句法分析、语义分析等多个独立步骤，每个步骤是一个独立的任务，其结果的好坏会影响到下一步骤，从而影响整个训练的结果，这是非端到端的。

而深度学习模型在训练过程中，从输入端（输入数据）到输出端会得到一个预测结果，与真实结果相比较会得到一个误差，这个误差会在模型中的每一层传递（反向传播），每一层的表示都会根据这个误差来做调整，直到模型收敛或达到预期的效果才结束，这是端到端的。

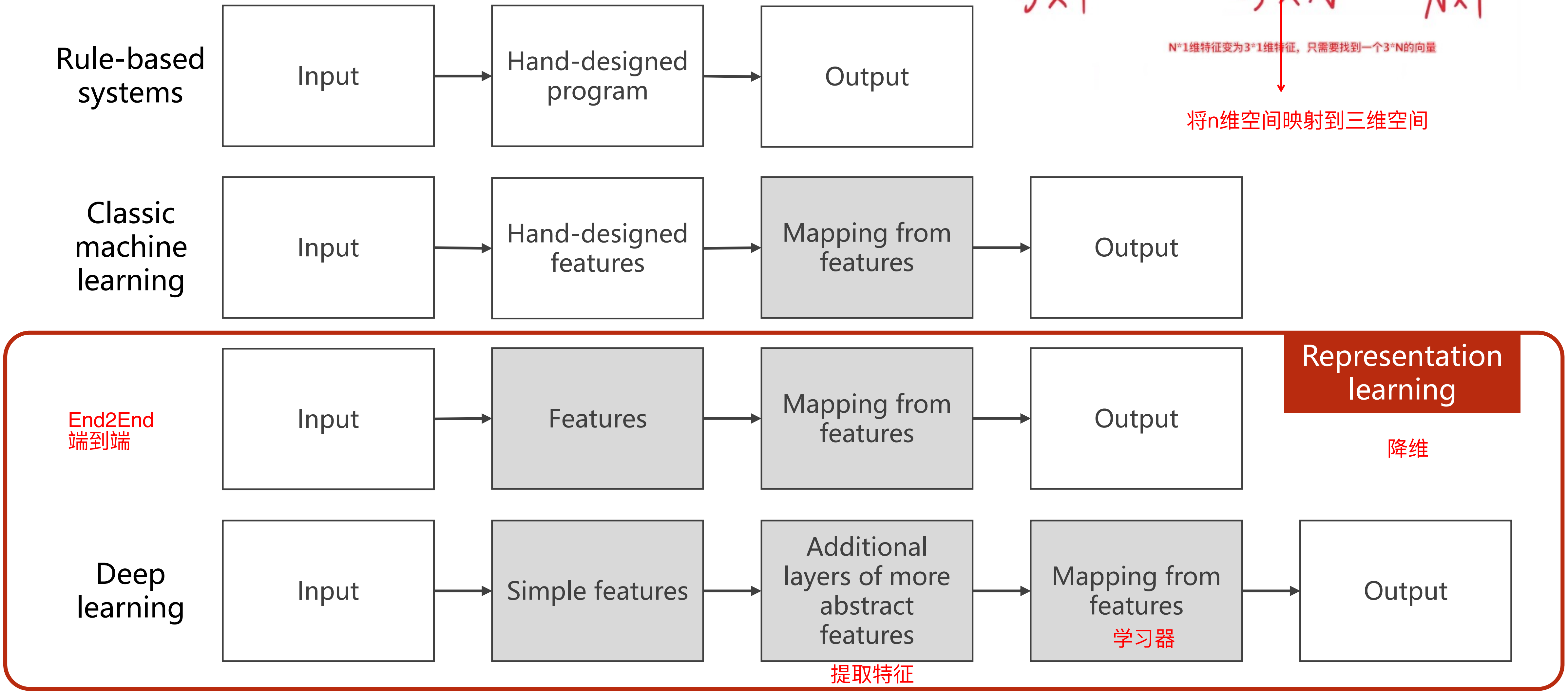
两者相比，端到端的学习省去了在每一个独立学习任务执行之前所做的数据标注，为样本做标注的代价是昂贵的、易出错的。

How to develop learning system?

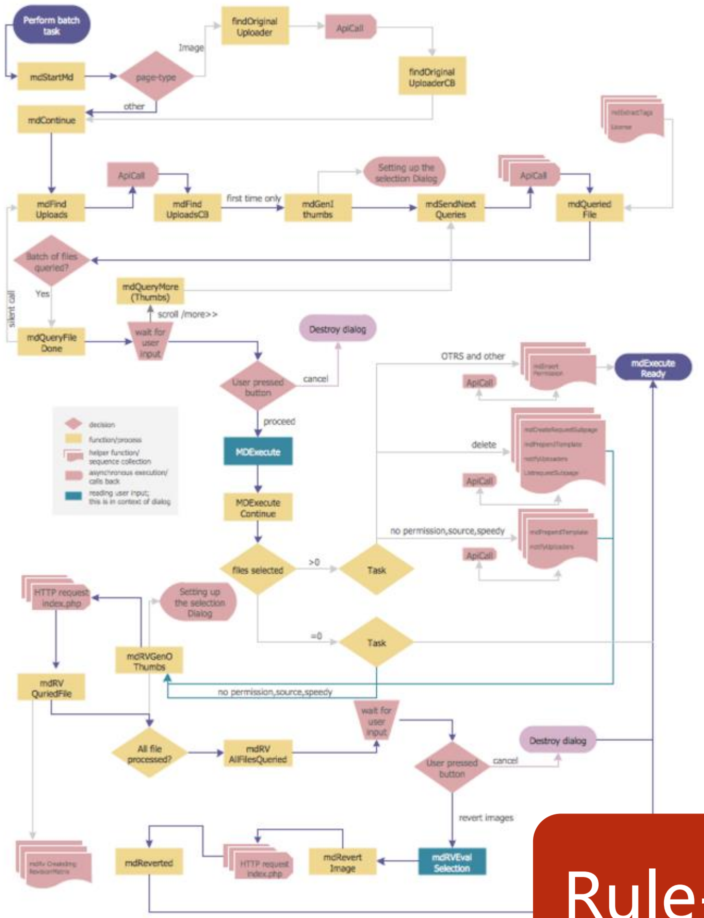
$$\underset{3\times 1}{[\ ]} = \underset{3\times N}{[\ \ \ ]}\,\underset{N\times 1}{[\ ]}$$

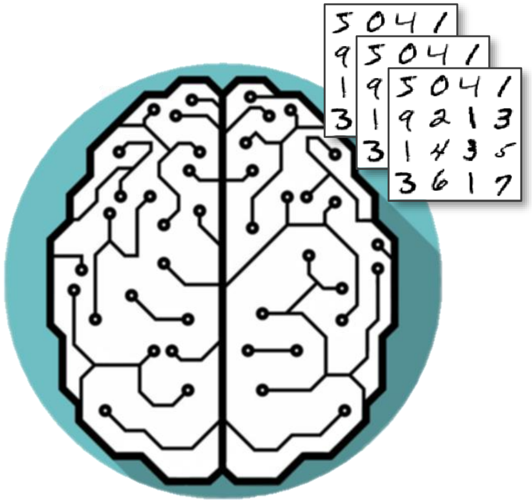
N*1维特征变为3*1维特征，只需要找到一个3*N的向量

将n维空间映射到三维空间



Rule-based system VS Representation learning

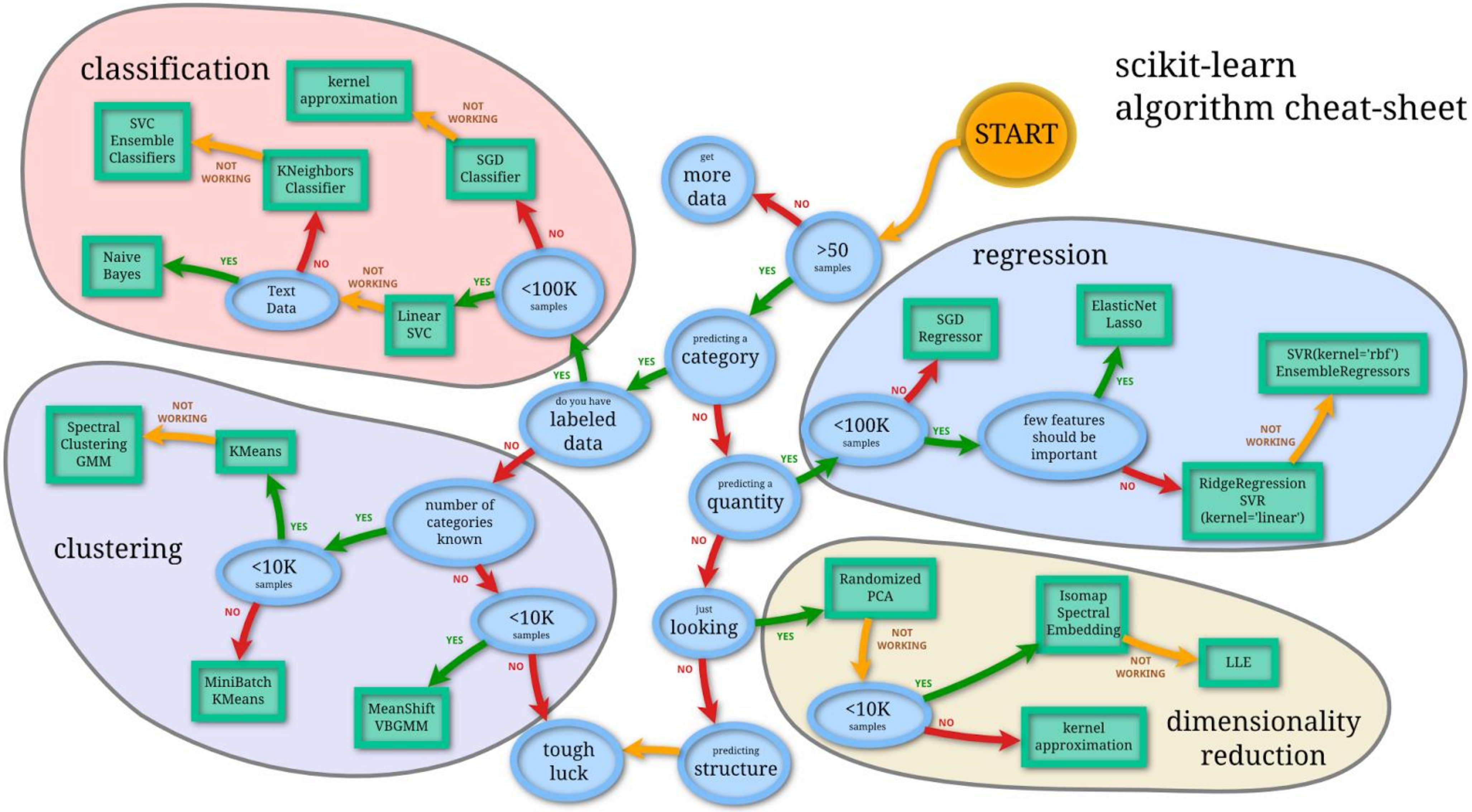




Rule-based system

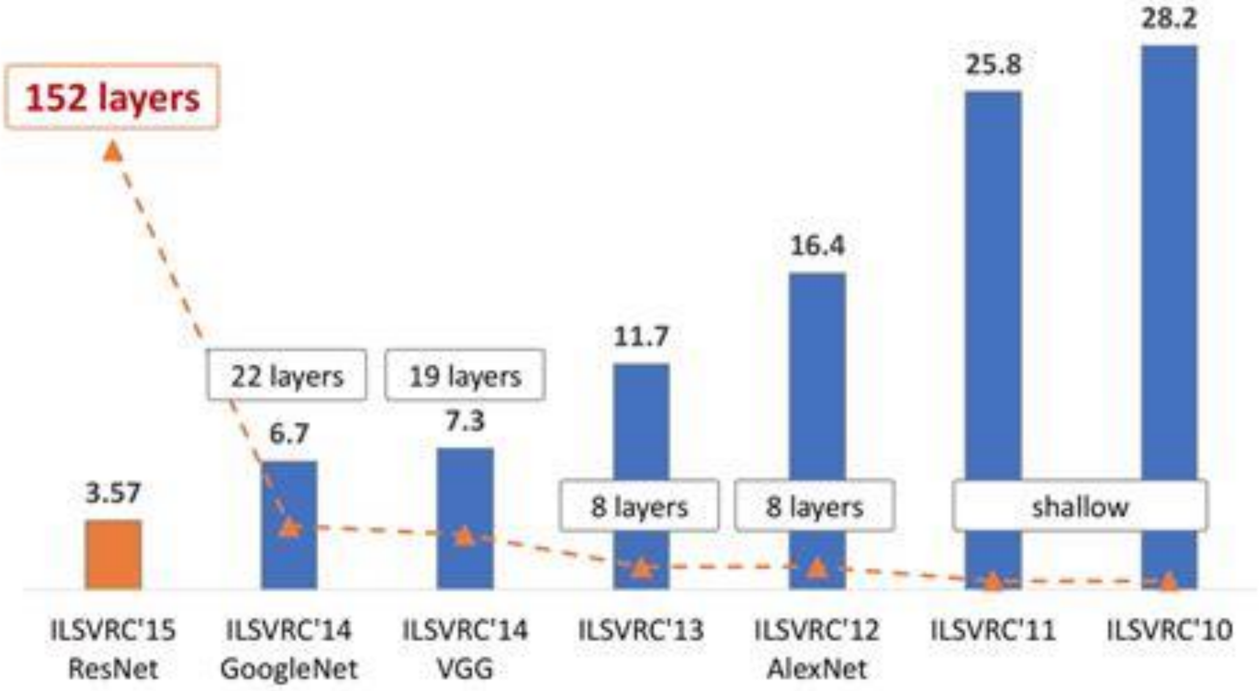
Training from data

Traditional machine learning strategy



New challenge

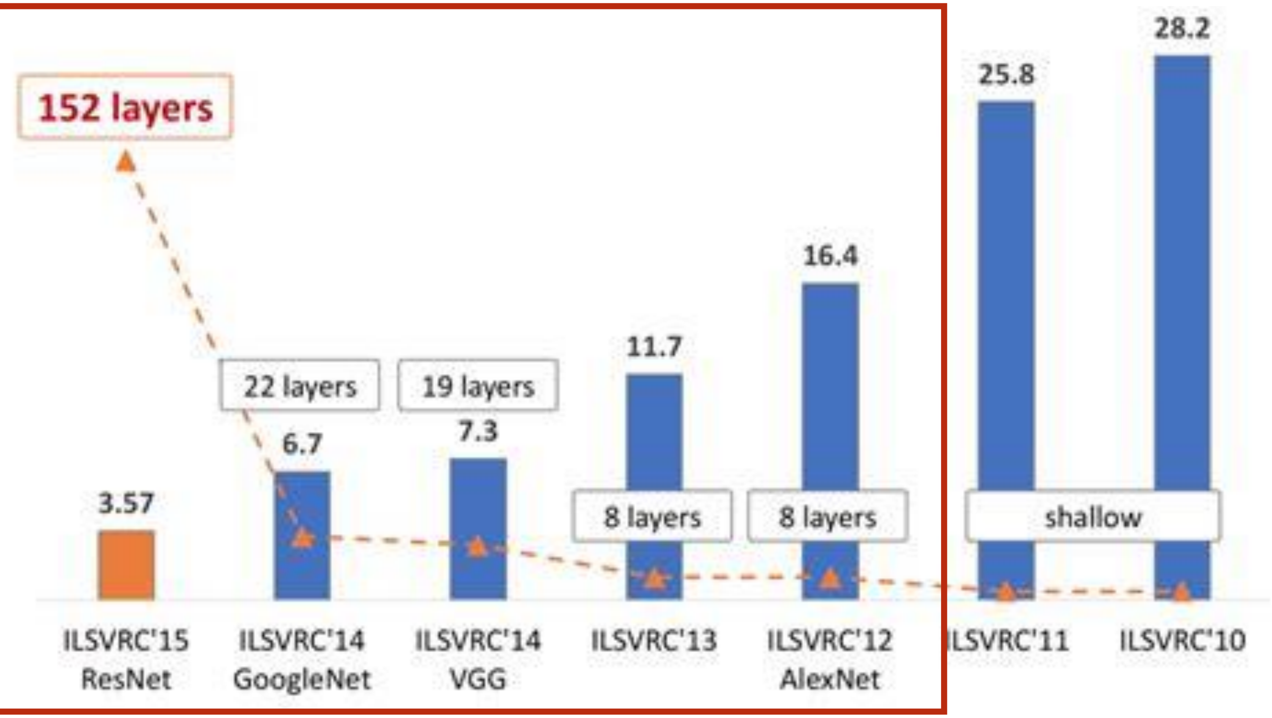
- Limit of hand-designed feature.
- SVM can not handle big data set well.
- More and more application need to handle unstructured data.

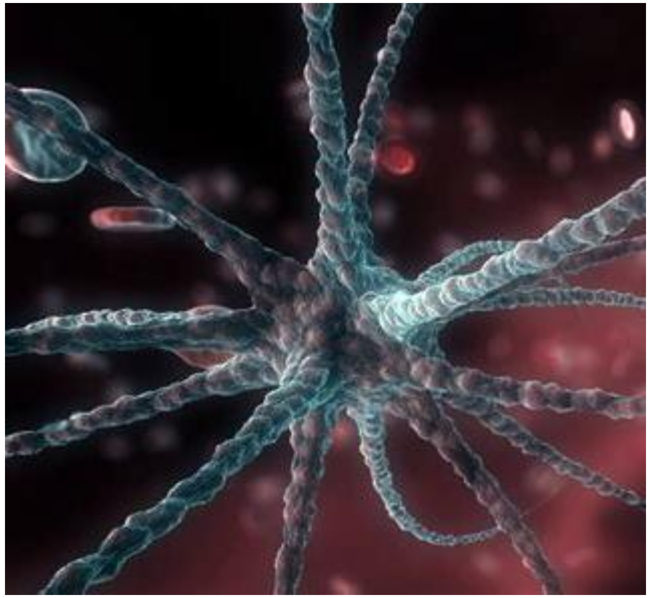


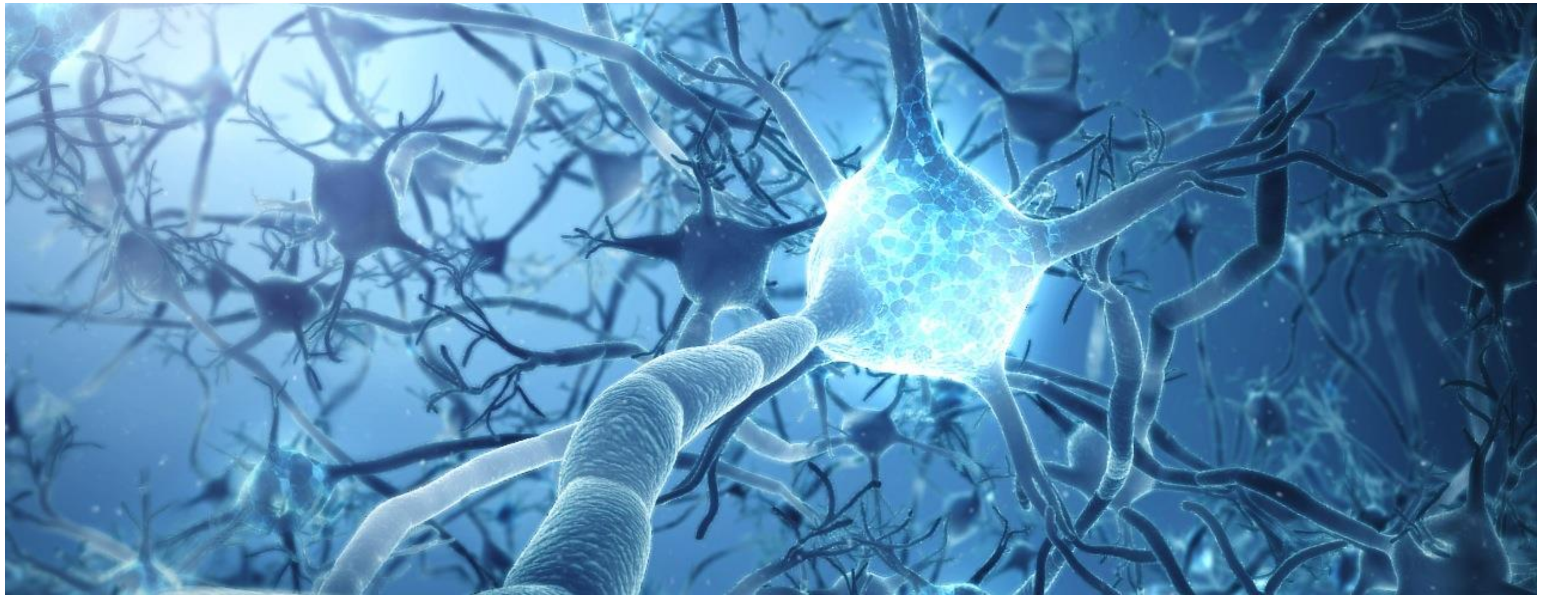
New challenge

- Limit of hand-designed feature.
- SVM can not handle big data set well.
- More and more application need to handle unstructured data.

Deep Learning







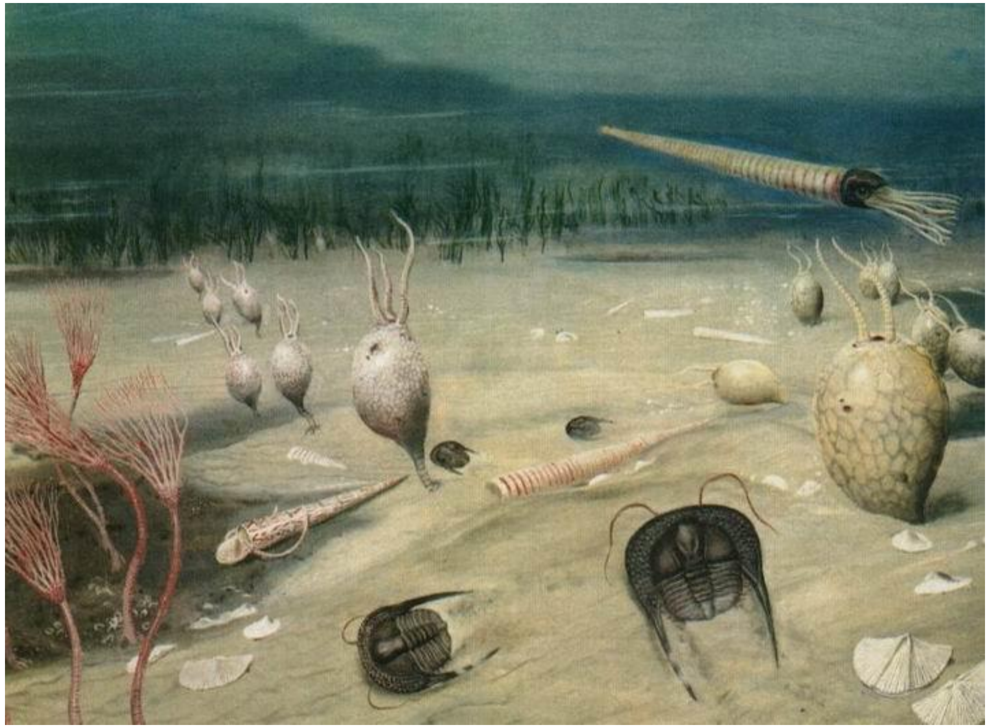
Brief history of neural networks

From neuroscience to mathematic & engineering

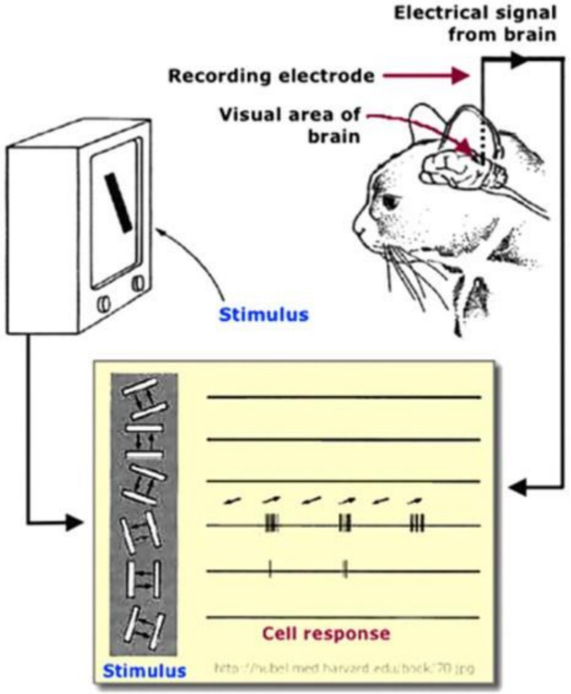
Brief history of neural networks

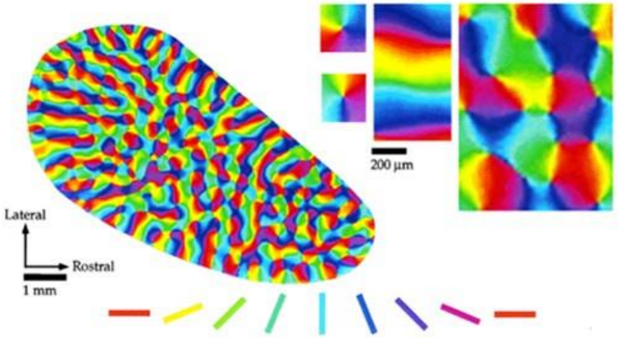
Cambrian
Period

543 million years, B.C.



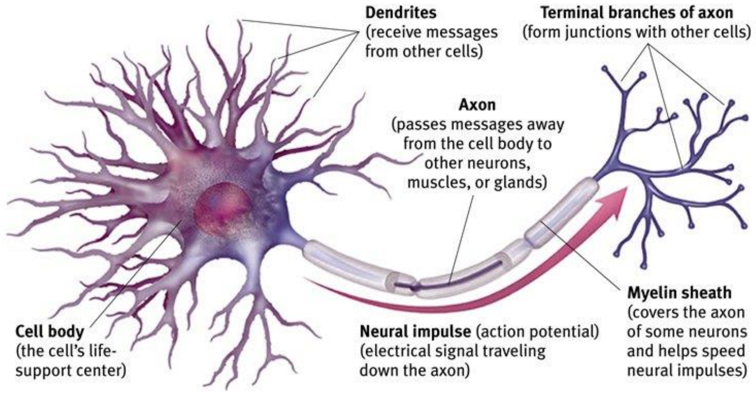
Brief history of neural networks



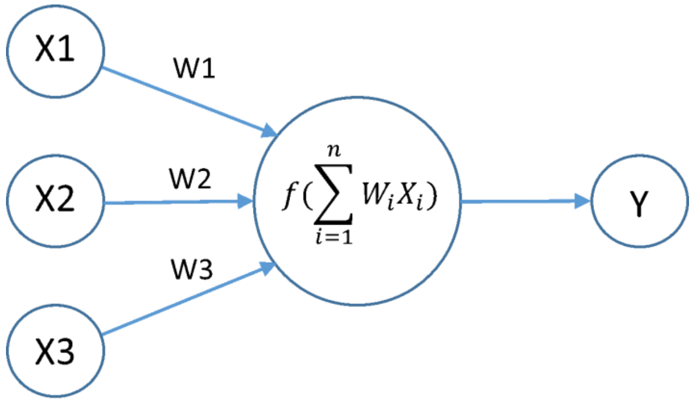


Hubel D H, Wiesel T N. Receptive fields of single neurones in the cat's striate cortex[J]. Journal of Physiology, 1959, 148(3):574.

Brief history of neural networks

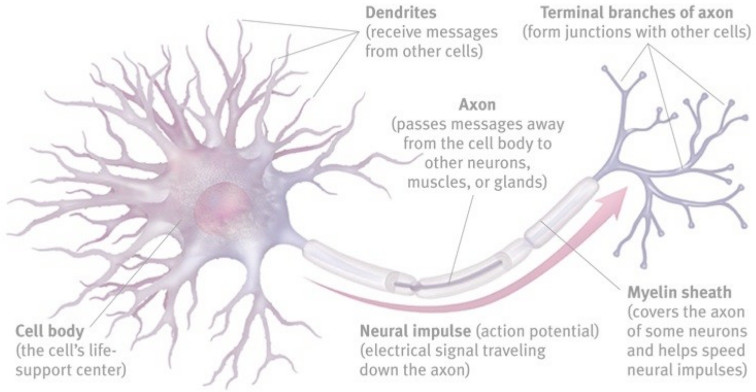


Single Neural Unit

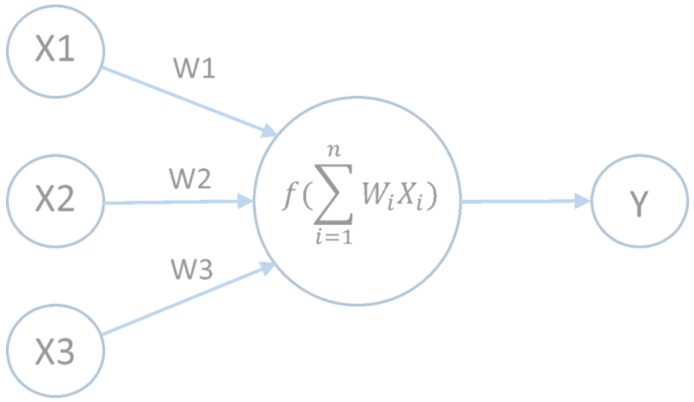


Perceptron 感知机

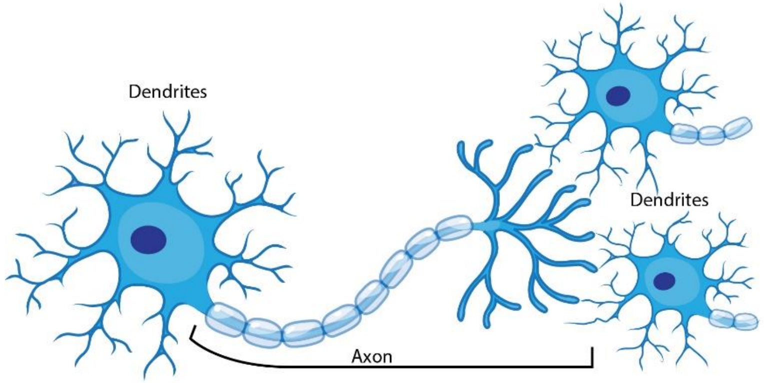
Brief history of neural networks



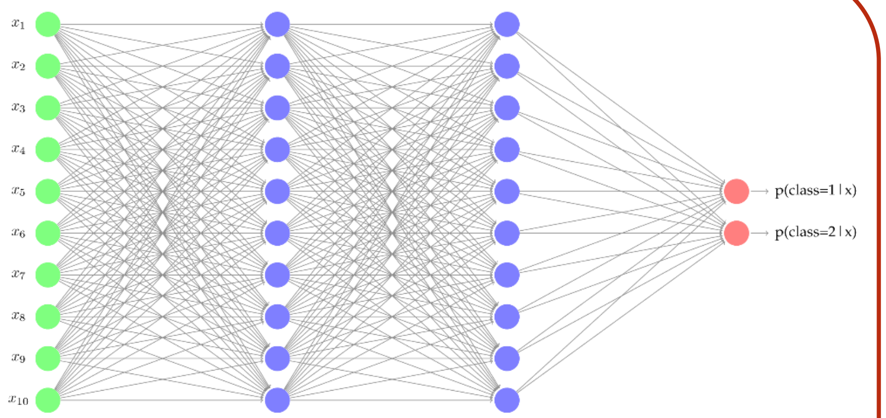
Single Neural Unit



Perceptron



Connected Neural Units



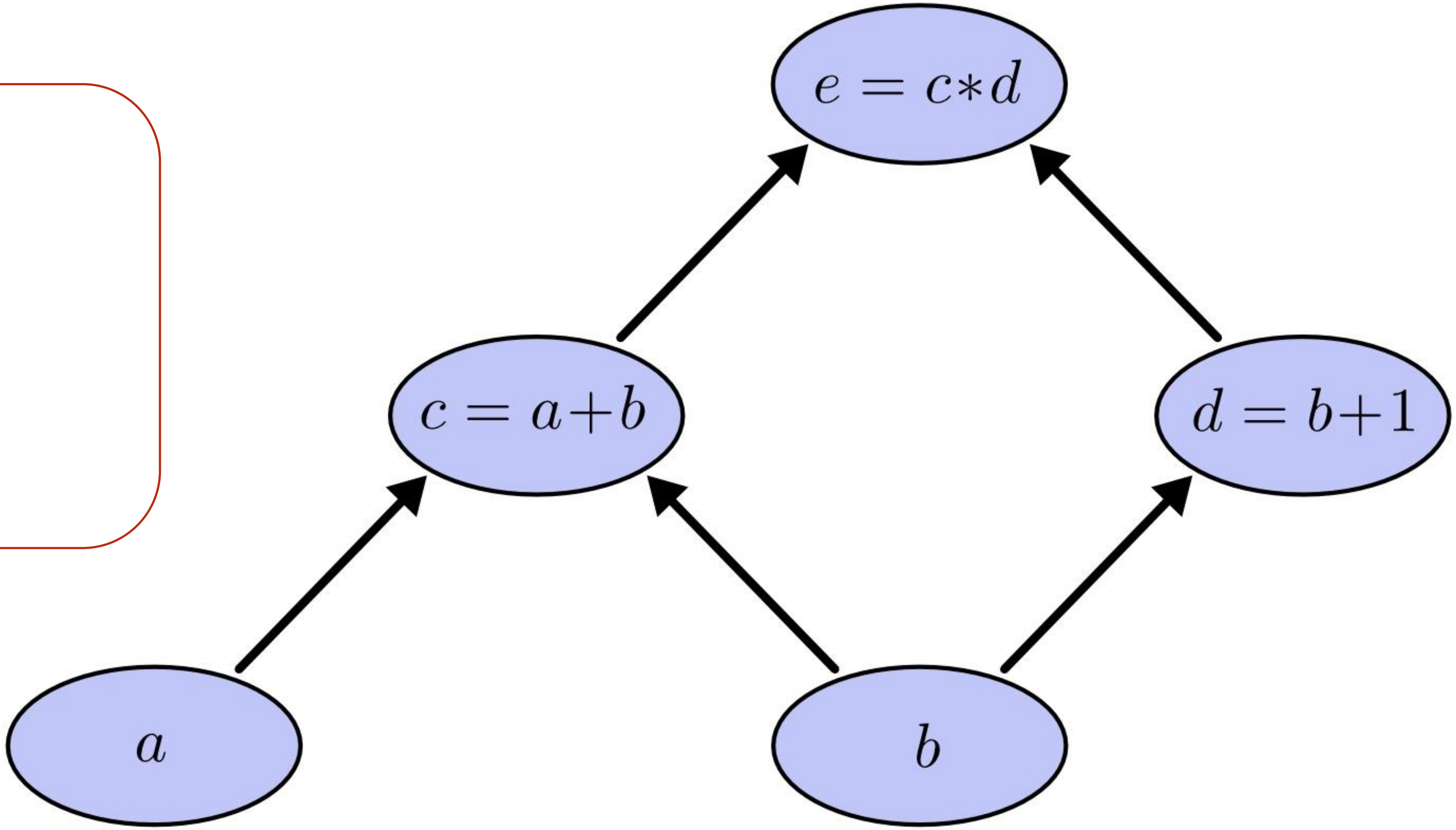
Artificial Neural Network

Back Propagation

$$\frac{\partial e}{\partial a} = ?$$

$$\frac{\partial e}{\partial b} = ?$$

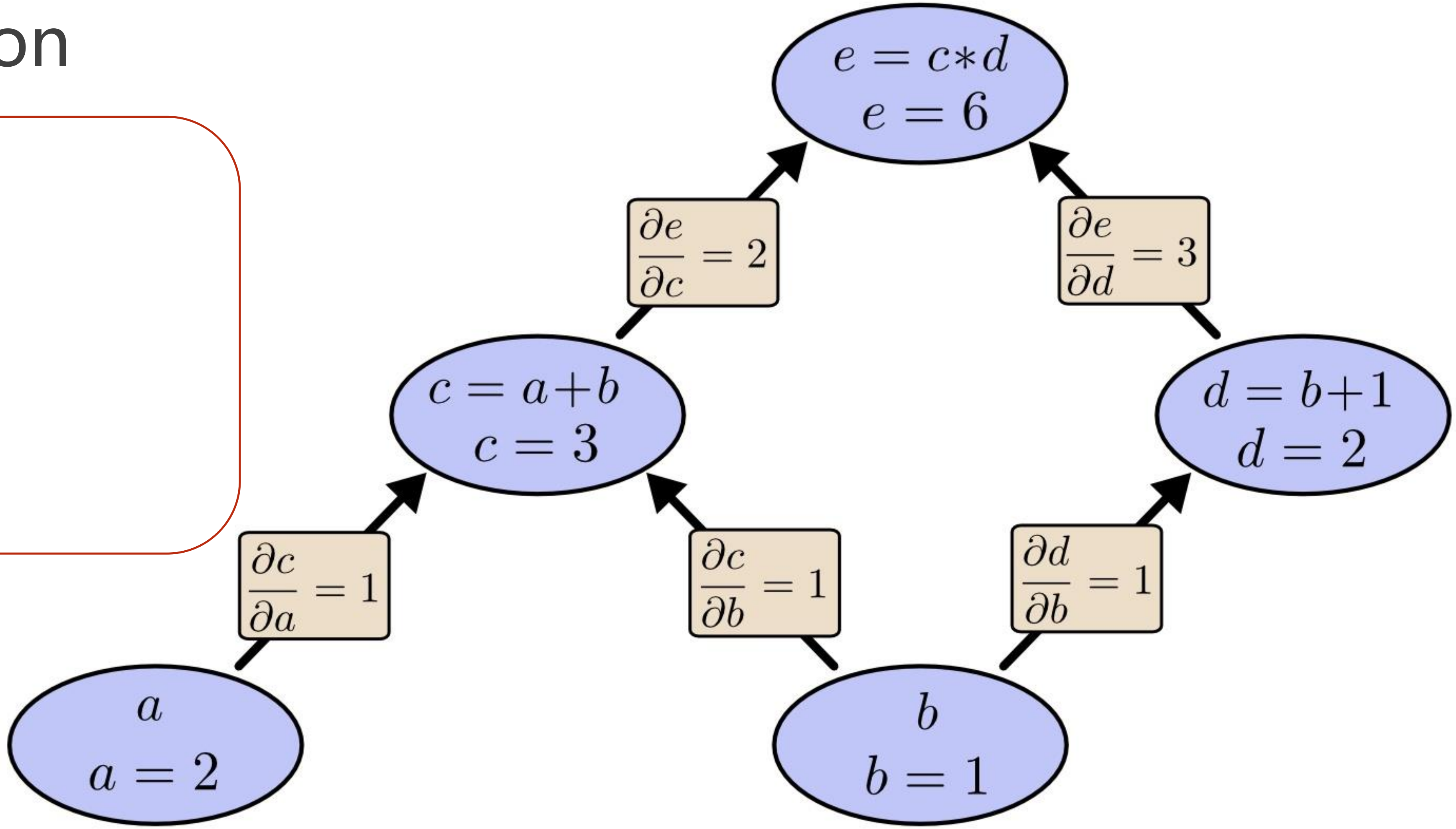
计算图



Back Propagation

$$\frac{\partial e}{\partial a} = ?$$

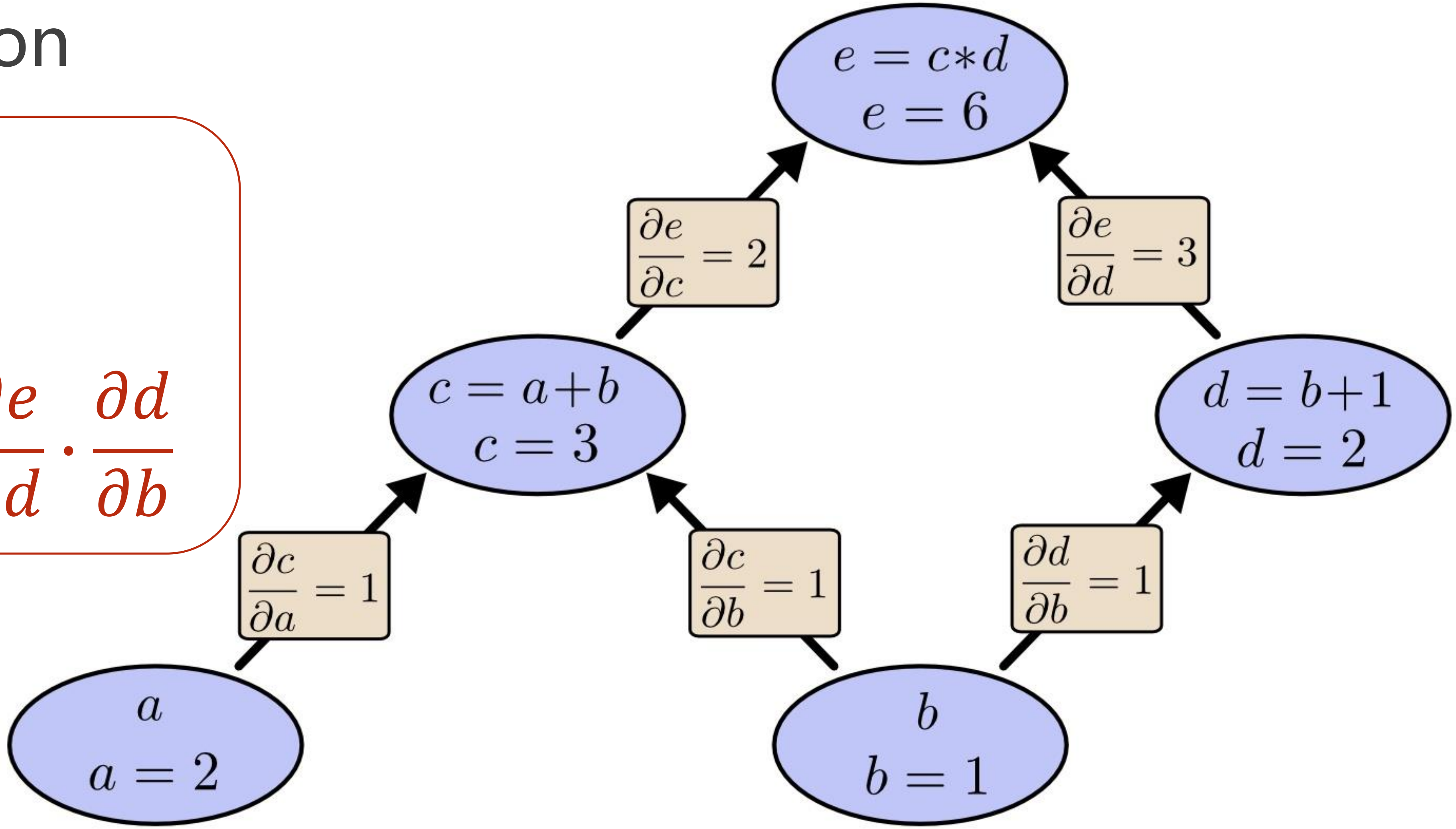
$$\frac{\partial e}{\partial b} = ?$$



Back Propagation

$$\frac{\partial e}{\partial a} = \frac{\partial e}{\partial c} \cdot \frac{\partial c}{\partial a}$$

$$\frac{\partial e}{\partial b} = \frac{\partial e}{\partial c} \cdot \frac{\partial c}{\partial b} + \frac{\partial e}{\partial d} \cdot \frac{\partial d}{\partial b}$$



LeNet-5 LeCun 1998

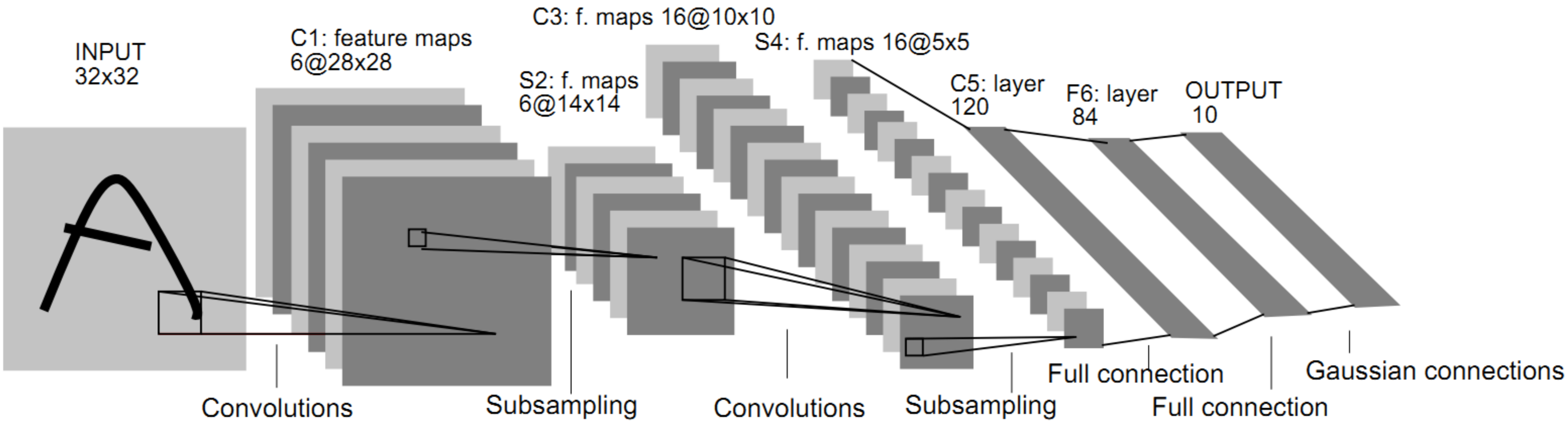
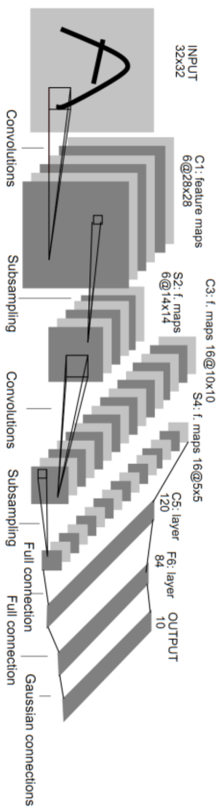


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

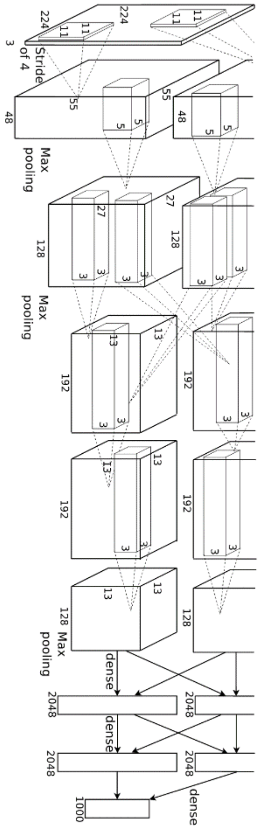
Y. LeCun, L. Bottou, Y. Bengio and P. Haffner: Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, 86(11):2278-2324, November 1998,

Brief history of neural networks

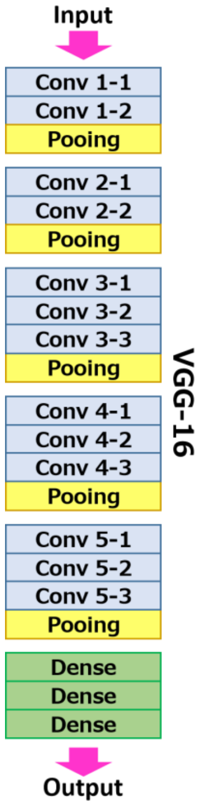
[1998] LeNet-5



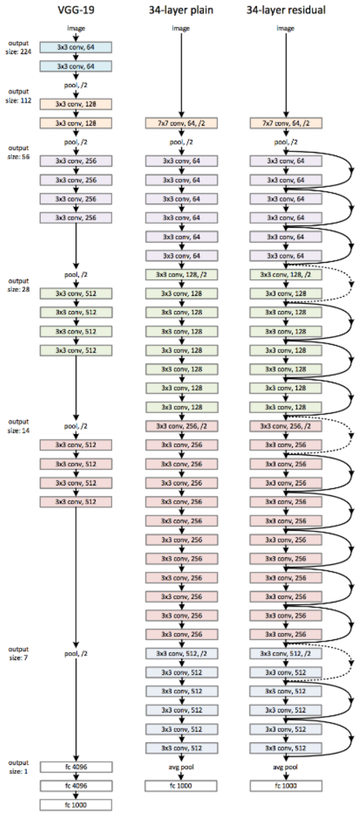
[2012] AlexNet



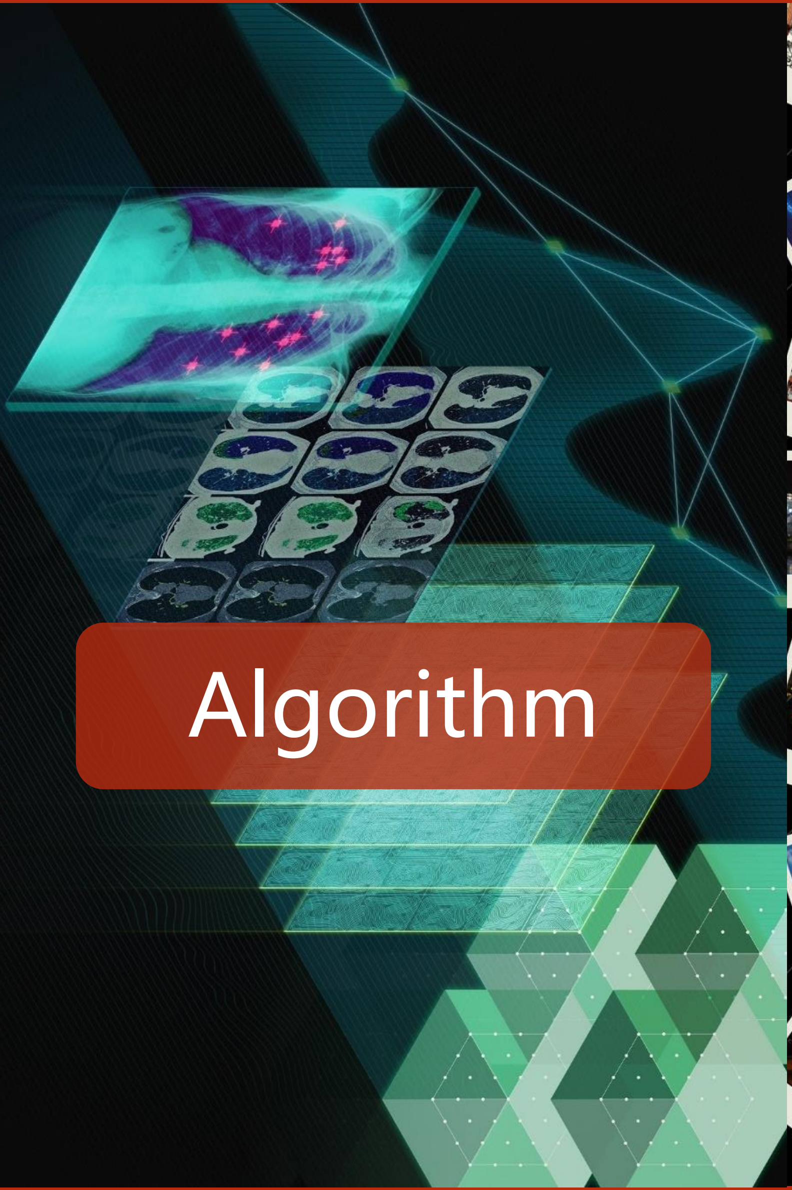
[2014] GoogLeNet & VGG



[2015] ResNet



Brief history of neural networks



Algorithm

Data

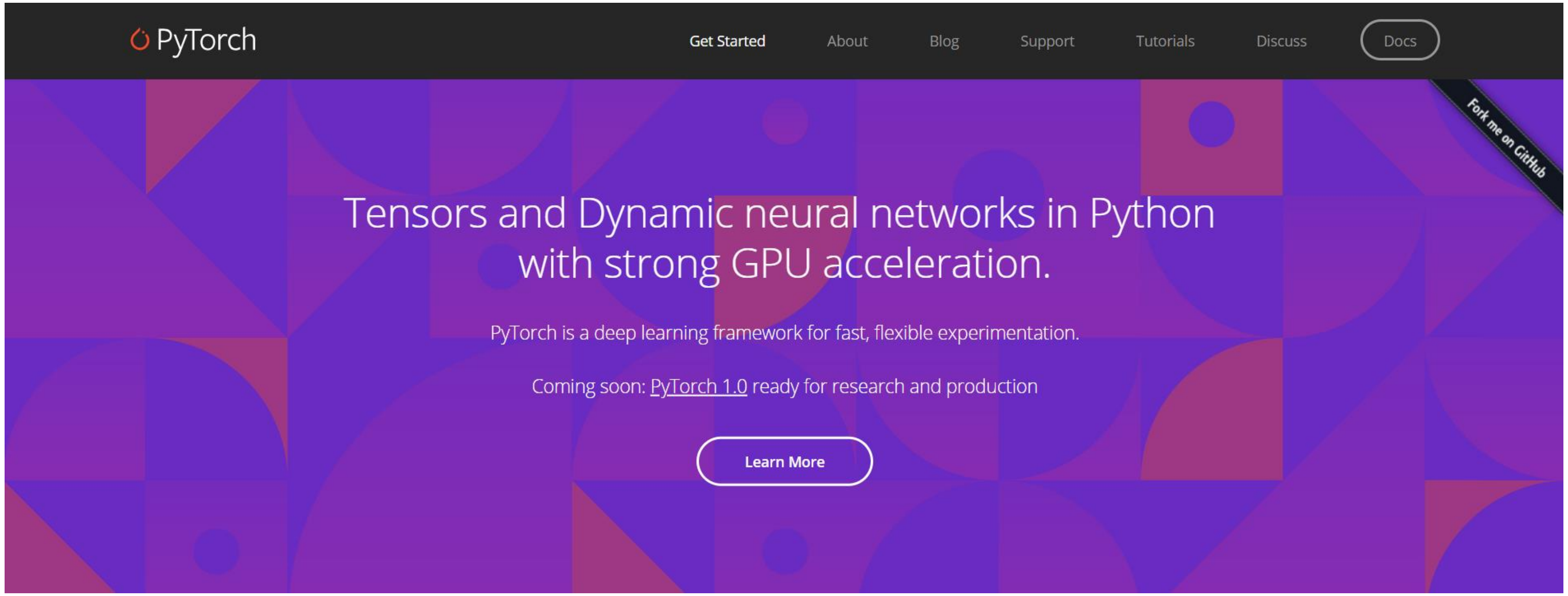
Computation

Good news

- Deep learning is not too difficult
  - Basic algebra + probability + python
  - Less than one year study
- There are lots of deep learning framework
  - Starting from scratch do not be required
  - Enabled efficient and convenient use of GPU
  - Lots of components of neural networks provided by framework
- Popular deep learning frameworks
  - Theano (University of Montreal) / TensorFlow (Google)
  - Caffe (UC Berkeley) / Caffe 2 (Facebook)
  - Torch (NYU & Facebook) / **PyTorch** (Facebook)

What is PyTorch

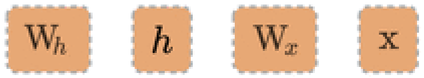
- PyTorch is a python package that provides two high-level features:
  - Tensor computation (like numpy) with strong GPU acceleration
  - Deep Neural Networks built on a tape-based autodiff system



Why PyTorch

- Dynamical graph
  - More flexible
  - Easy to debug
  - Intuitive and cleaner code
- More neural networkic
  - Write code as network works
  - AutoGrad for forward / backward

A graph is created on the fly




```
x = torch.randn(1, 10)
prev_h = torch.randn(1, 20)
W_h = torch.randn(20, 20)
W_x = torch.randn(20, 10)
```

Install PyTorch

Get Started.

Select your preferences, then run the PyTorch install command.

Please ensure that you are on the latest pip and numpy packages.
Anaconda is our recommended package manager

OS	Linux	MacOS	Windows	
Package Manager	conda	pip	Source	
Python	2.7	3.5	3.6	
CUDA	8	9.0	9.1	None

Run this command:

```
conda install pytorch cuda90 -c pytorch
pip3 install torchvision
```

Click here for previous versions of PyTorch

https://pytorch.org

After install PyTorch on your computer

```
PS C:\Users\liuii> python
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> print(torch.__version__)
0.4.0
>>> # Perfect!
```

PyTorch Tutorial

01. Overview