Deep Learning

Concepts and Its Applications

Thanapat Kangkachit, Ph.D.

Master of Engineering (Big Data Engineering)

College of Innovative Technology and Engineering (CITE)

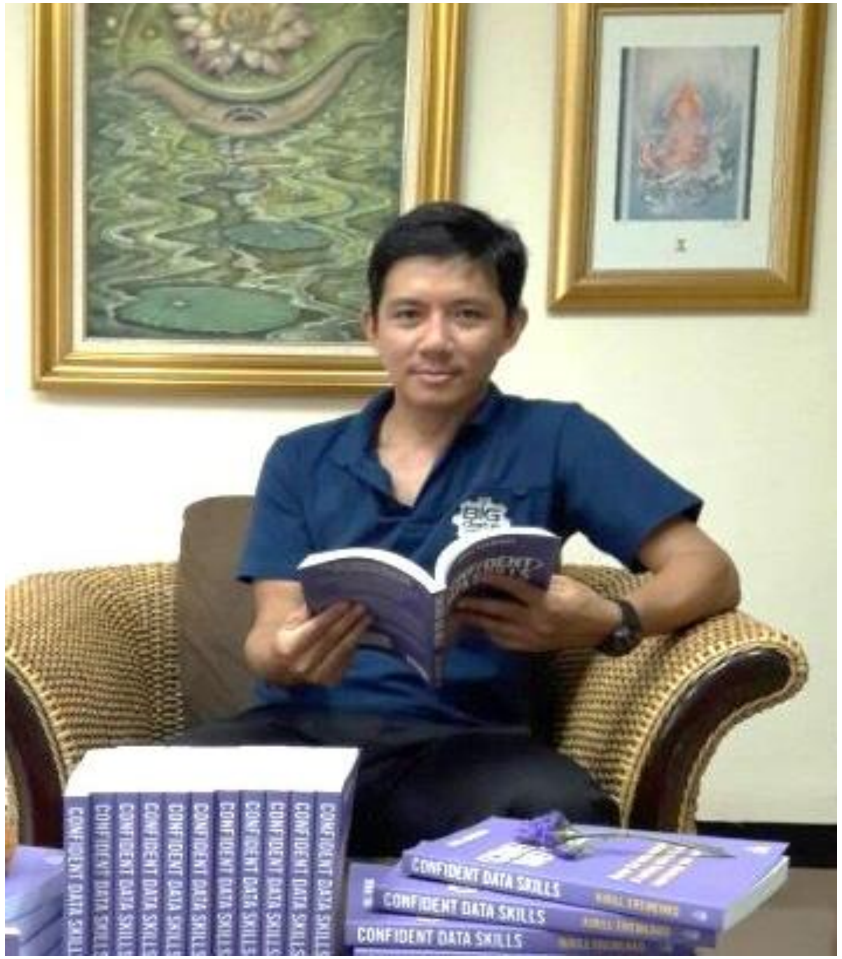
2 June 2018

About Me

Name: Thanapat Kangkachit
ธนภัทร ฆังคะจิตร

Work: Lecturer at Big Data Engineering, CITE, DPU since 2017

Education:
- Ph.D. in Computer Engineering, Kasetsart University
- M.Eng. in Computer Engineering, Kasetsart University
- B.Eng. in Computer Engineering, Kasetsart University



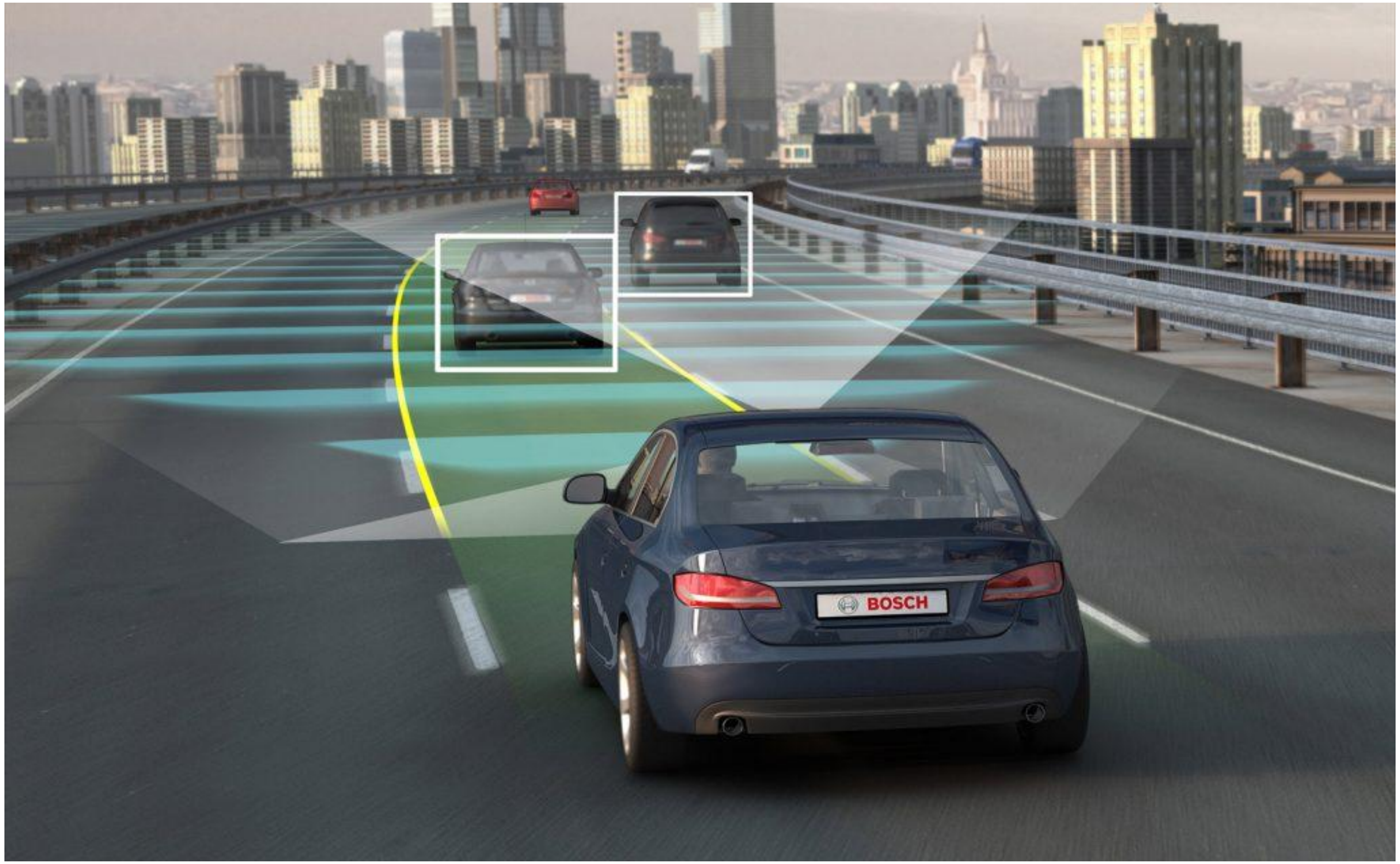
Agenda

- 01 – What is AI, Machine Learning and Deep Learning?
- 02 – Applications of Deep Learning
- 03 – Getting Start with Deep Learning

Google DeepMind [2016]

Reference [1]

Self-Driving Cars



Reference [2]

Real-time Face Recognition [2018]

Reference [3] (starting from 9th minute)

01

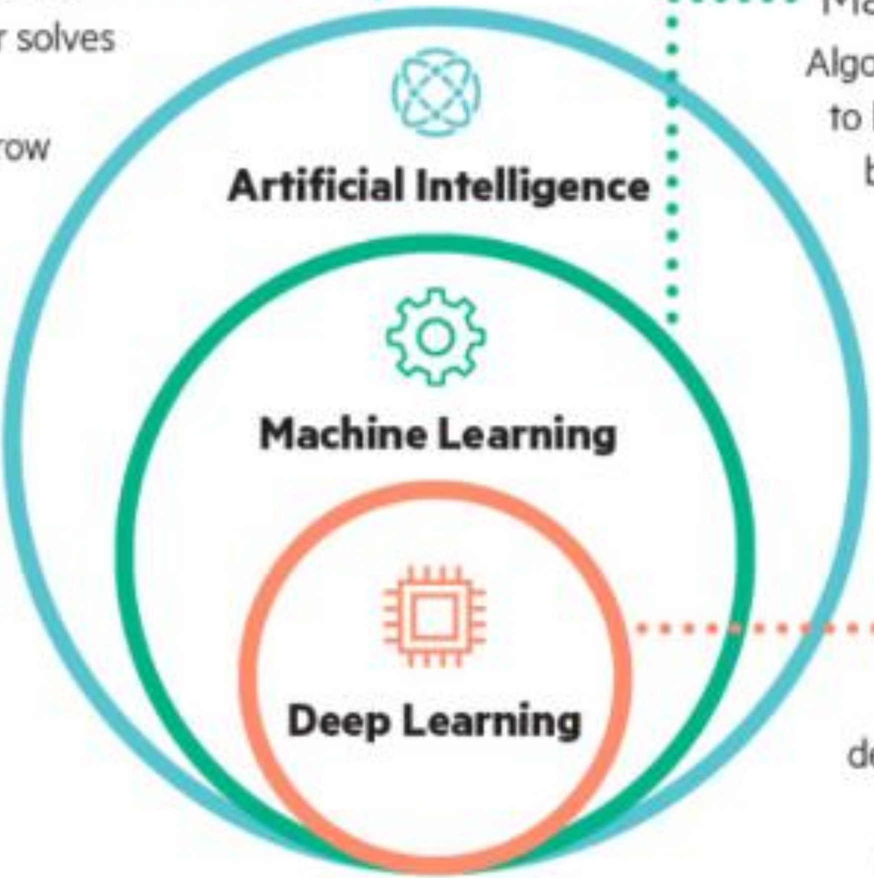
What is AI, Machine Learning and Deep Learning?

What Makes a Machine Intelligent?

While AI is the headliner, there are actually subsets of the technology which can be applied to solving human problems in different ways.

Artificial Intelligence (AI)

A process where a computer solves a task in a way that mimics human behavior. Today, narrow AI—when a machine is trained to do one particular task—is becoming more widely used, from virtual assistants to self-driving cars to automatic tagging your friends in your photos on Facebook.



Machine Learning (ML)

Algorithms that allow computers to learn from examples without being explicitly programmed.

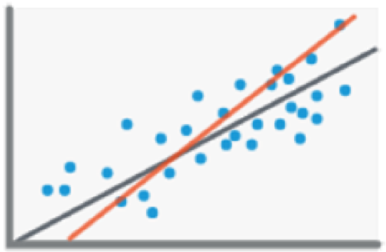
Deep Learning (DL)

A subset of ML which uses deep artificial neural networks as models and does not require feature engineering.

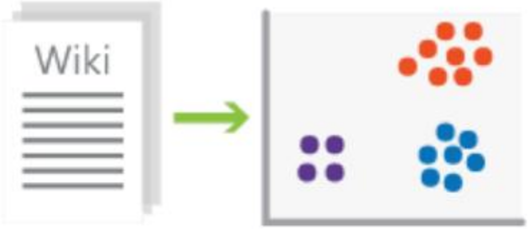
By Curt Hopkins, Managing Editor, Hewlett Packard Labs

Machine Learning: Problem Types

Classification
(supervised – predictive)



Regression
(supervised – predictive)



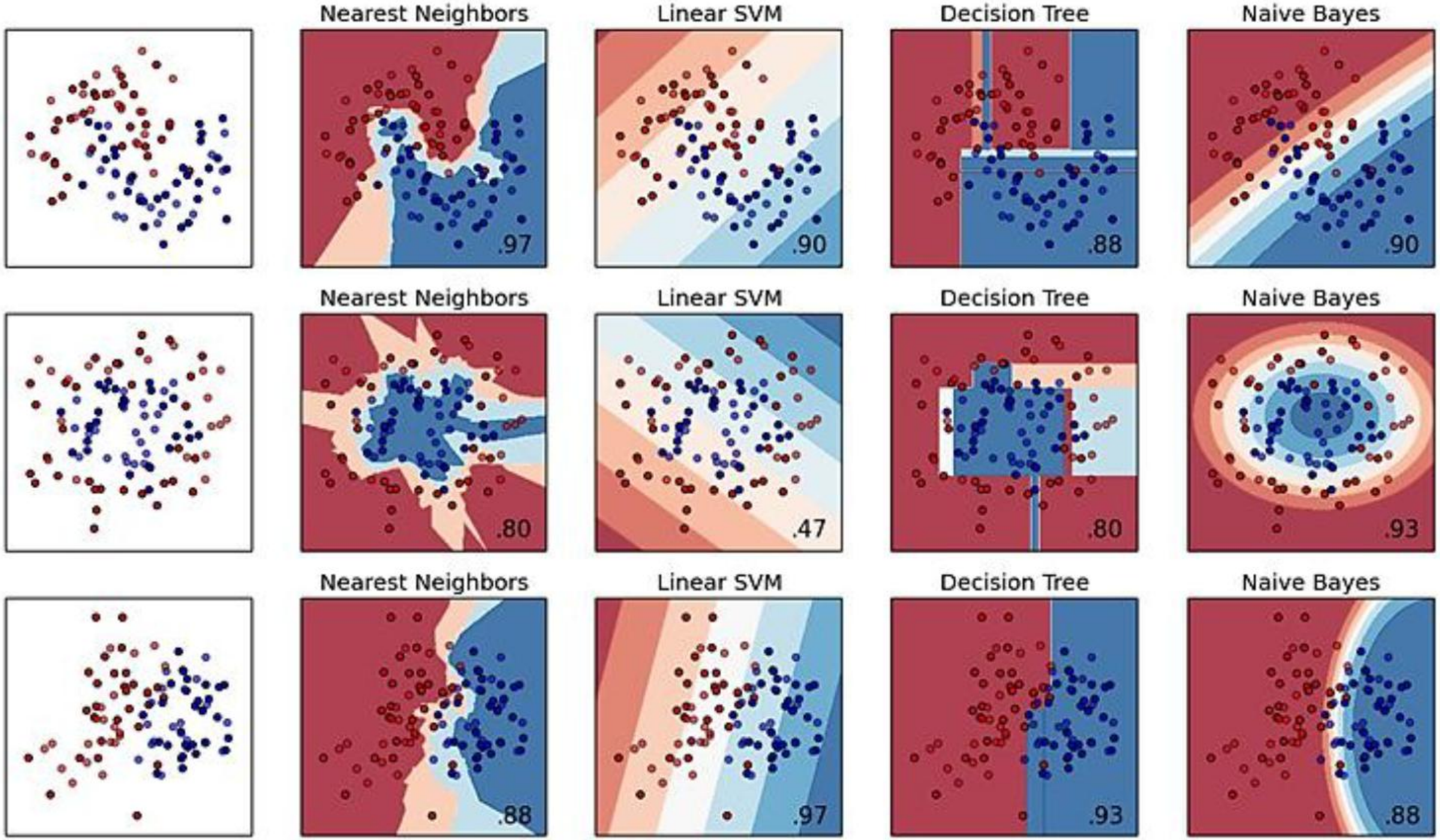
Clustering
(unsupervised – descriptive)

Anomaly Detection
(unsupervised– descriptive)

Reference [4]

College of Innovative Technology and Engineering

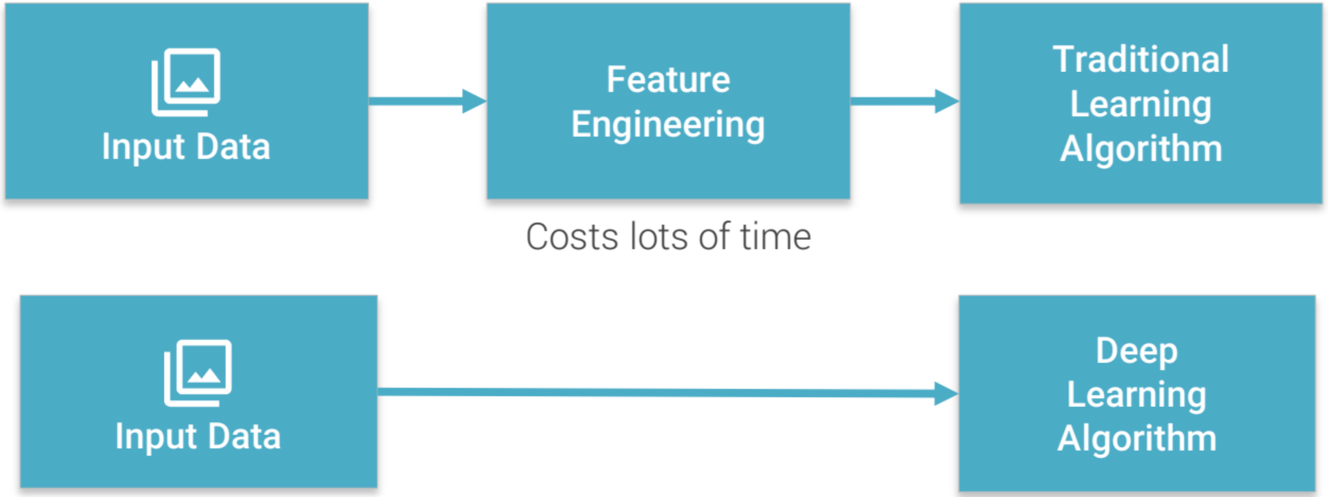
Machine Learning: Classification Algorithms



Reference [4]

What is Deep Learning?

- A subset of **machine learning** field
- Uses **deep artificial neural networks** as models
- Does not require feature engineering



Hype or Reality?

I have worked all my life in Machine Learning, and I've never seen one algorithm knock over benchmarks like Deep Learning
– Andrew Ng (Stanford & Baidu)

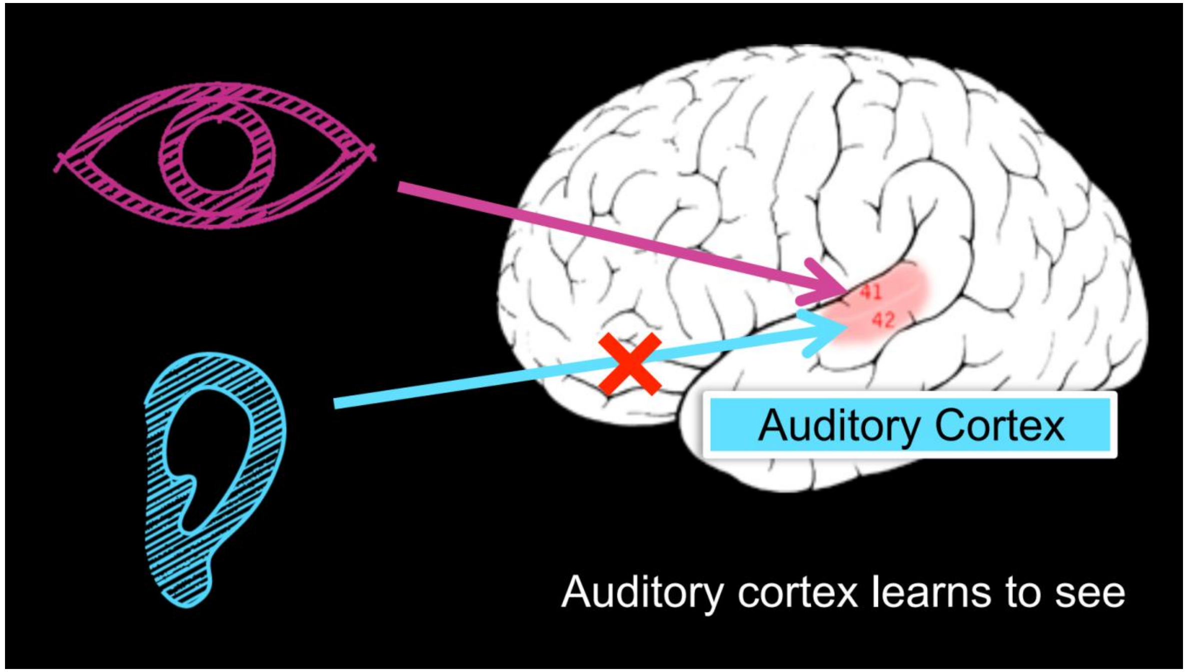
Deep Learning is an algorithm which has no theoretical limitations of what it can learn; the more data you give and the more computational time you provide, the better it is *– Geoffrey Hinton (Google)*

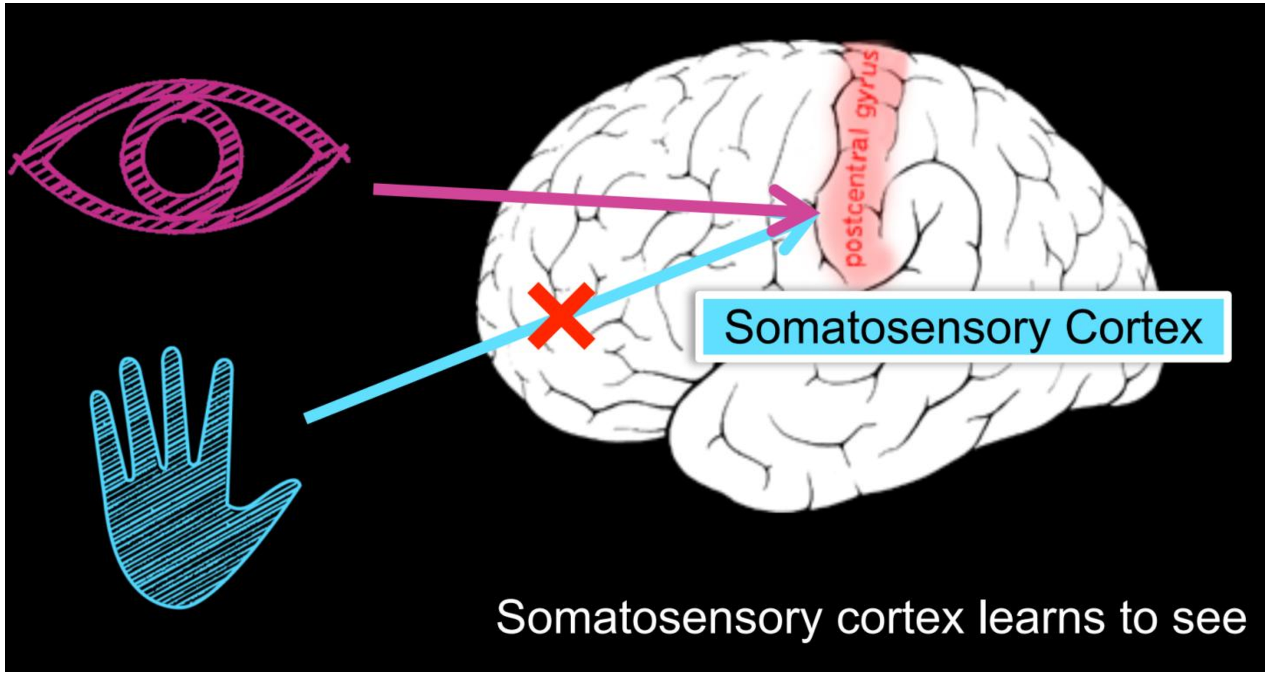
Human-level artificial intelligence has the potential to help humanity thrive more than any invention that has come before it *– Dileep George (Co-Founder Vicarious)*

For a very long time it will be a complementary tool that human scientists and human experts can use to help them with the things that humans are not naturally good *– Demis Hassabis (Co-Founder DeepMind)*

Reference [4]

The “one learning algorithm” hypothesis

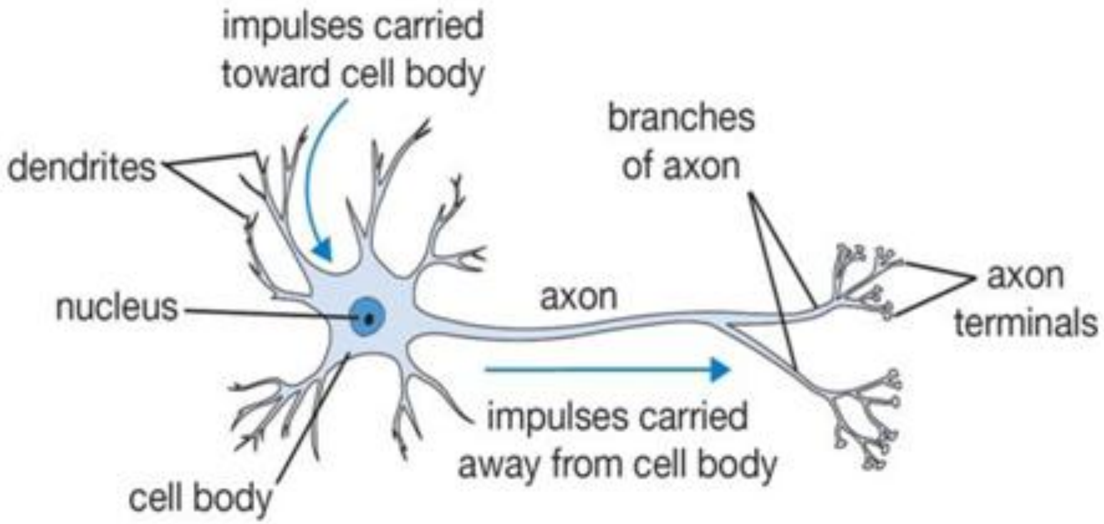


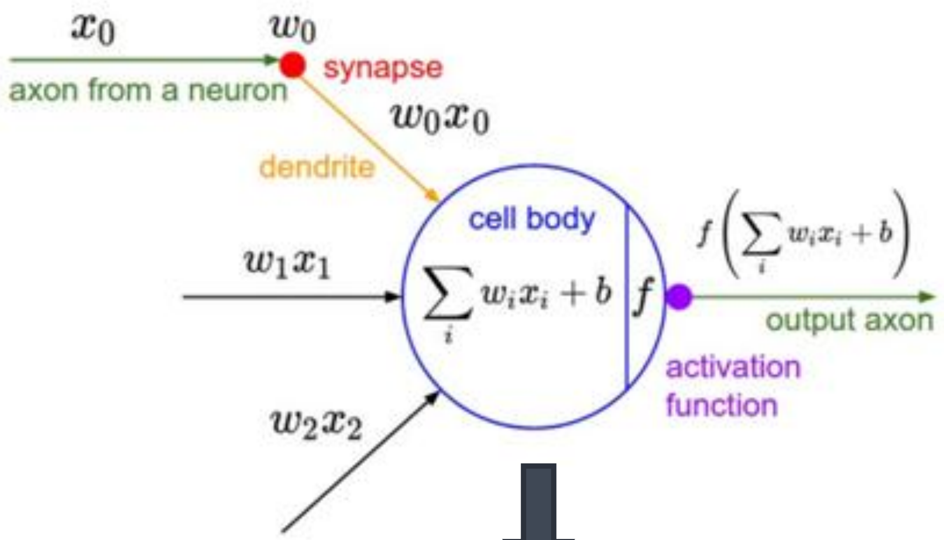


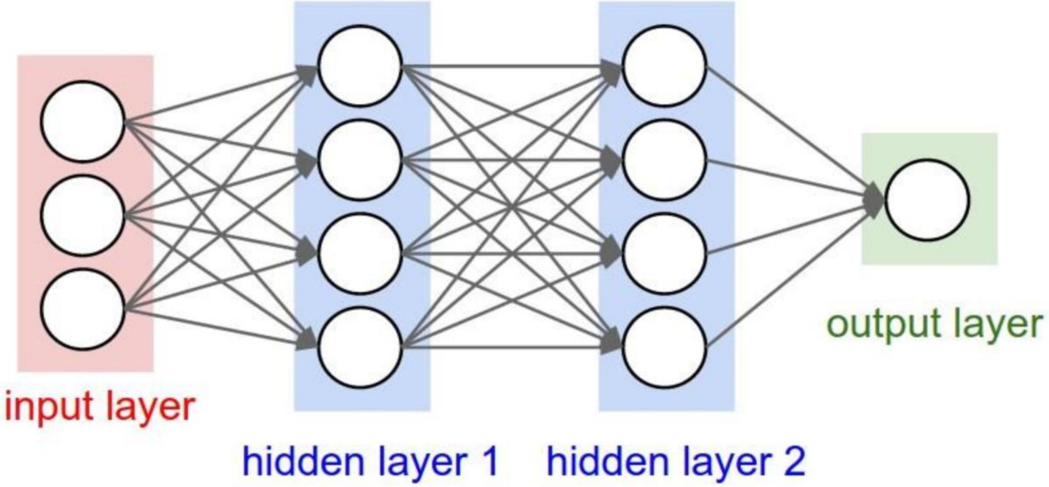
[Roe et al., 1992]

[Metin & Frost, 1989]

Neurons in the Brain VS Artificial Neurons



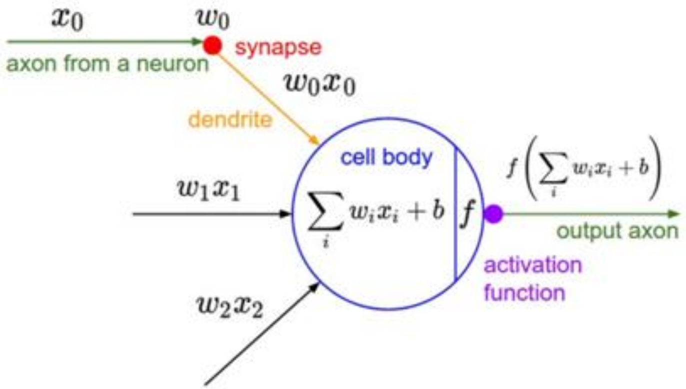


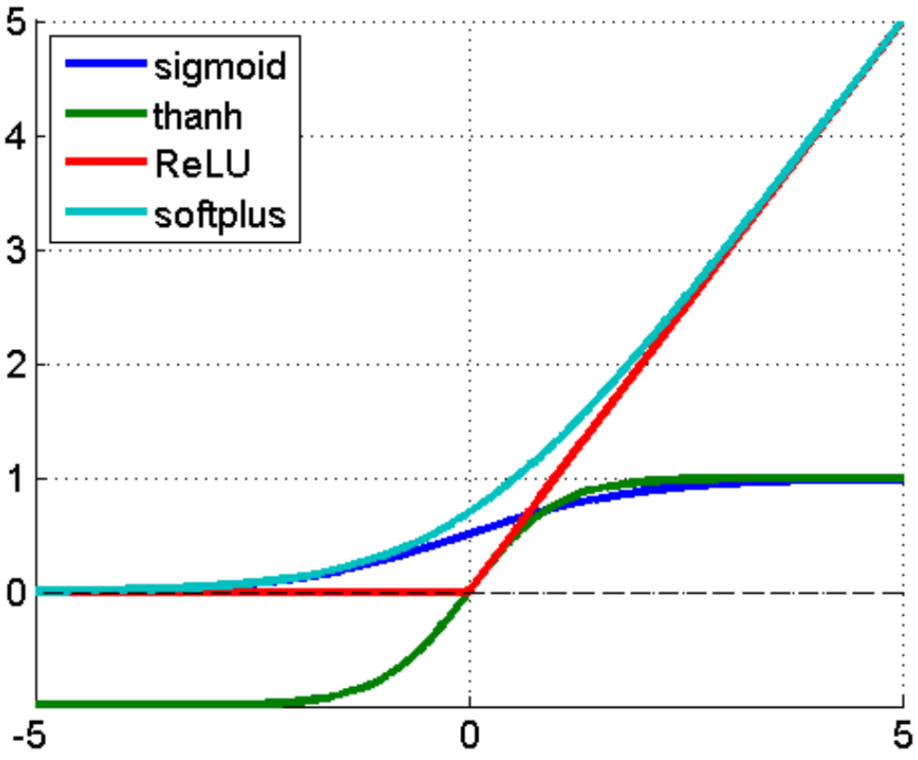


Artificial Neural Networks

Reference [4]

Non-linear Activation Function



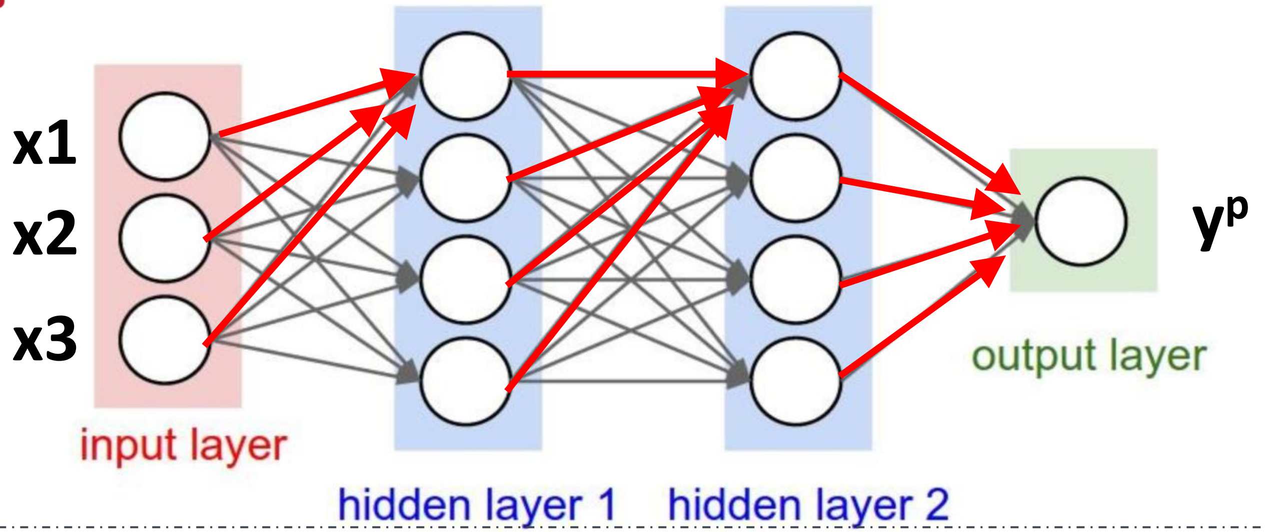


ReLU, max(0, x)

- used by **most deep networks**
- much faster training time
- prevent **gradient vanishing** problem

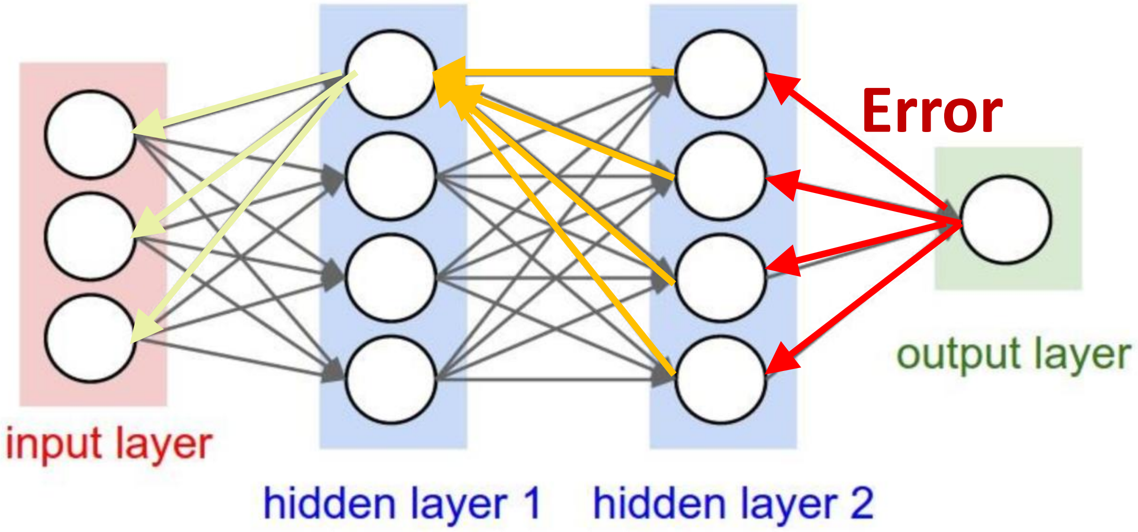
Reference [4]

Artificial Neural Networks: The Training Process



Forward Propagation

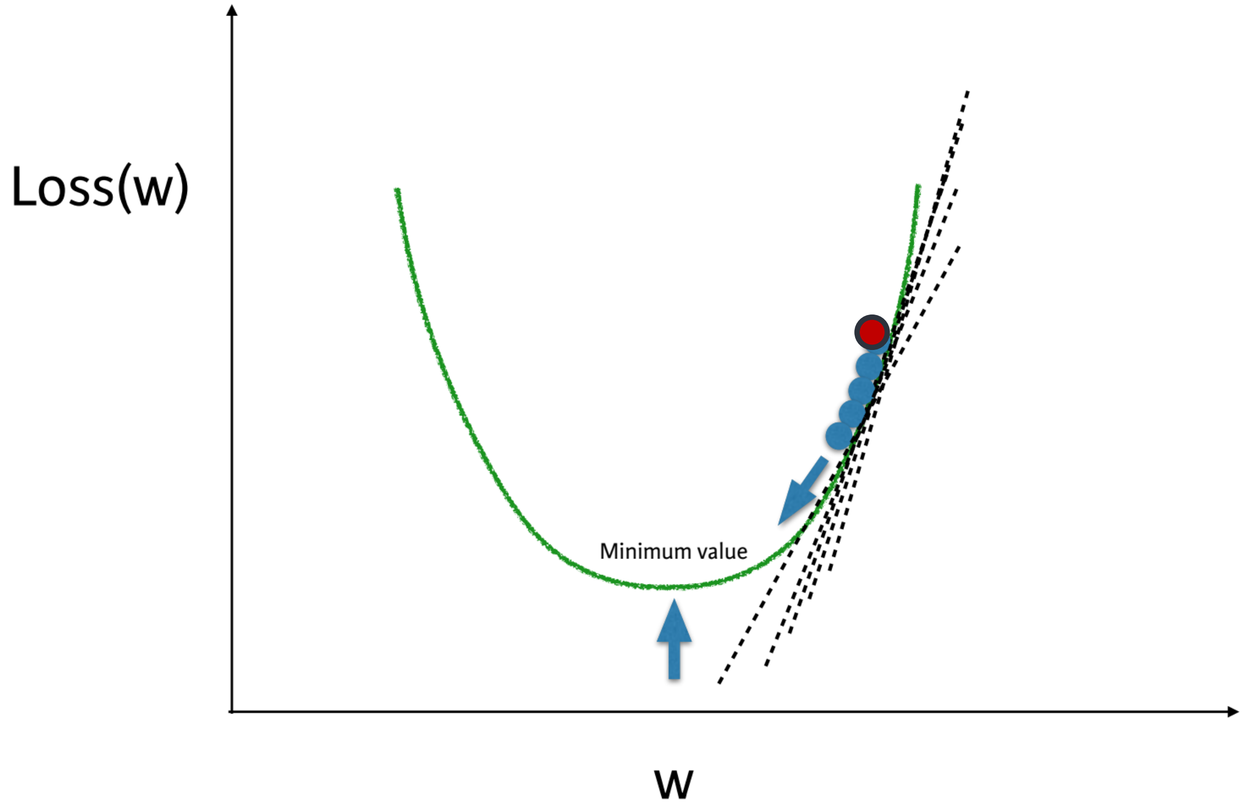
- Sample labeled data (x1, x2, x3, **Y**)
- **Forward** it through the network to get **predictions (y^p)**



Backward Propagation

- Compute the **Error** (**$Y - y^p$**)
- Update the connection weight using **gradient descent**

Gradient Descent



Imagine you are in **a pitch dark field** and want to find the **lowest point**

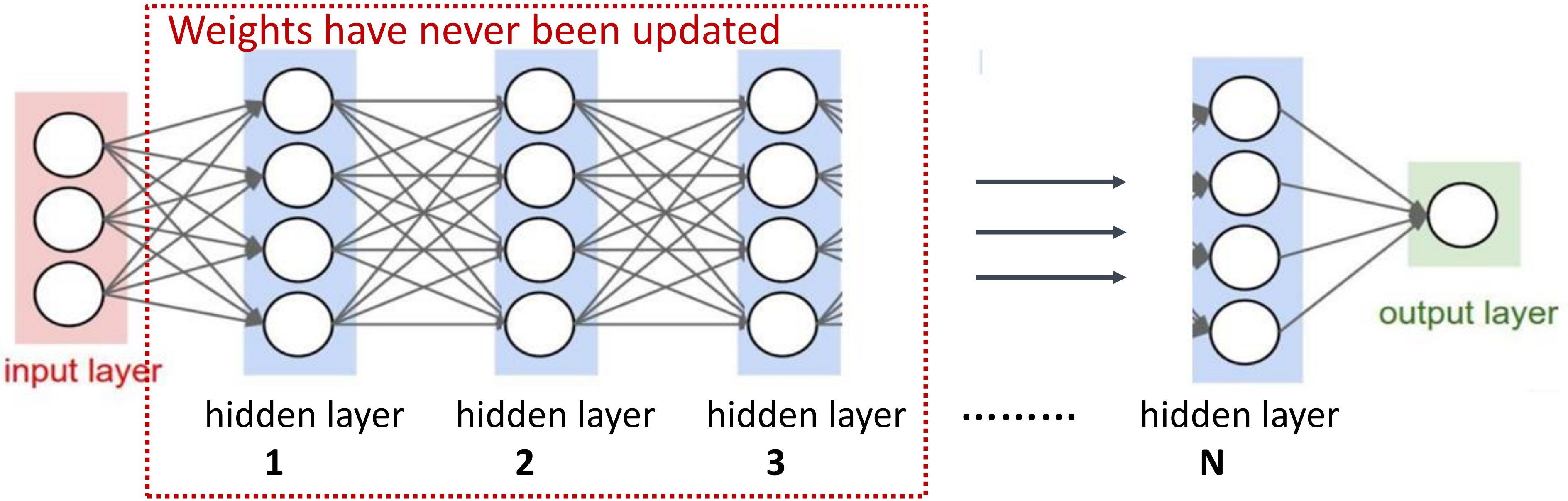
- **Feel the ground** to see how it slopes
- Take a **small step downhill** (learning rate)
- **Repeat until** it is **uphill** in every direction

Update each weight (w)

w_{new} = w – (learning rate * slope)

Reference [5]

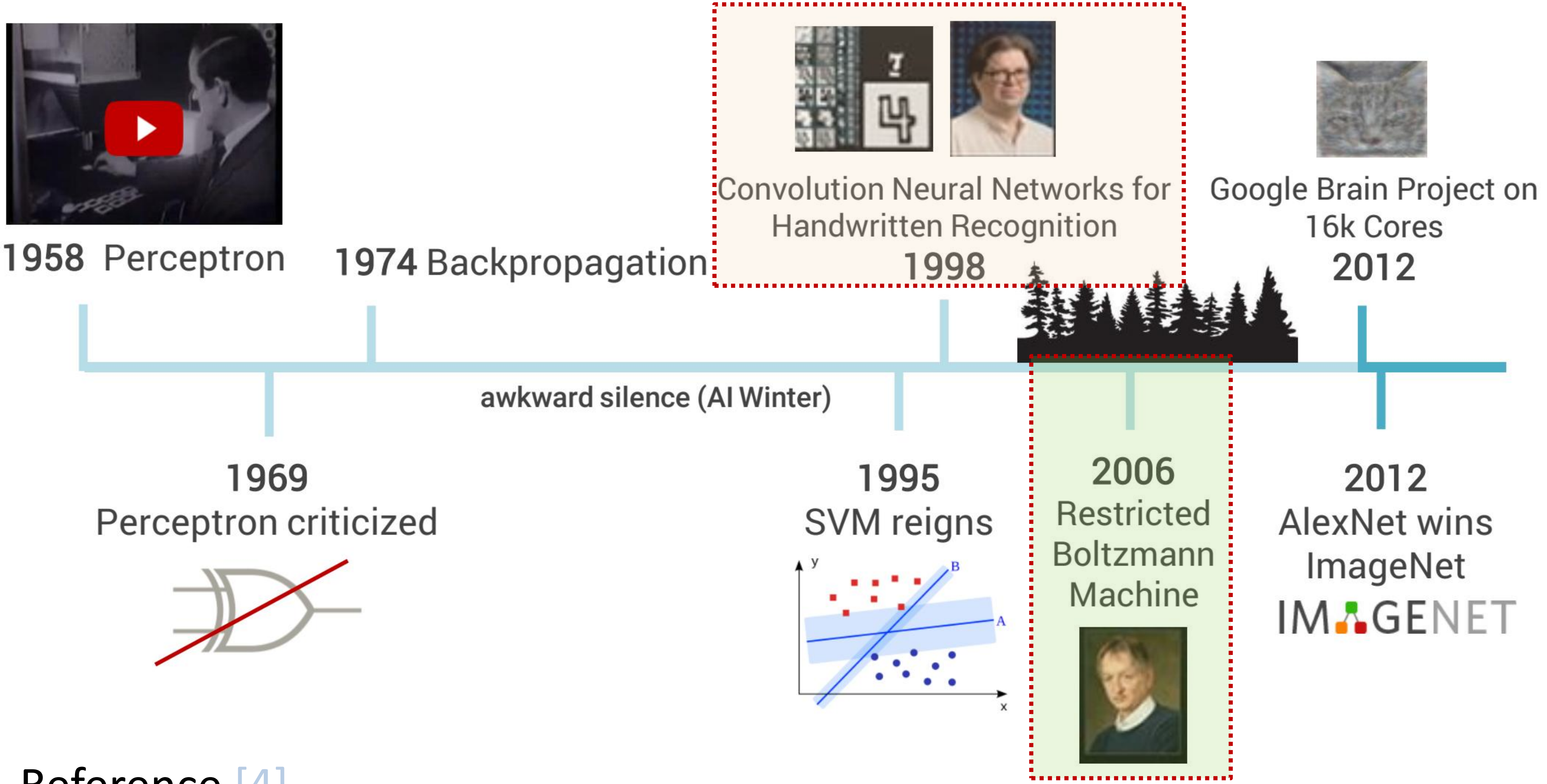
Vanishing Gradient in Deep Networks



- Connection **weights** of the **first** couple hidden layers have **never** been **updated**.
  - Unfortunately, they are **random weights.**

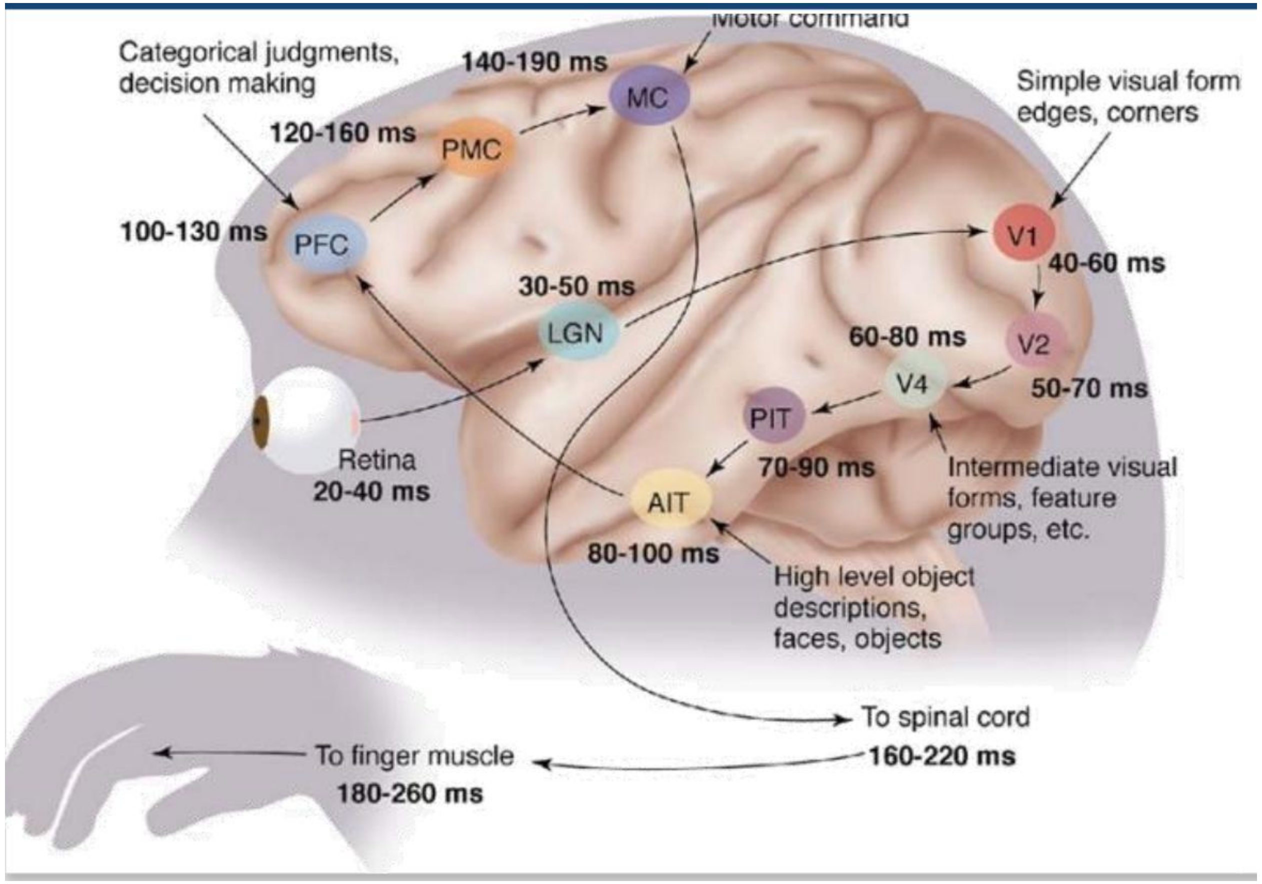
- In 2006, **Geoff Hinton et. al**. showed how a many-layered neural networks could be effectively **pre-trained one layer at a time**, treating each layer in turn as an unsupervised **restricted Boltzmann machine**, then fine-tuning it using supervised backpropagation.

A brief History of ANNs



Reference [4]

Inspired by the Visual Cortex Brain



In the visual cortex:

- The first hierarchy of neurons (**V1**) are sensitive to specific edges, corners.
- The brain regions further down the visual pipeline (PIT, AIT) are sensitive to more complex structure such as faces, objects.

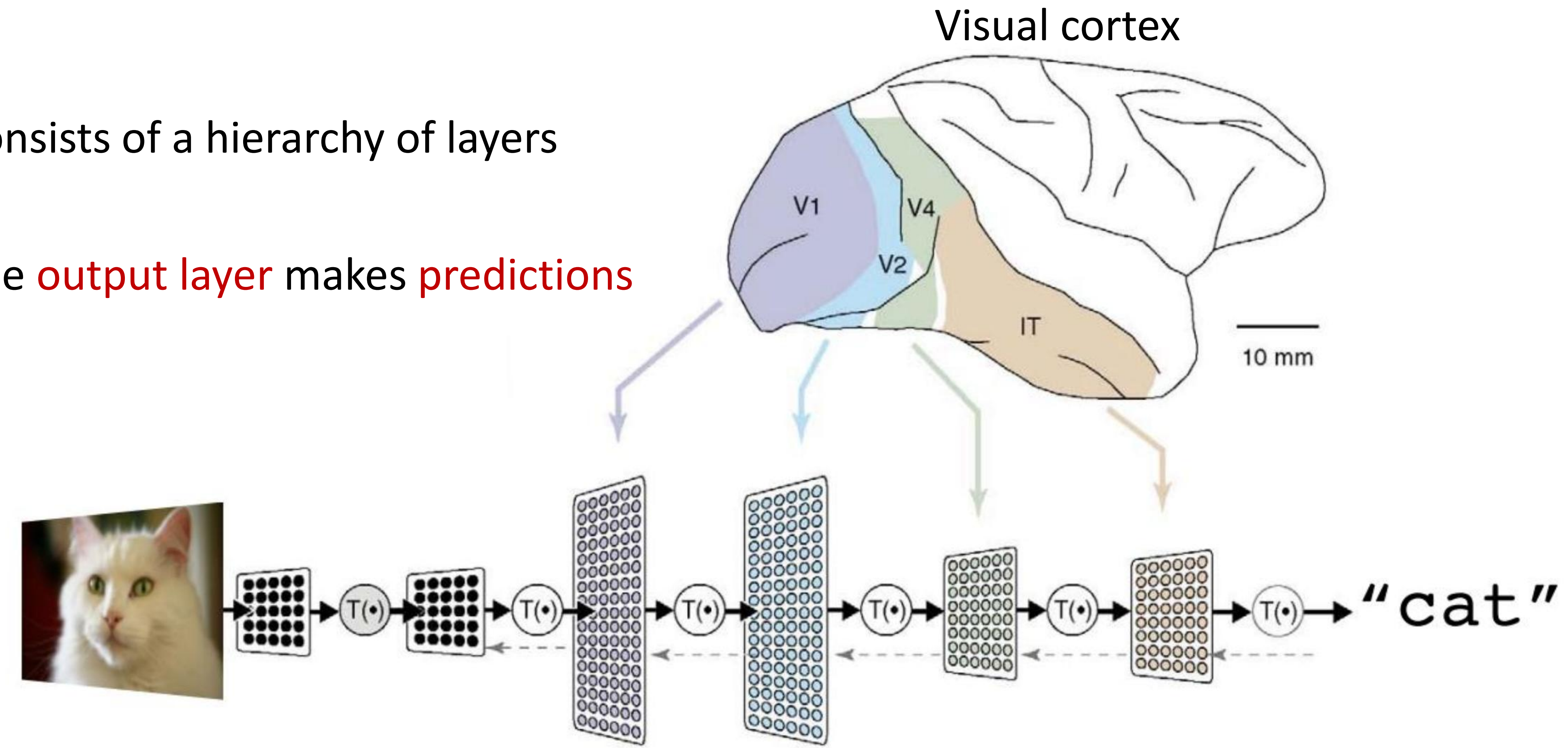
=> Convolutional Nueral Networks (CNNs)

[Yann Lecun, 1998]

Reference [4]

CNNs Architecture

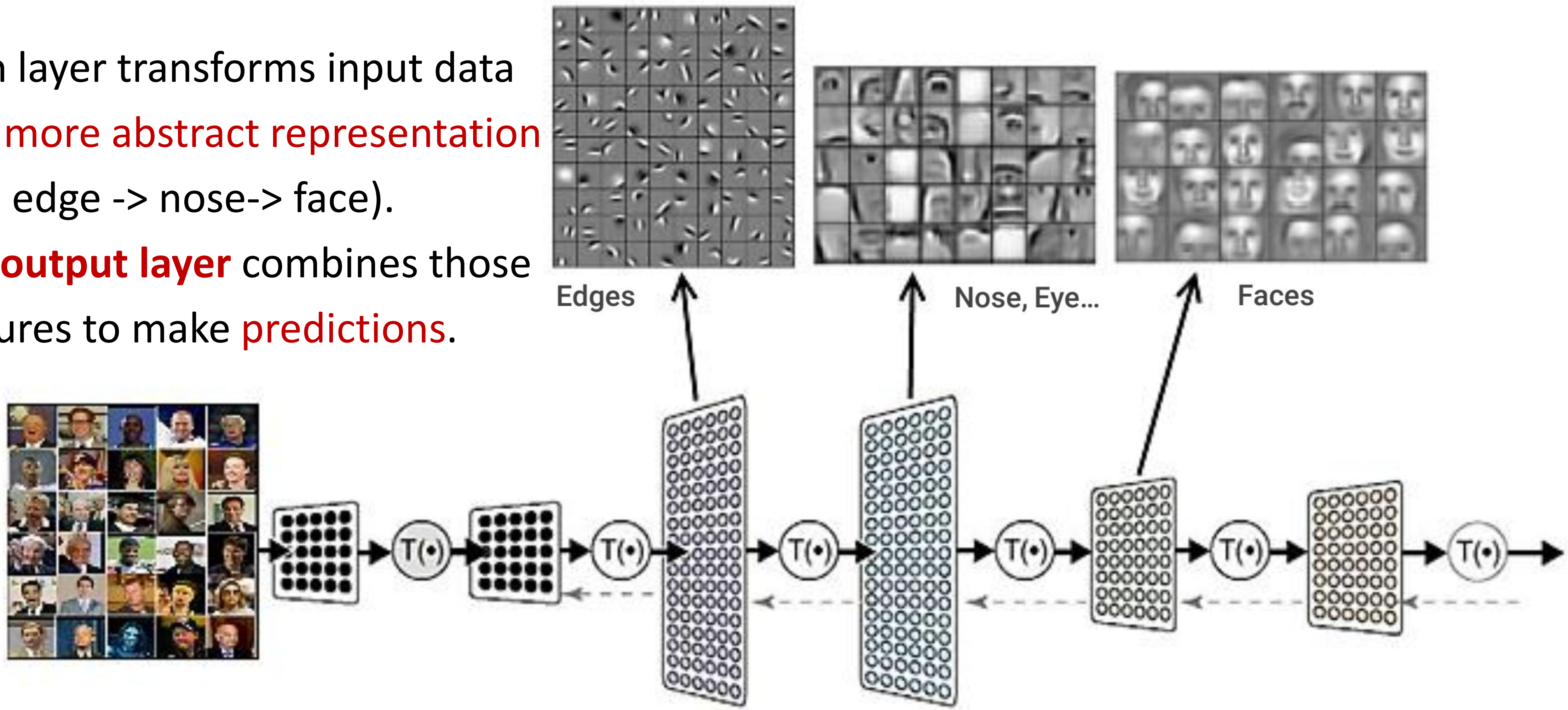
- Consists of a hierarchy of layers
- The output layer makes predictions



Reference [4]

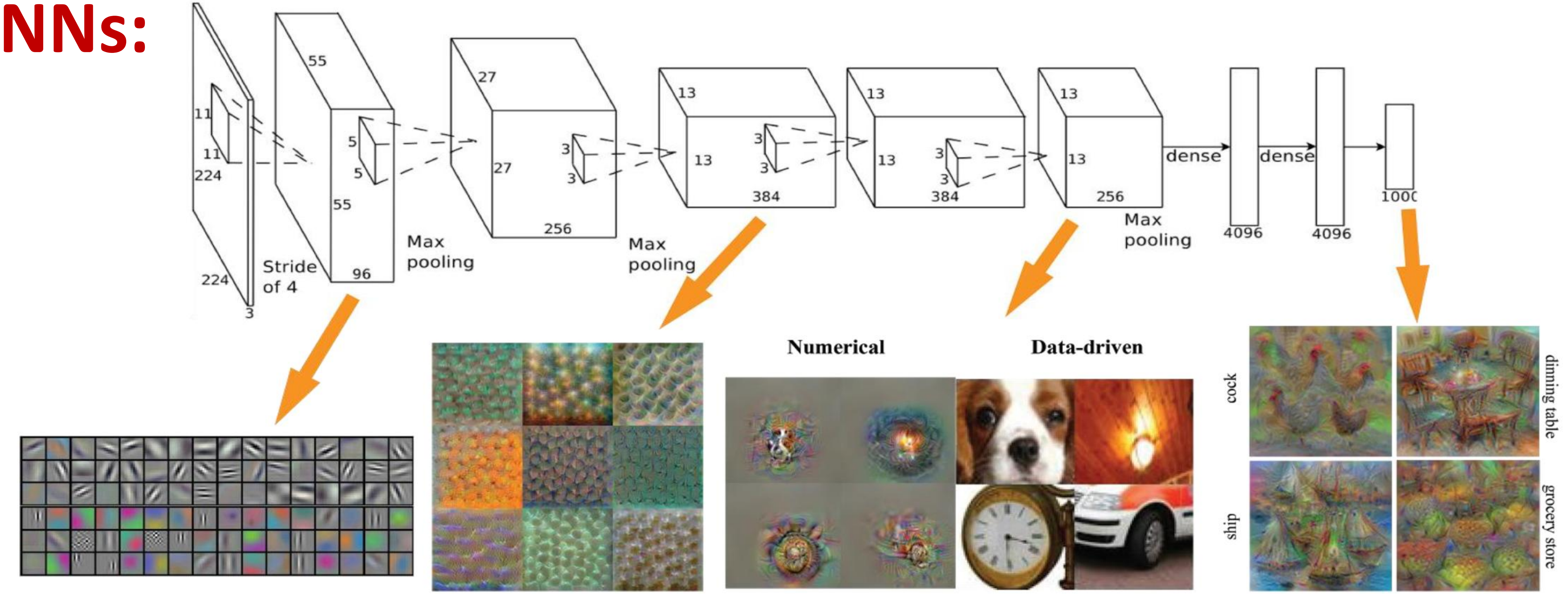
CNNs Architecture

- Each layer transforms input data into more abstract representation (e.g. edge -> nose-> face).
- The **output layer** combines those features to make predictions.



Reference [4]

CNNs:



Conv 1: Edge+Blob

Conv 3: Texture

Conv 5: Object Parts

Fc8: Object Classes

Convolutional Layer

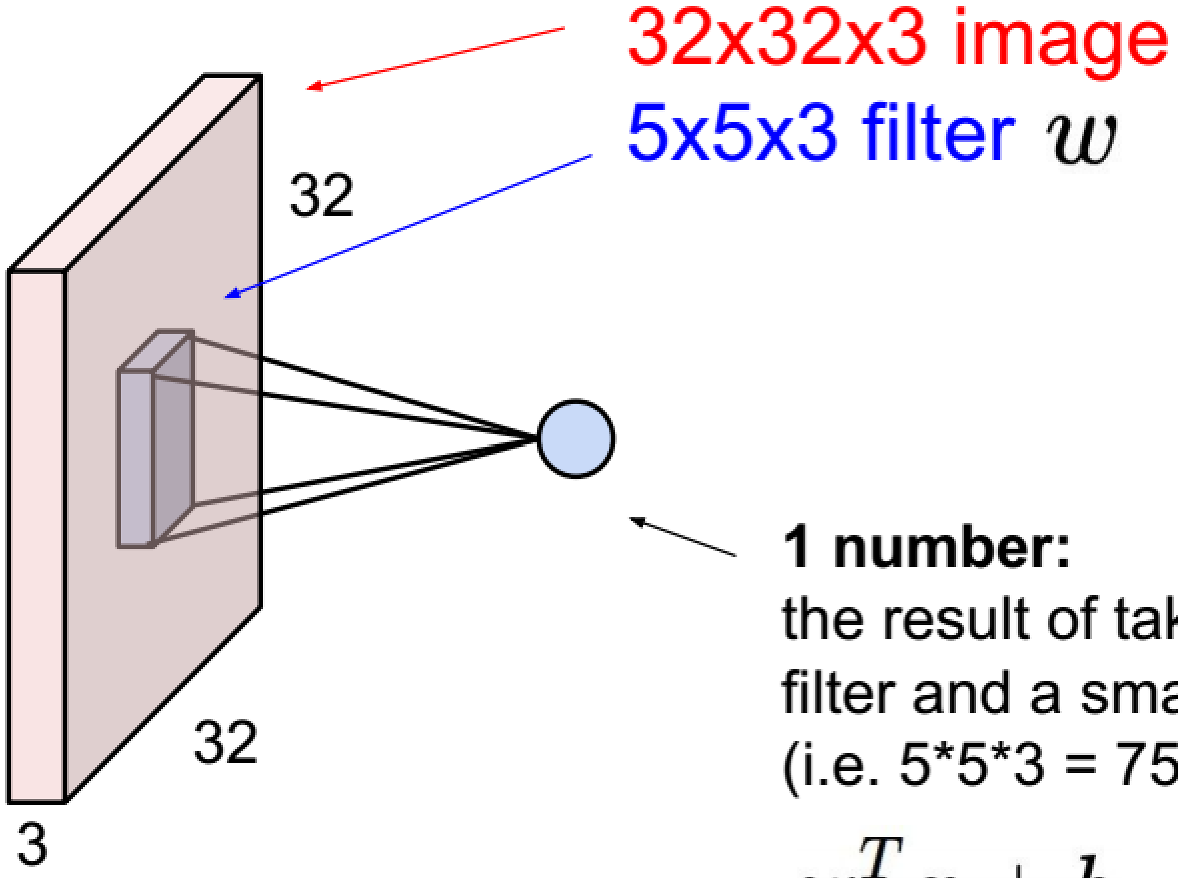
- Is a **feature detector**
- Learn to **filter out** not needed info using **kernels**

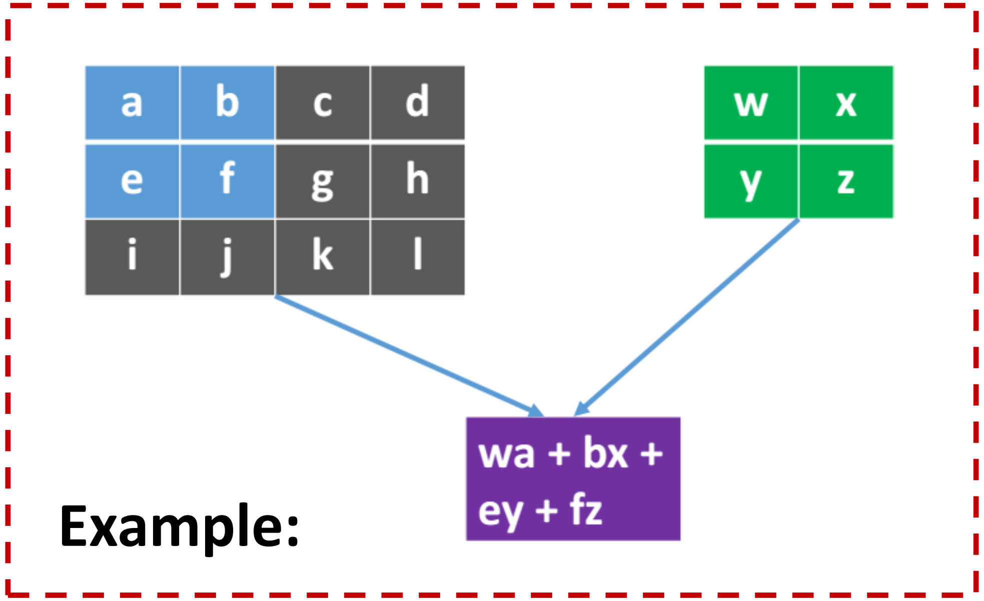
Pooling Layers

- compute max or average value of a particular feature over a region
- Downsizing input images

Reference [4]

Convolutional Layer



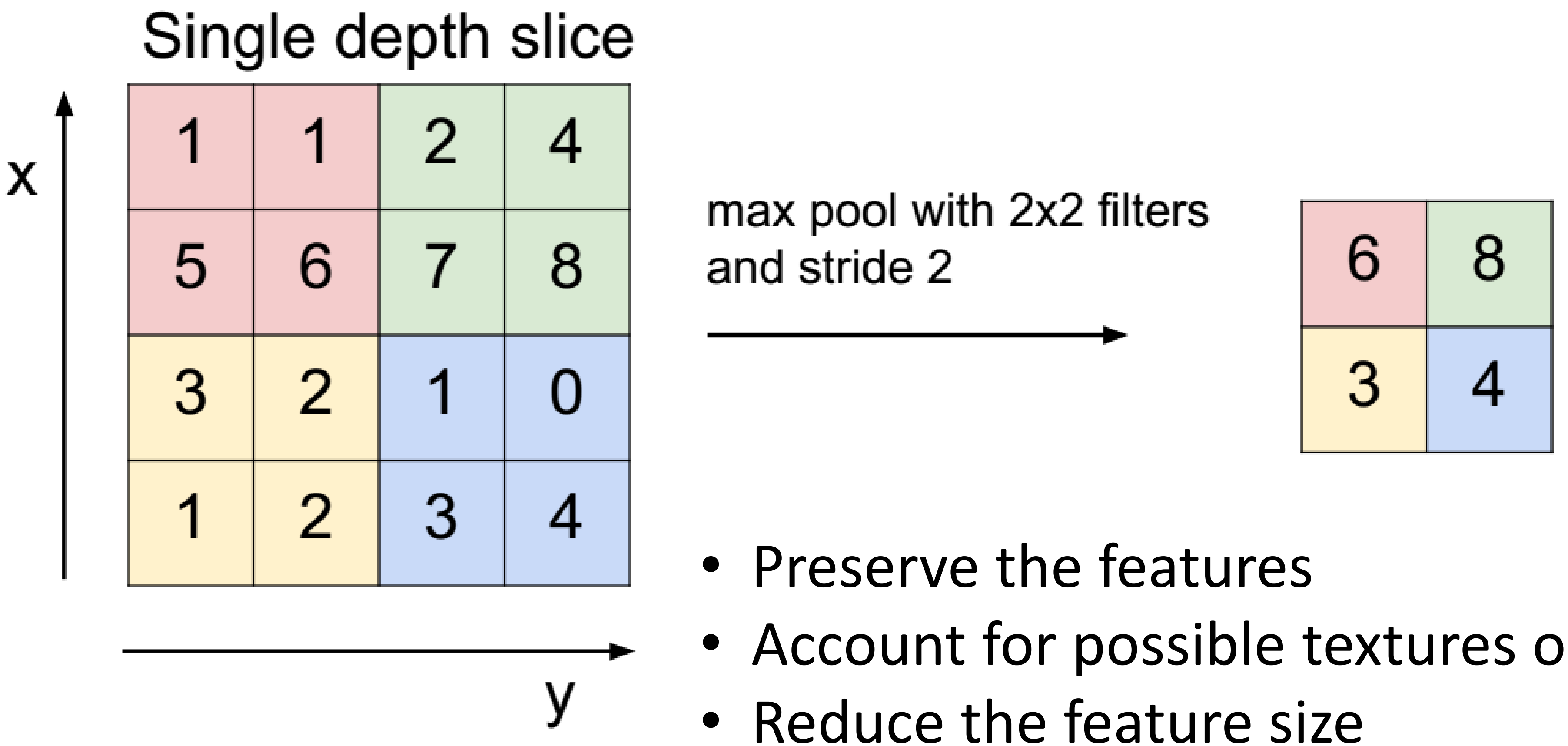


1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image (i.e. 5*5*3 = 75-dimensional dot product + bias)

$$w^T x + b$$

Reference [5]

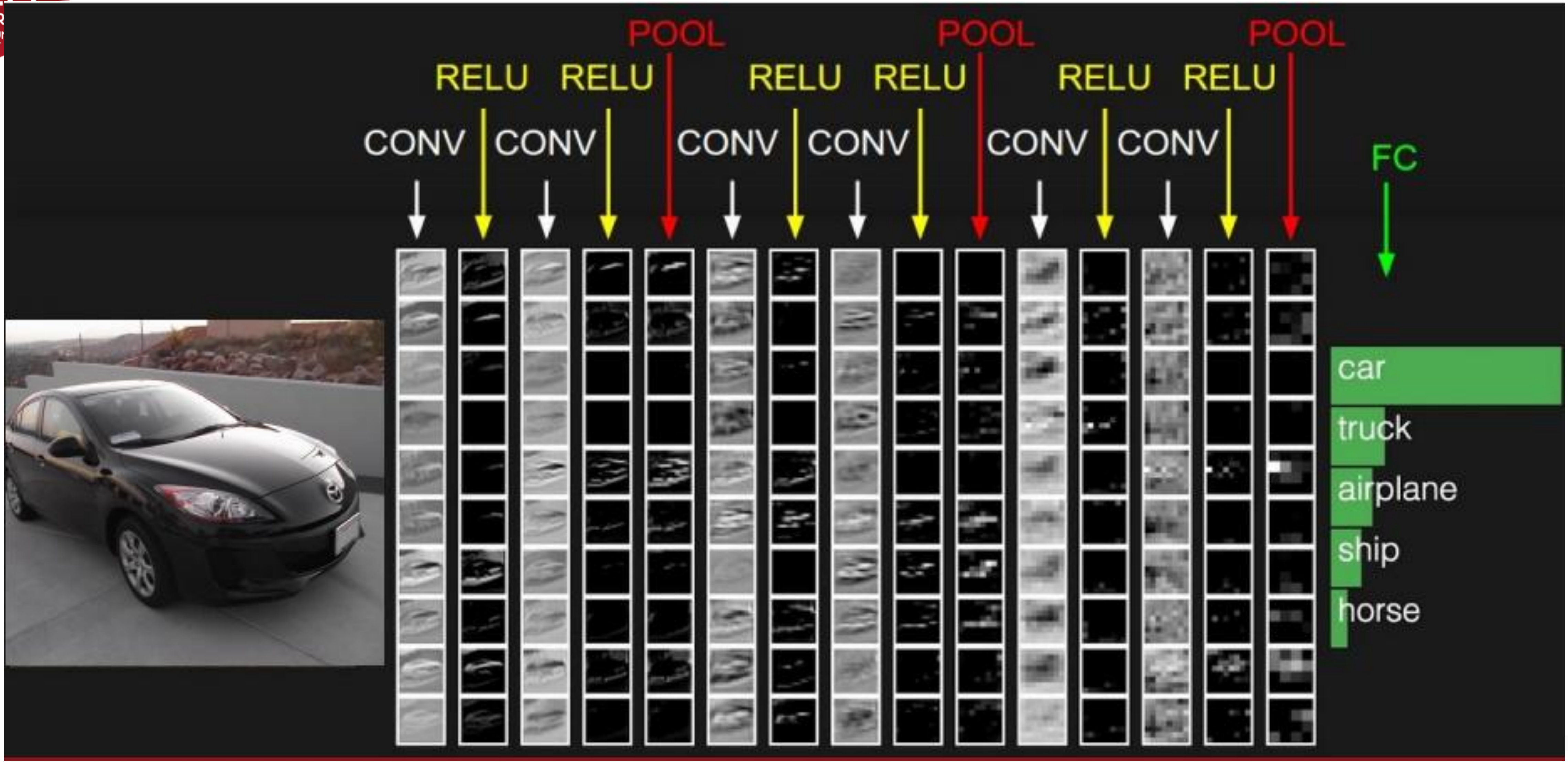
Pooling Layer



- Preserve the features
- Account for possible textures or distortions
- Reduce the feature size
- Prevent overfitting

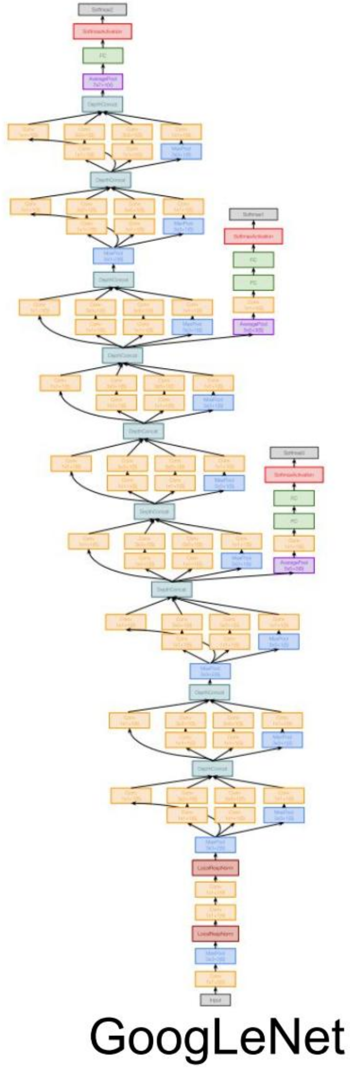
Reference [5]

Typical CNNs Architecture:



Reference [5]

CNNs Architecture:



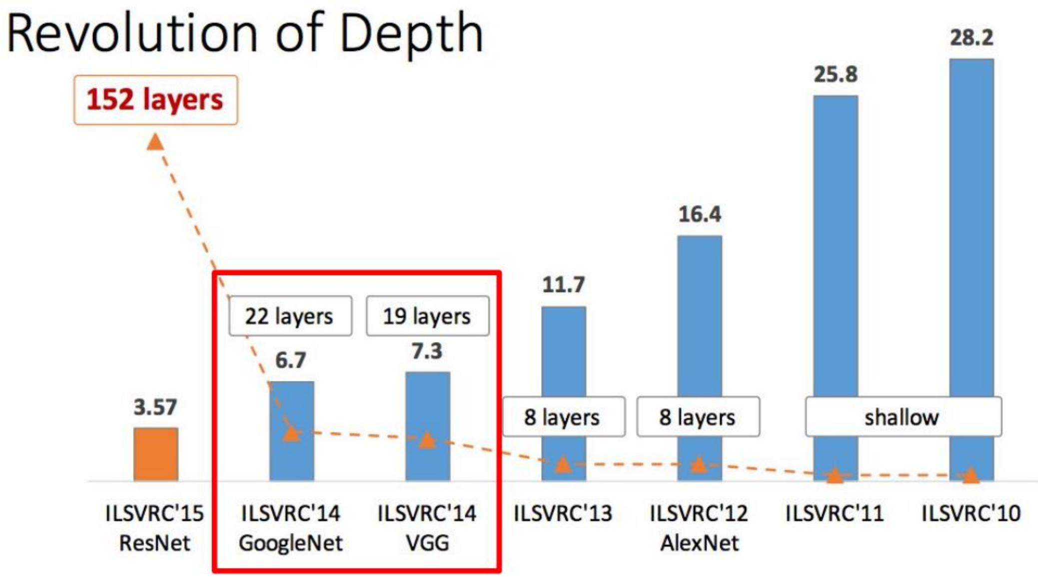


Figure copyright Kaiming He, 2016. Reproduced with permission.

Reference [5]

02

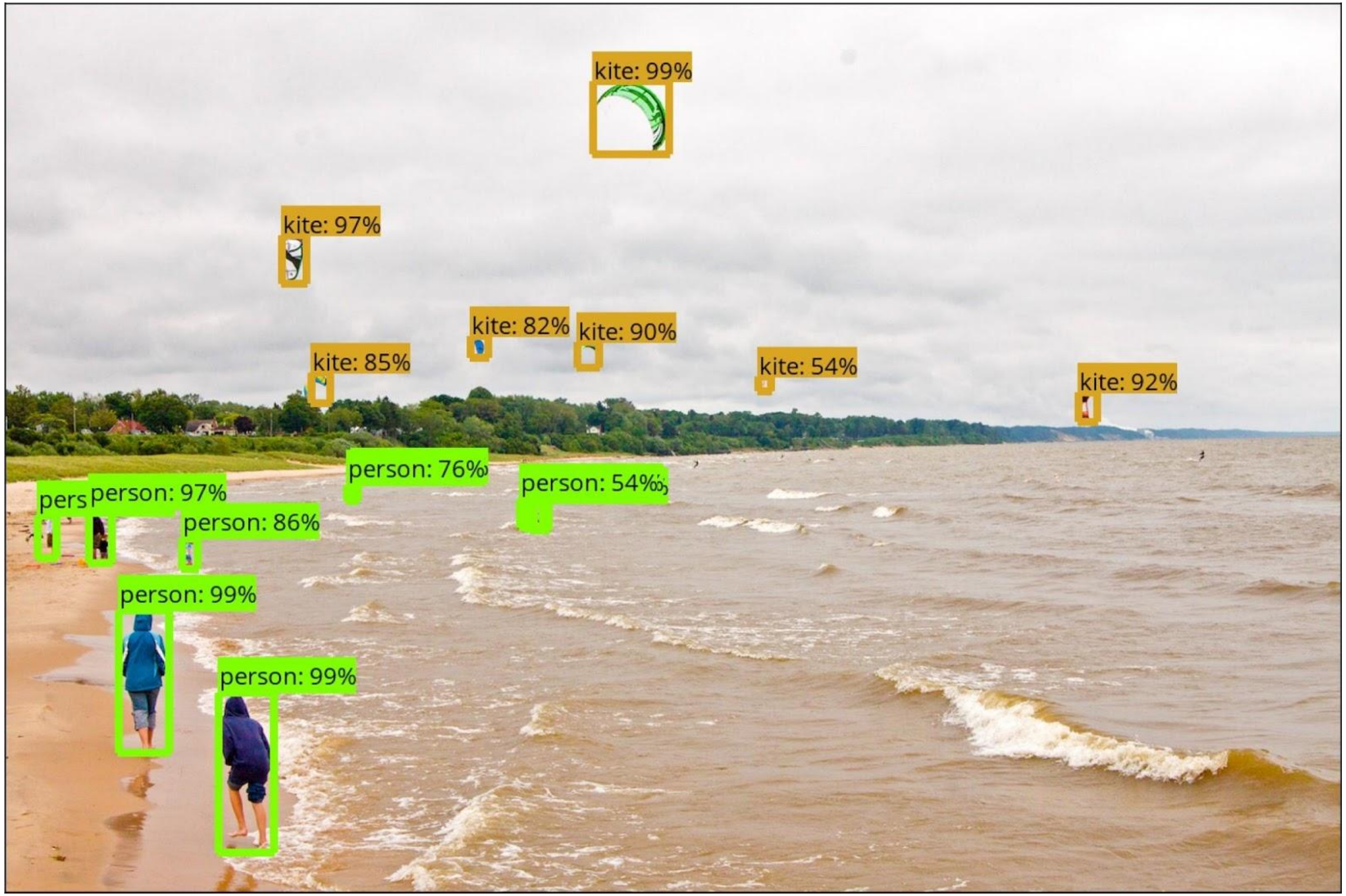
APPLICATIONS OF DEEP LEARNING

*COMPUTER VISION

Object Classification

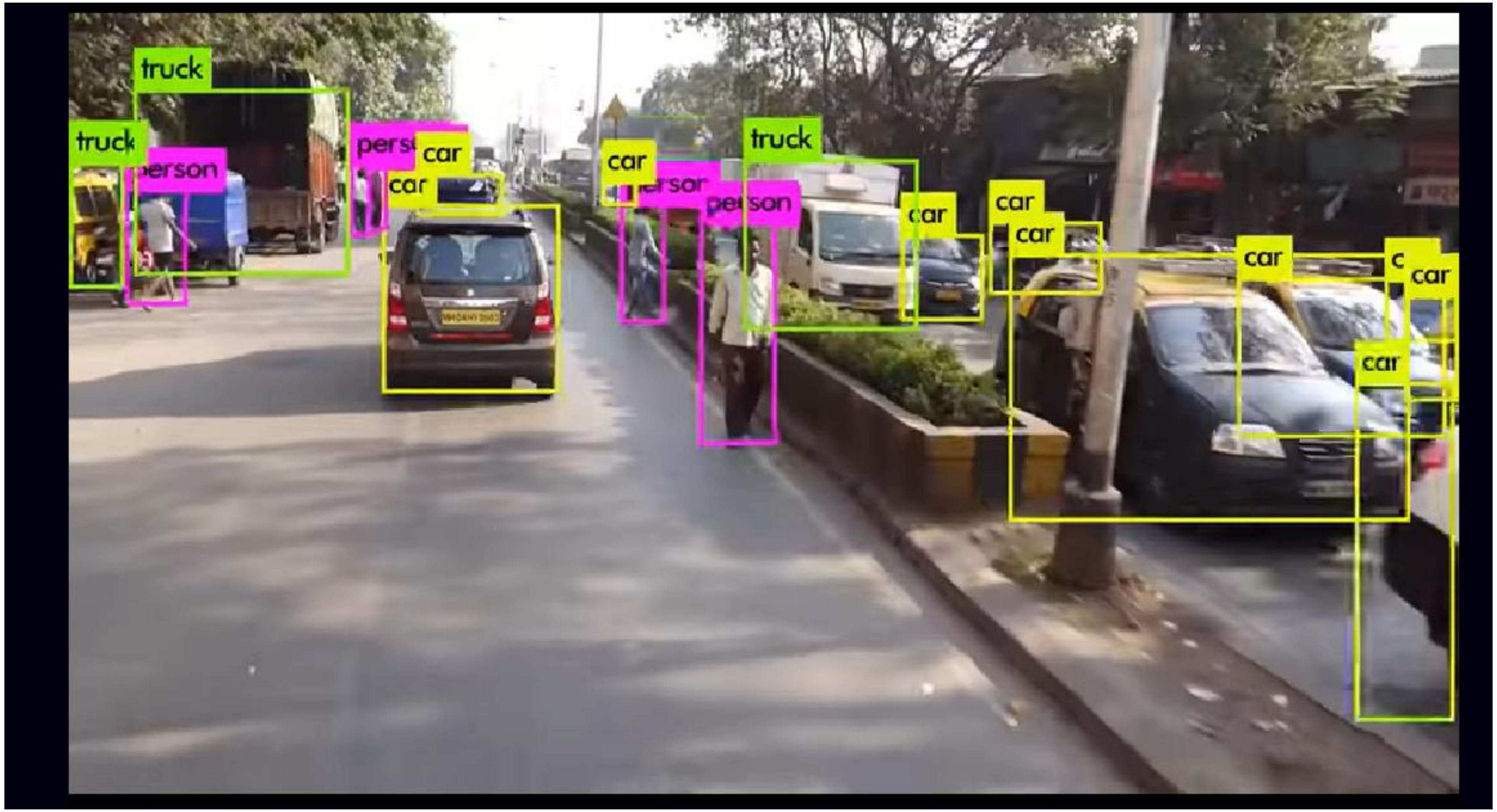
Image Retrieval

TensorFlow Object Detection API [2016]



Detected objects in a sample image (from the COCO dataset) made by one of our models. Image credit: Michael Miley, original image

YOLOv3: Real-time Object Detection [2018]



YOLOv3-608

- **mAP:** **57.9**
- **FPS:** **20**

Reference [7]

Image Segmentation

[Farabet et al., 2012]

Mask R-CNN: Object Detection & Segmentation

- **Facebook research**
- **2017**

Reference [8]

Mask R-CNN:
pose estimation & instance segmentation

Reference [9]

My Research Topics:

Alcohol Brand Logos Classification

ICDAMT 2018 conference

- **CNNs**
- **4 Thai alcohol brands vs Non-alcohol**
- **Accuracy: 89.16%**

My Research Topics:

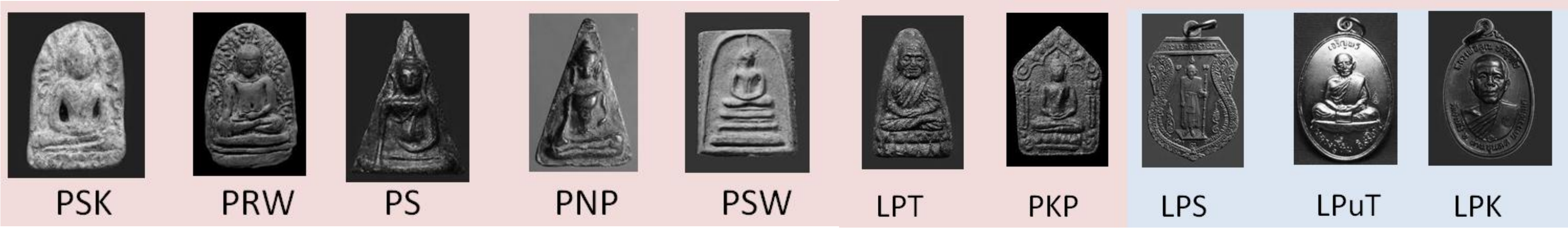
Facial Expressions Recognition

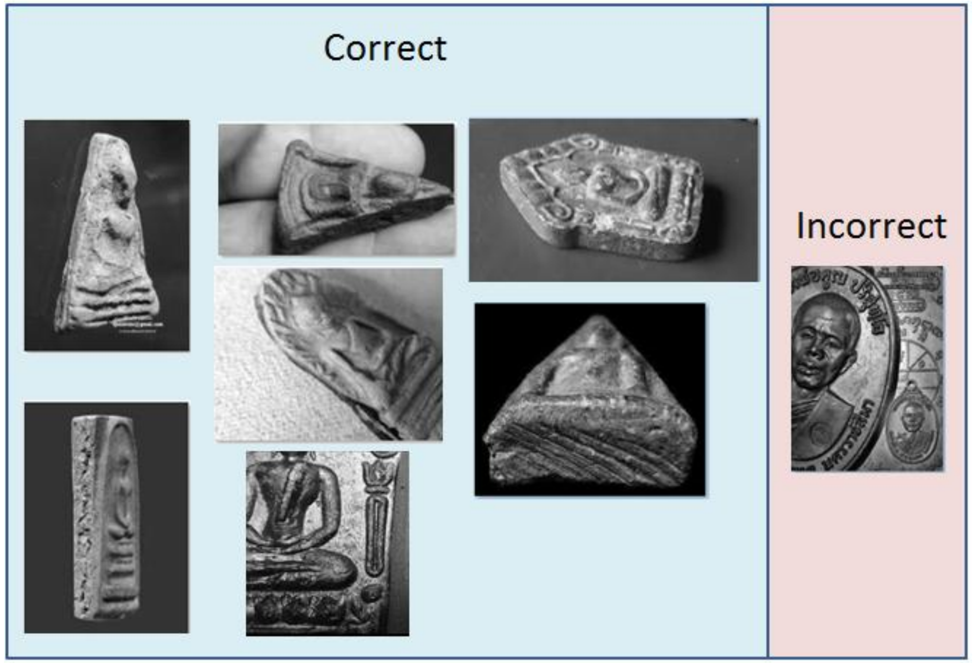
JCSSE 2018 conference

- XCEPTION
- 7 facial expressions
- Accuracy
  - Our model: 71.69%
  - Human: 65-70%

My Research Topics:

Buddha Amulets Classification





Submitted to KSE 2018 conf

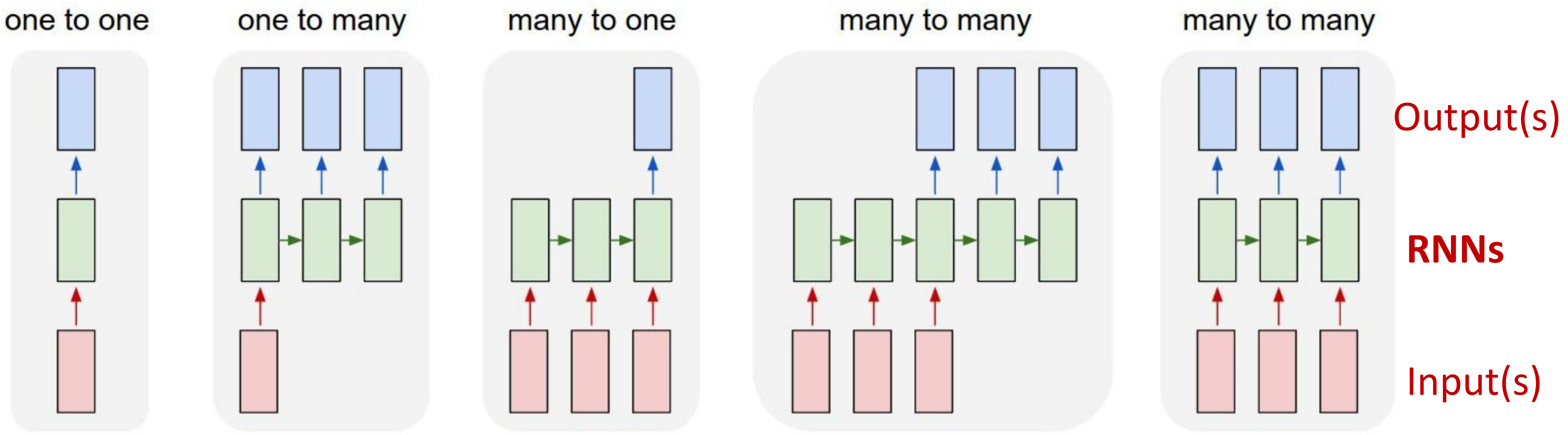
- **CNNs: 15 layers**
- **34.5 M parameters**
- **10 famous editions**
- **Accuracy: ~91%**

02

Applications of Deep Learning

*Sequence data, Time Series

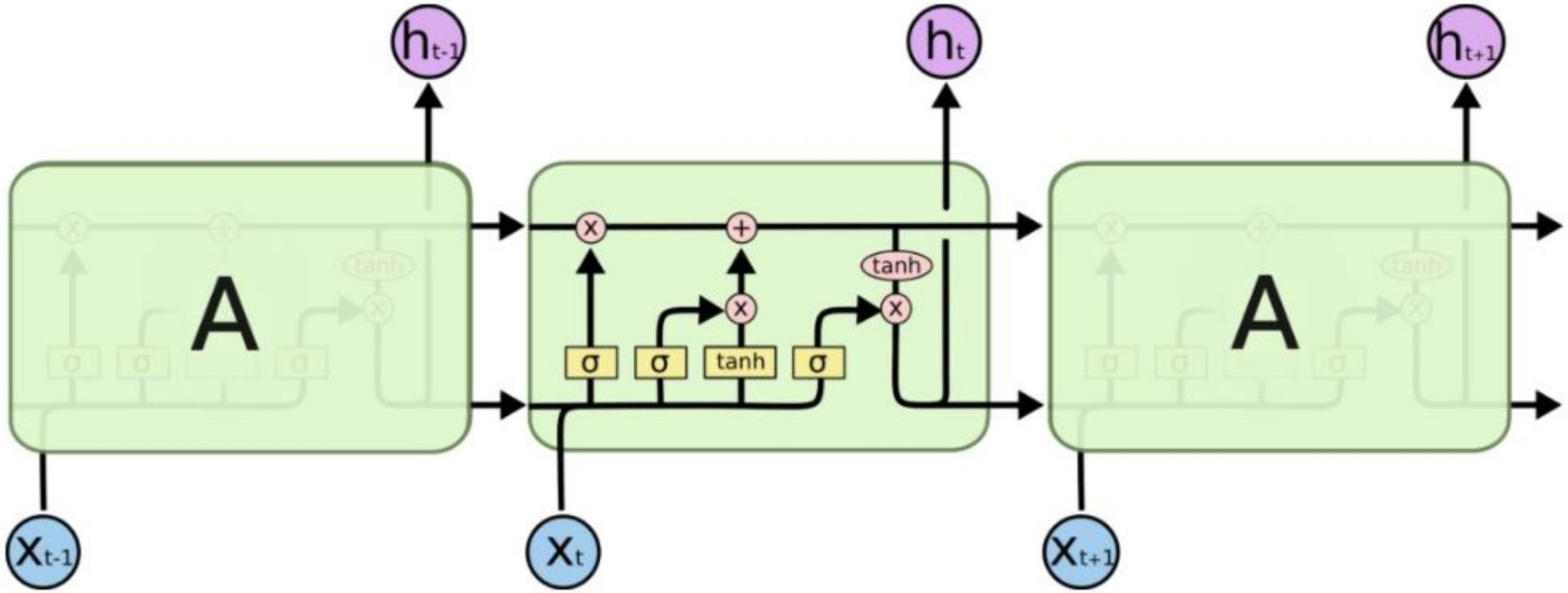
Recurrent Neural Networks (RNNs):



- Learn algorithms to **map input sequences to output sequences** (flexible-sized vectors).
- The **output** vector's content are **influenced** by the **entirely of inputs**.

Reference [10]

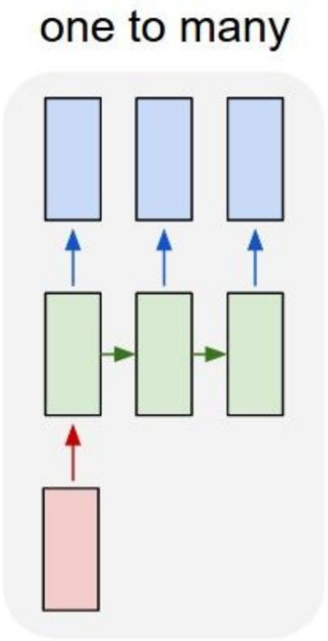
Long Short-Term Memory RNNs (LSTM)



- LSTM contains memory cells with read, write and reset operations.
- The network can learn
  - when it should remember data =>Long term
  - when it should throw it away (forget) =>Short term

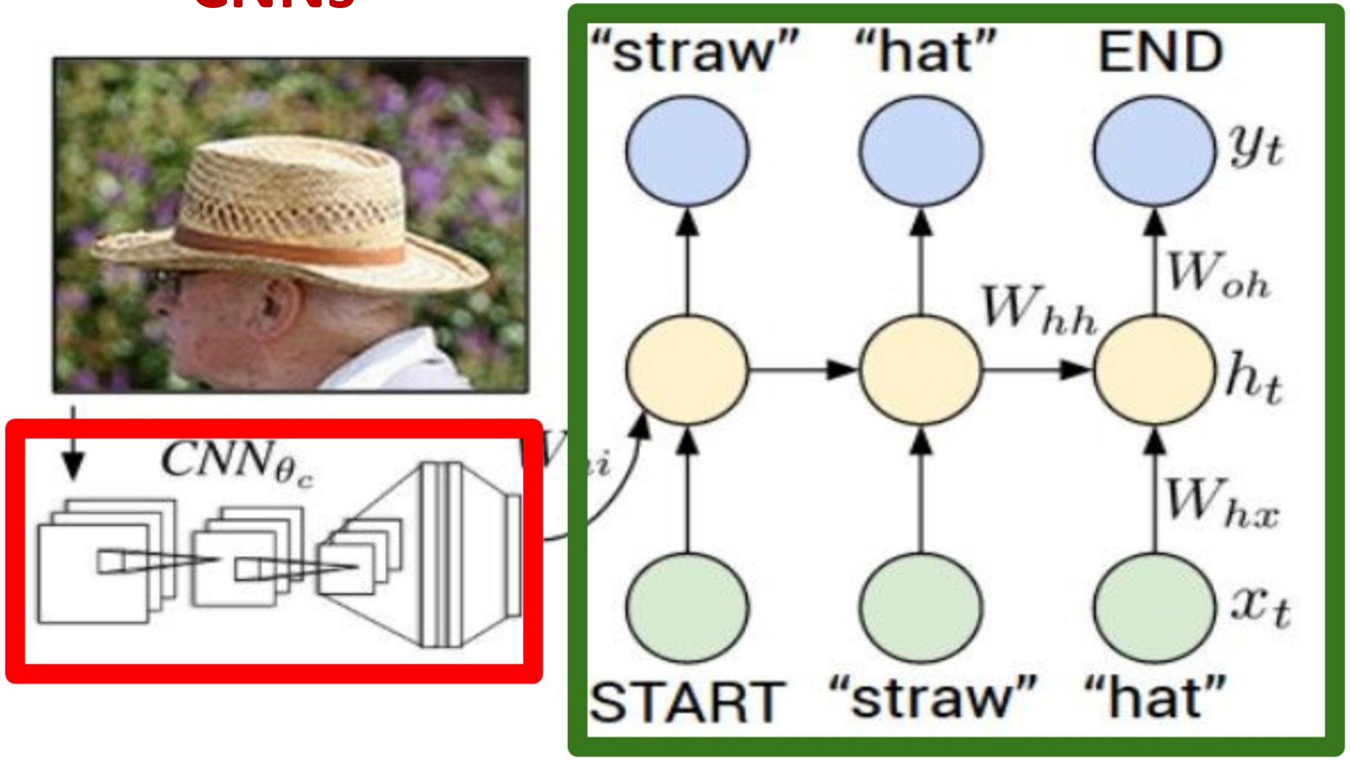
Reference [11]

Image Captioning:



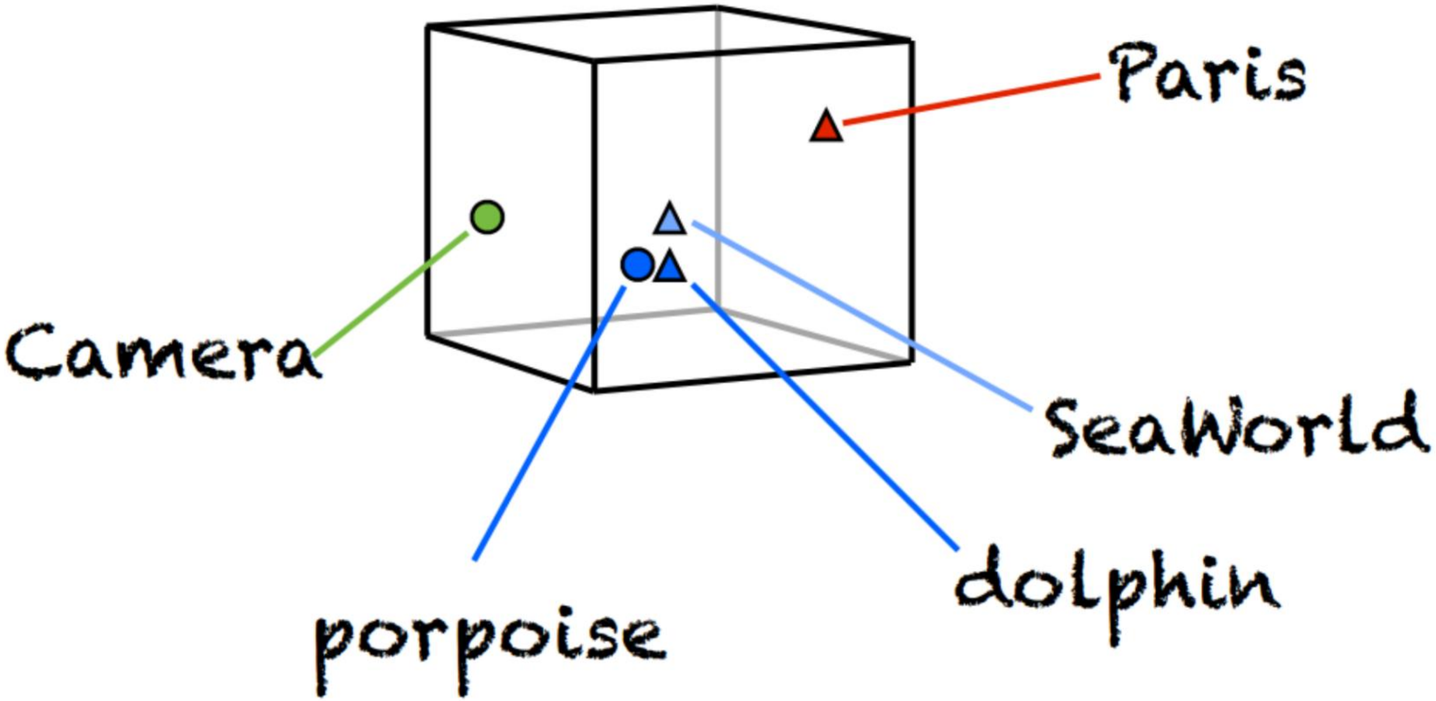
CNNs

RNNs



Reference [11]

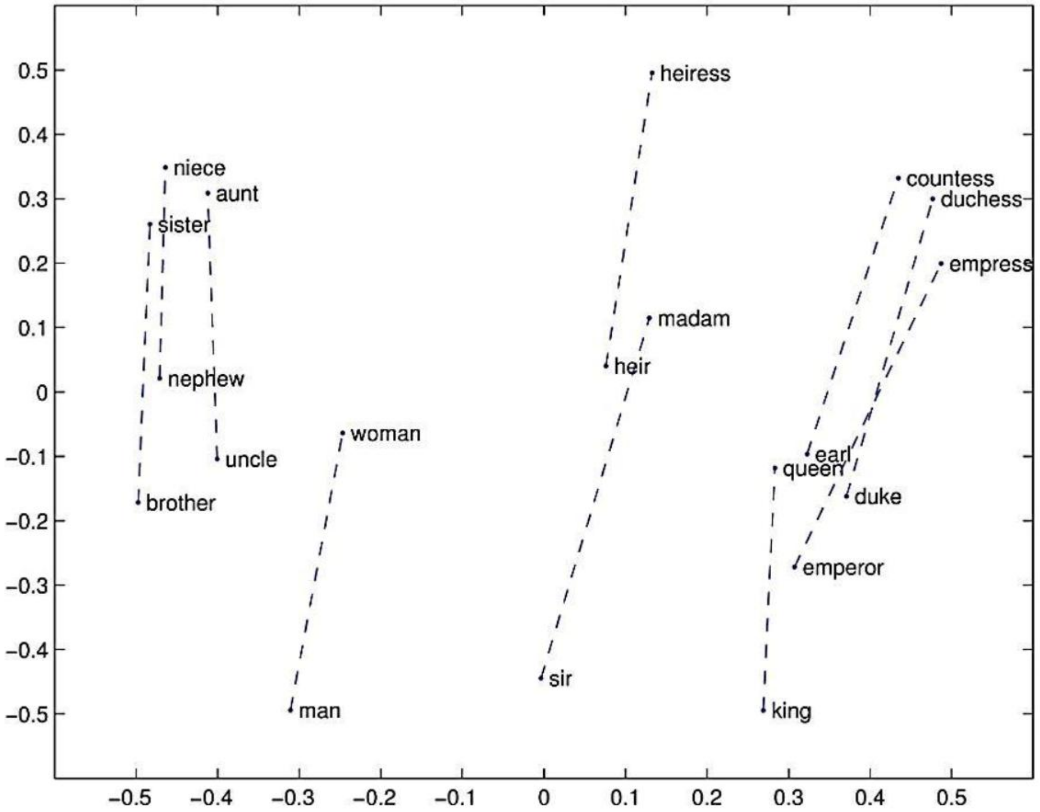
Natural Language Processing- Embeddings

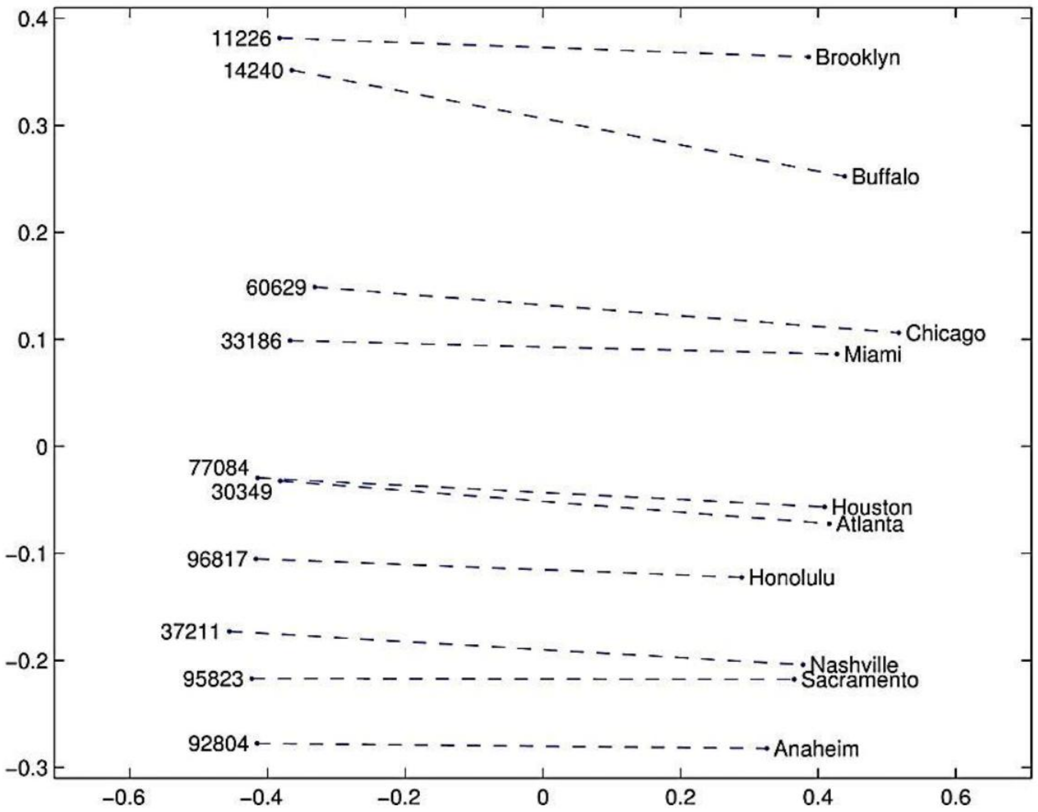


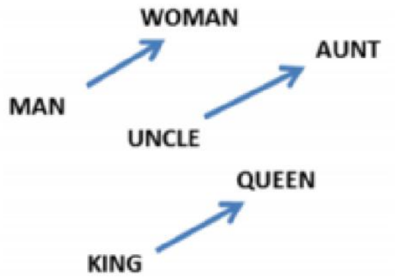
- Turn **textual data (words, sentences, paragraphs)** into high dimension **vector** representation
- Can **group** them together with semantically data in **vectorspace**

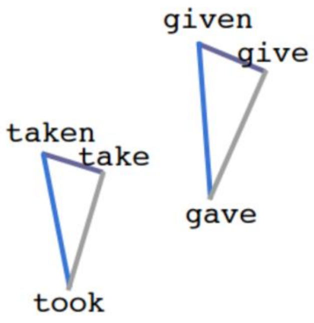
Reference [4]

Word2Vec









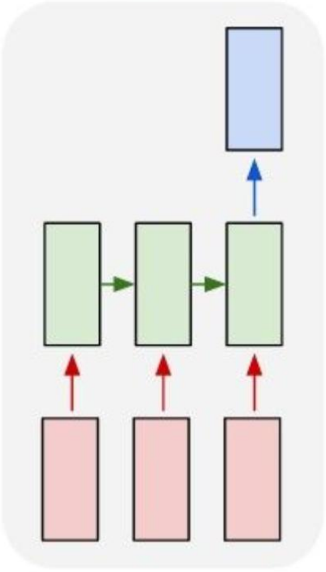
Woman – Man ≈ Aunt - *Uncle*
King - Male + Female ≈ *Queen*
Human - Animal ≈ *Ethics*

Reference [4]

College of Innovative Technology and Engineering

Sentiment Analysis:

many to one



Don't fly with @British_Airways. They can't keep track of your luggage.

:(

Following

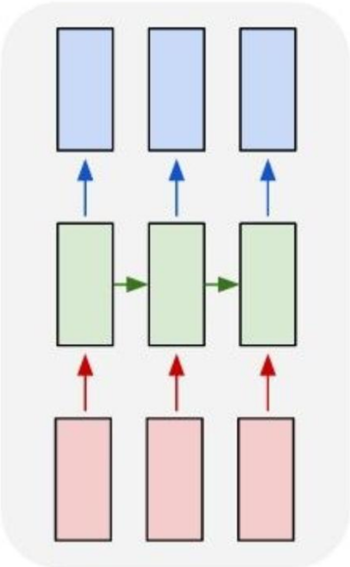
Happy Birthday to my best friend, the ♥of my life, my soul!!!! I love you beyond words! instagram.com/p/aTgfI-OS-a/

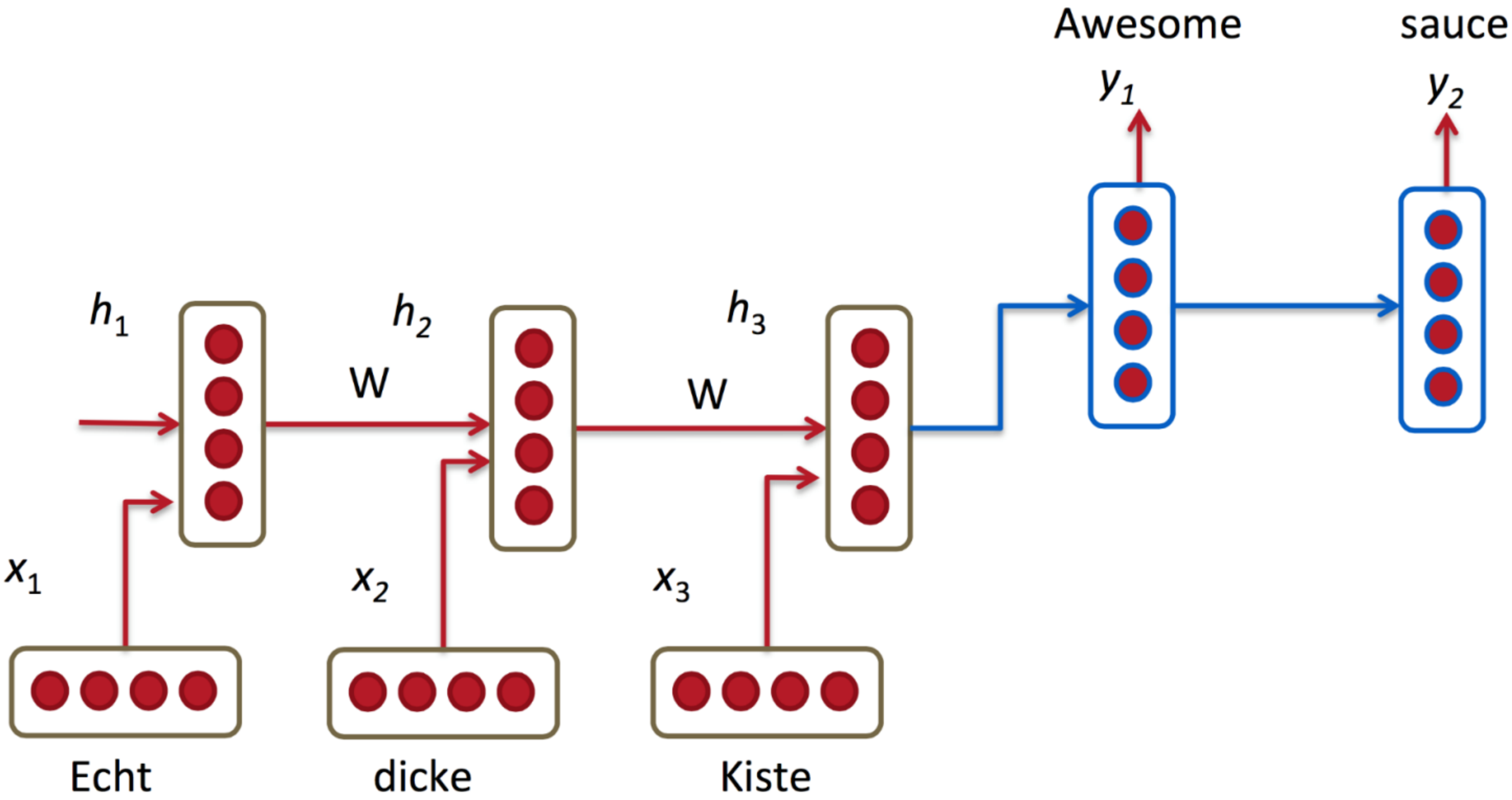
:)

Reference [11]

Machine Translation:

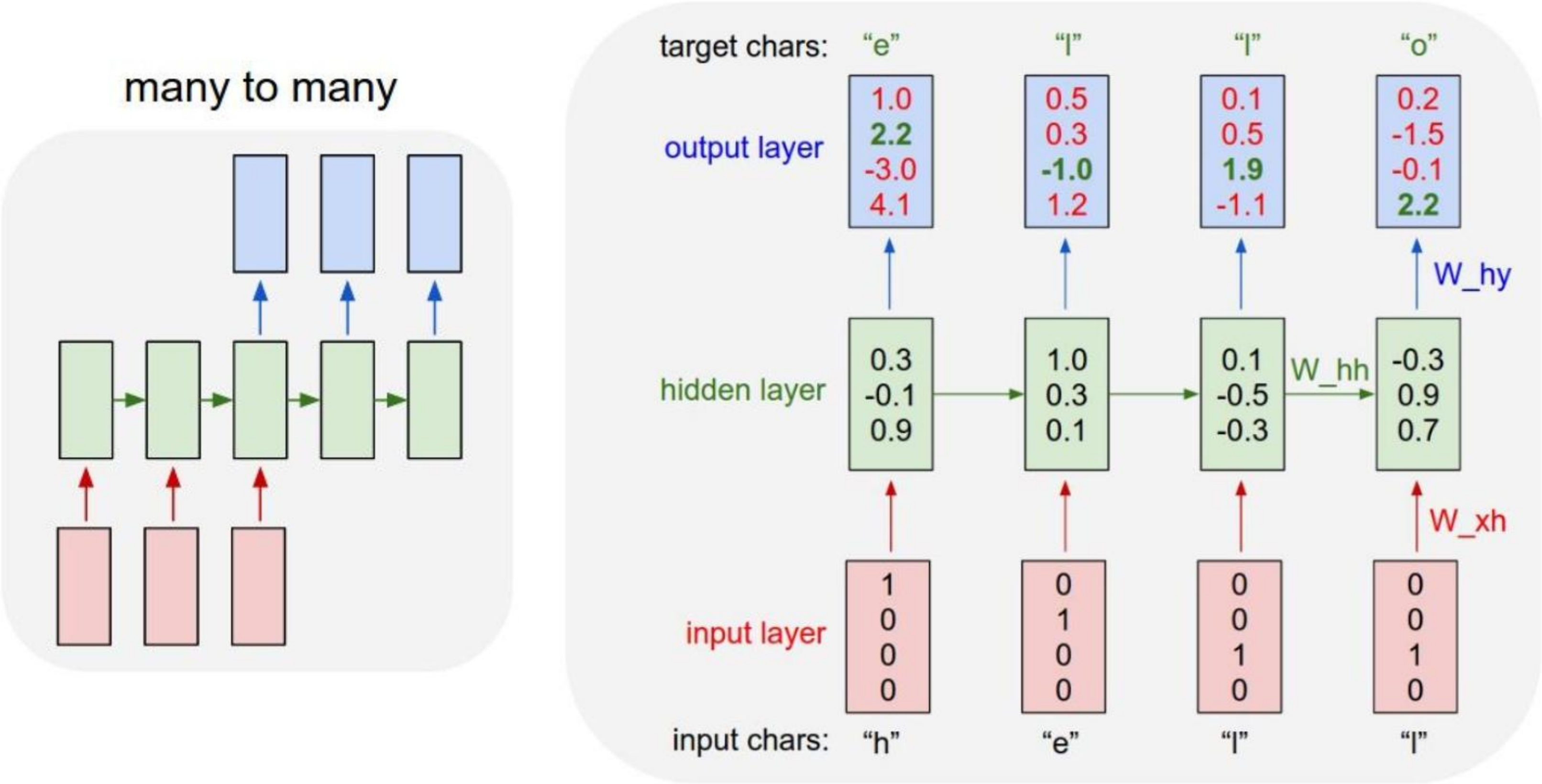
many to many





Reference [11]

Generating Text:



Life Is About The Weather!
Life Is About The True Love Of Mr. Mom
Life Is About Kids
Life Is About An Eating Story
Life Is About The Truth Now

The meaning of life is
literary recognition

The meaning of life is
the tradition of the ancient human reproduction

Andrej Karpathy. "The Unreasonable Effectiveness of Recurrent Neural Networks." (2015).

Reference [11]

Usage Requirements

- Large dataset with good quality (input-output mappings)
- Measurable and describable goals (define the cost)
- Enough computing power (AWS GPU Instance)
- Excels in tasks where the basic unit (pixel, word) has very little meaning in itself, but **the combination of such units has a useful meaning**

Reference [4]

03

GET STARTED WITH DEEP LEARNING

Step 0: Pre-requisites

- **Basics of Math**

  Resource : “Math | Khan academy”

  Especially Calculus, Probability and Linear Algebra)
- **Basics of Python**

  Resource: “Intro to Computer Science”, edX course)
- **Basics of Statistics**

  Resource: “Intro to Stats”, Udacity course)
- **Basics of Machine Learning**

  Resource: “Intro to Machine Learning”, Udacity course

Reference [12]

Step 1: Setup Google CoLab

- **Google's free cloud service for AI developers**
  - improve your **Python** programming language coding skills
- Develop deep learning applications on the **GPU for free**
  - using popular libraries such as **Keras**, **TensorFlow**, **PyTorch,** and **OpenCV**
- Google CoLab Free GPU **Tutorial**:
  https://medium.com/deep-learning-turkey/google-colab-free-gpu-tutorial-e113627b9f5d

Reference [12]

Step 2: Basic Deep Learning

- **CS231n: Convolutional Neural Networks for Visual Recognition**
  [http://cs231n.stanford.edu/2017/syllabus.html]
  - Introduction to Neural Networks
  - Loss Functions and Optimization
  - CNNs, RNNs, LSTM

- **Popular Libraries:**
  - **TensorFlow (using Keras => Recommended)**
  - Caffe
  - Torch

Reference [12]

Step 3: Advanced Deep Learning

- **Deep Learning for Computer Vision**
  Primer : "DL for Computer Vision" blog.
  Project : "Facial Keypoint Detection" Tutorial
  Required libraries : Nolearn
  Associated Course : "CS231n: Convolutional Neural Networks for Visual Recognition"

- **Deep Learning for Natural Language**
  ProcessingPrimer : "Deep Learning, NLP, and Representations" blog.
  Project : "Deep Learning for Chatbots": "Part 1", "Part 2"
  Required library : Tensorflow
  Associated Course : "CS224d: Deep Learning for Natural Language Processing"

Reference [12]

Step 4: Setup your own Machine (optional)

- A good enough **GPU** (4+ GB), preferably **Nvidia**
- An **OK CPU** (e.g. Intel Core i3 is ok, Intel Pentium may not be OK)
- **16 GB of RAM** or depending upon the dataset.
- **Power supply** (+100 to 120 w)

Reference [12]

Which GPU(s) to Get?

Best GPU overall (by a small margin): Titan Xp

Cost efficient but expensive: GTX 1080 Ti, GTX 1070, GTX 1080

Cost efficient and cheap: GTX 1060 (6GB)

I work with data sets > 250GB: GTX Titan X (Maxwell), NVIDIA Titan X Pascal, or NVIDIA Titan Xp

I have little money: GTX 1060 (6GB)

I have almost no money: GTX 1050 Ti (4GB)

I do Kaggle: GTX 1060 (6GB) for any "normal" competition, or GTX 1080 Ti for "deep learning competitions"

I am a competitive computer vision researcher: NVIDIA Titan Xp; do not upgrade from existing Titan X (Pascal or Maxwell)

I am a researcher: GTX 1080 Ti. In some cases, like natural language processing, a GTX 1070 or GTX 1080 might also be a solid choice — check the memory requirements of your current models

I want to build a GPU cluster: This is really complicated, you can get some ideas here

I started deep learning and I am serious about it: Start with a GTX 1060 (6GB). Depending of what area you choose next (startup, Kaggle, research, applied deep learning) sell your GTX 1060 and buy something more appropriate

I want to try deep learning, but I am not serious about it: GTX 1050 Ti (4 or 2GB)

Reference [13]

Q & A

Thank You!

BigData ENGINEERING

References

[1] https://www.youtube.com/watch?time_continue=20283&v=qUAmTYHEyM8
[2] https://www.geospatialworld.net/news/toyota-to-invest-2-8-billion-in-self-driving-car-project/
[3] https://www.youtube.com/watch?v=wMUmPumXtpw
[4] https://www.slideshare.net/LuMa921/deep-learning-a-visual-introduction
[5] https://www.datacamp.com/courses/deep-learning-in-python
[6] http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture5.pdf
[7] https://pjreddie.com/darknet/yolo/
[8] https://github.com/karolmajek/Mask_RCNN
[9] https://www.youtube.com/watch?v=KYNDzlcQMWA
[10] http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture10.pdf
[11] https://selfdrivingcars.mit.edu/
[12] https://www.analyticsvidhya.com/blog/2016/08/deep-learning-path/

http://timdettmers.com/2017/04/09/which-gpu-for-deep-learning/