Computational Intelligence

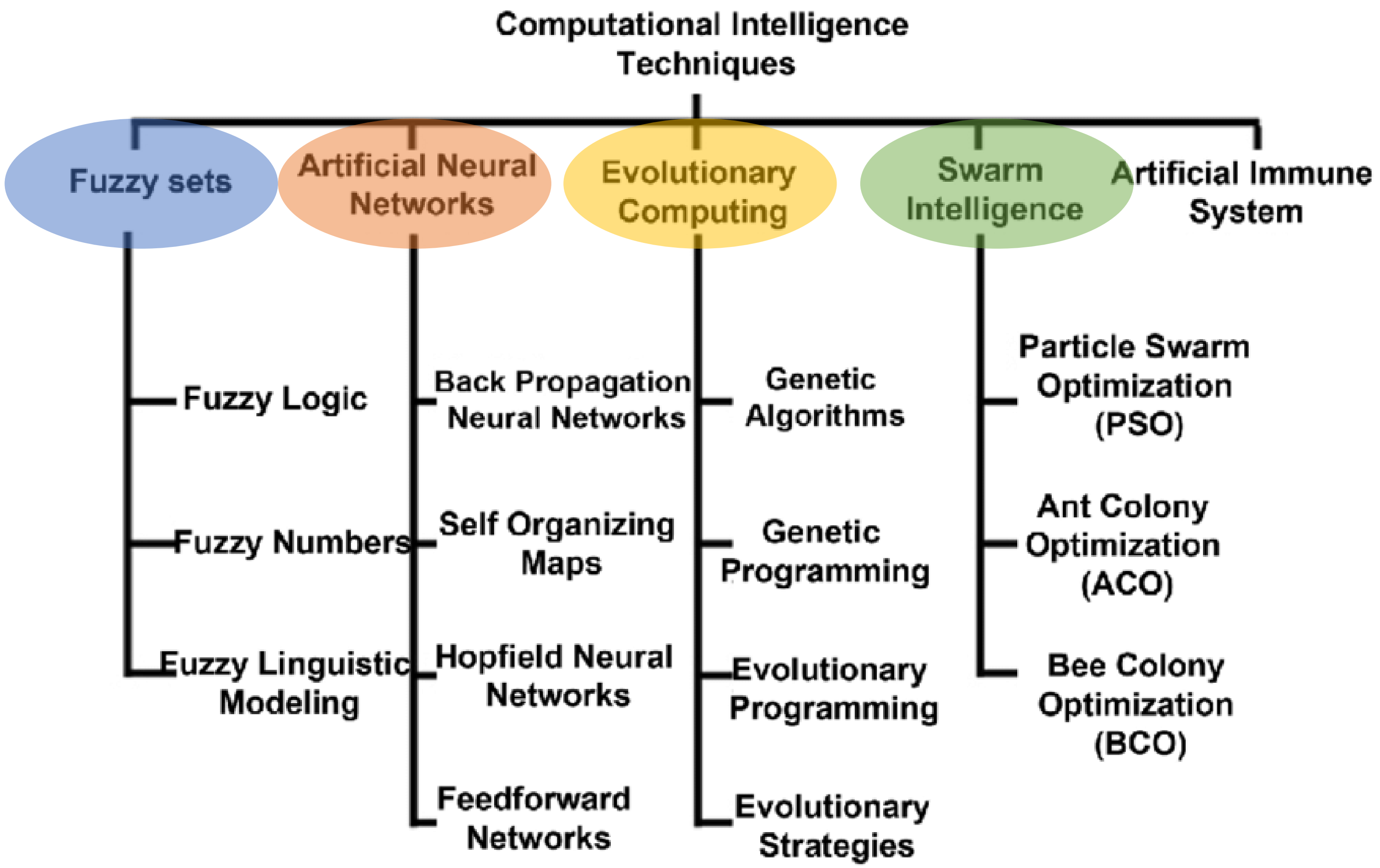
Samaneh Hosseini

Isfahan University of Technology

Outline

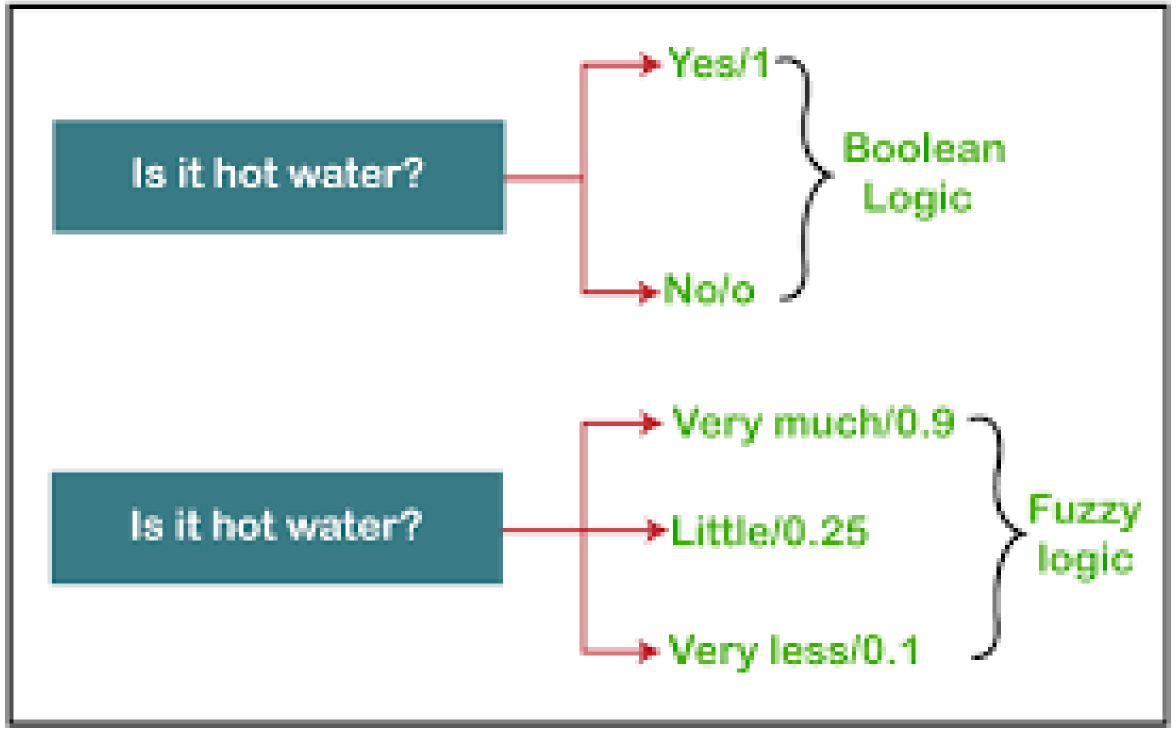
- What is Computational Intelligence?
- Computational Intelligence Applications
- Why Computational Intelligence and Why now?
- Course administrations

What is Computational Intelligence?



Fuzzy logic introduction

- The term fuzzy refers to things that are not clear or are vague

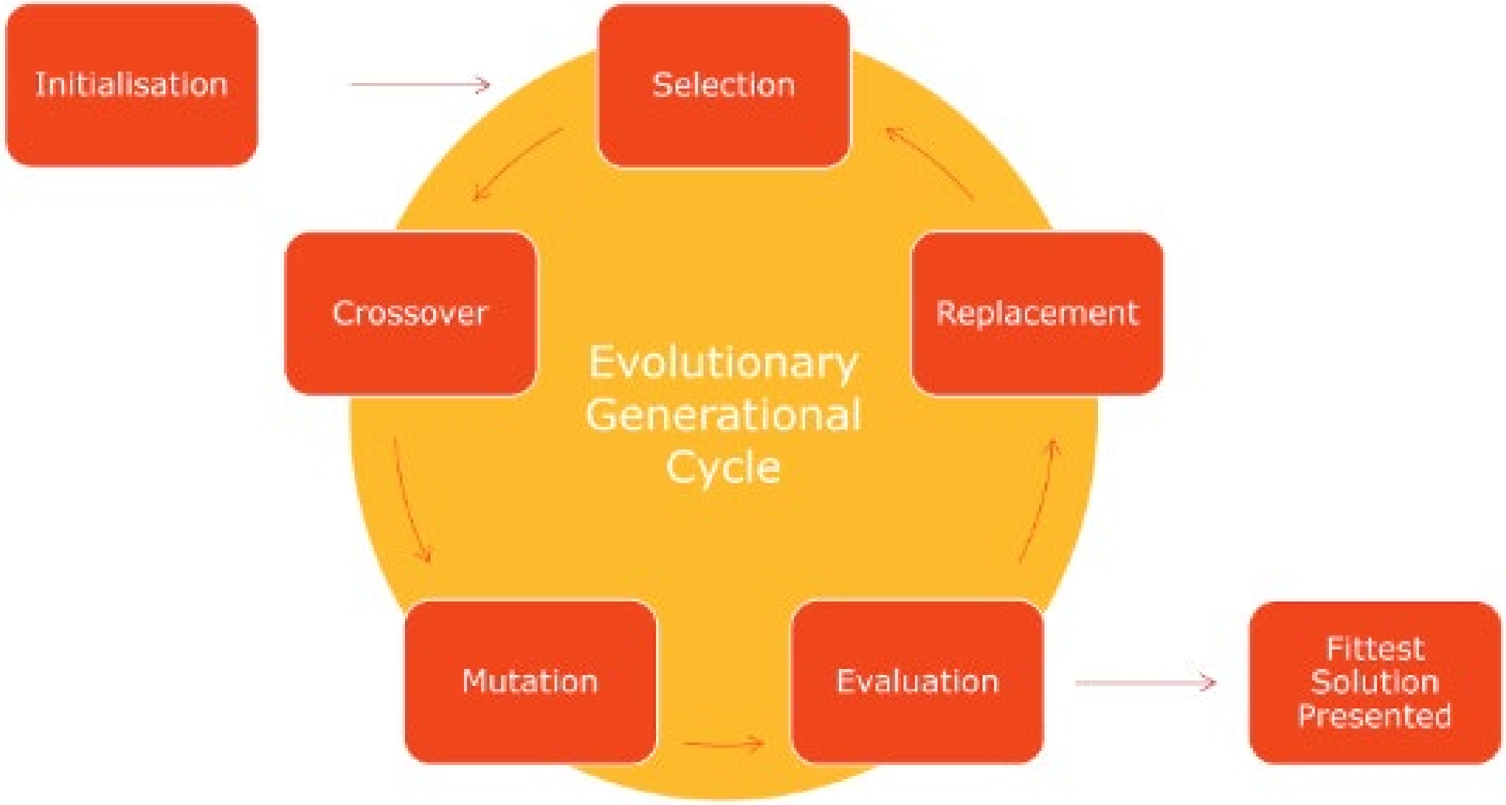


Evolutionary Computing

- simulate physical and/or biological behavior in nature to solve optimization problems.
- Try to solve problems that:
  1. Very difficult to model mathematically.
  2. Computationally expensive to solve.
  3. Involves a large number of parameters.

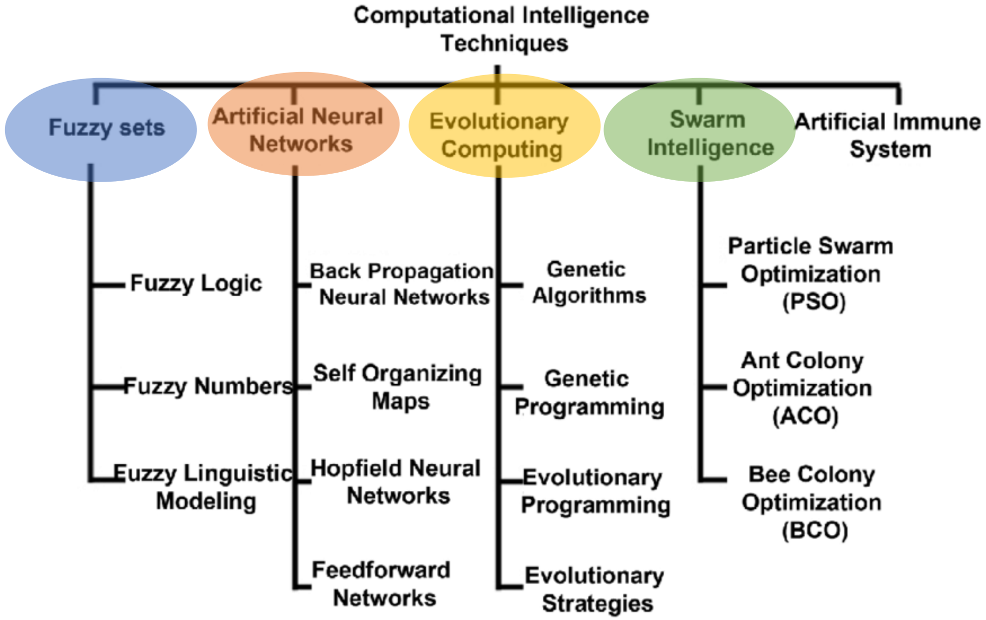
Genetic Algorithms

- a subset of evolutionary algorithms
- simulates Genetics and Evolution (biological behavior)



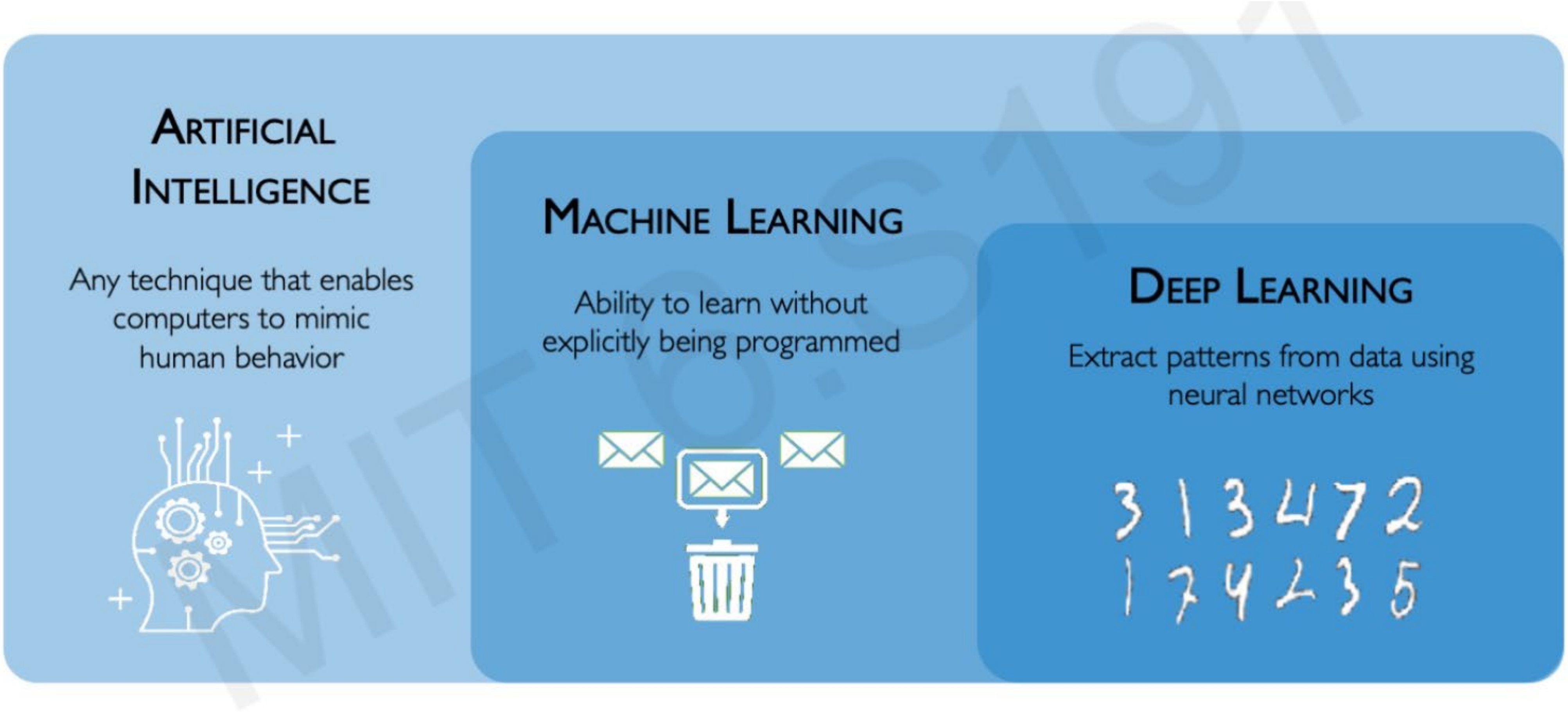
Swarm intelligence

What is Computational Intelligence?



L1-Introduction
L2-Neuron math model
L3-Perceptron
L4-Building and Applying NN
L5-Gradient descent
L6-Vectorization
L7-Overfitting
L8-Regularization I
L9-Regularization II
L10-Optimization Algorithms I (mini-batches)
L11-Optimization Algorithms II(exponentially weighted averages)
L12-Hyperparameter Tuning
L13-Batch Normalization
L14-Soiftmax
L15-Convolutional Neural Networks
L16-Padding, Strided convolution
L17-Simple Convolutional Network
L18- Genetic Algorithms
L19- Genetic Algorithms
L1-Introduction
L6-Particle Swarm Intelligence (PSO)
L7-Particle Swarm Intelligence (PSO)
L8-Discrete PSO
L9-PSO Hyper-parameters tuning
L10-PSO Applications
L11-PSO Applications
L17-Ant Colony Optimization (ACO)
L18-Ant Colony Optimization (ACO)
L19-Ant Colony Optimization (ACO)
L20-ACO applications
L21-ACO applications
L22-ACO applications

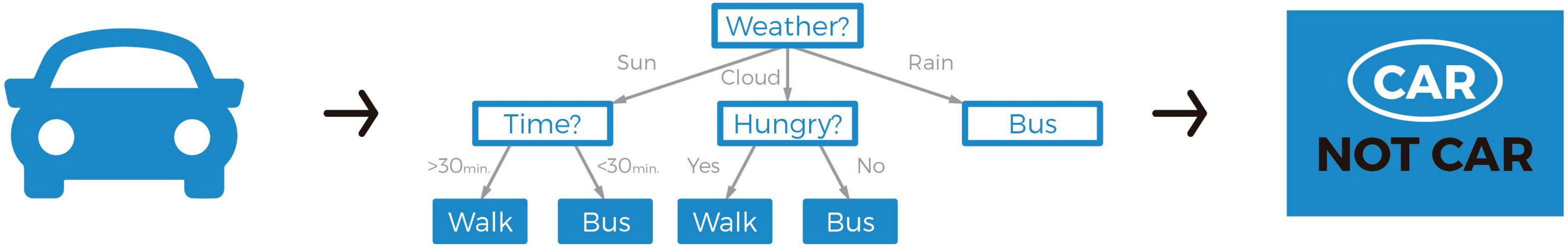
What is Neural Network?



Why do we need to neural networks?

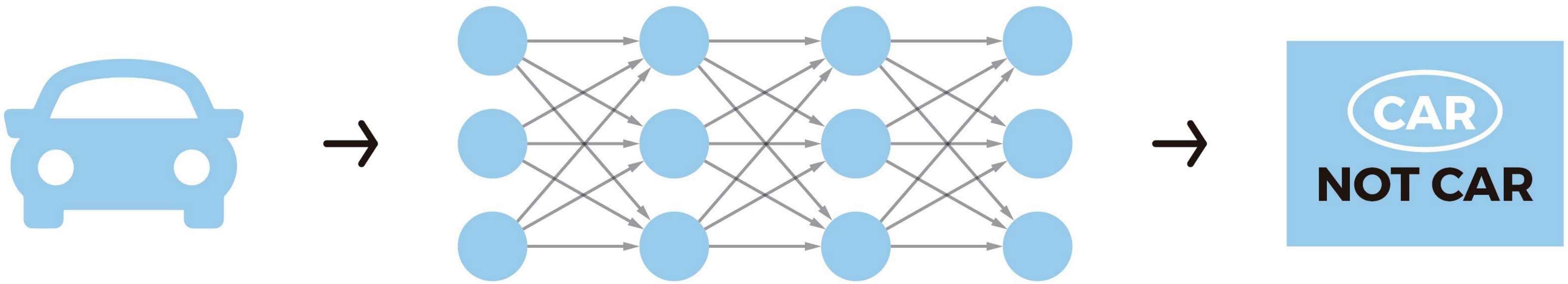
- ML algorithms are mathematical algorithms that allow machines to learn by imitating the way humans learn,
- Machine learning is basically a way to get artificial intelligence.
- Instead of writing a program by hand, we collect lots of examples that specify the correct output for a given input.
- A machine learning algorithm then takes these examples and produces a program that does the job.

Machine Learning



Input Decision tree Output

Deep Learning



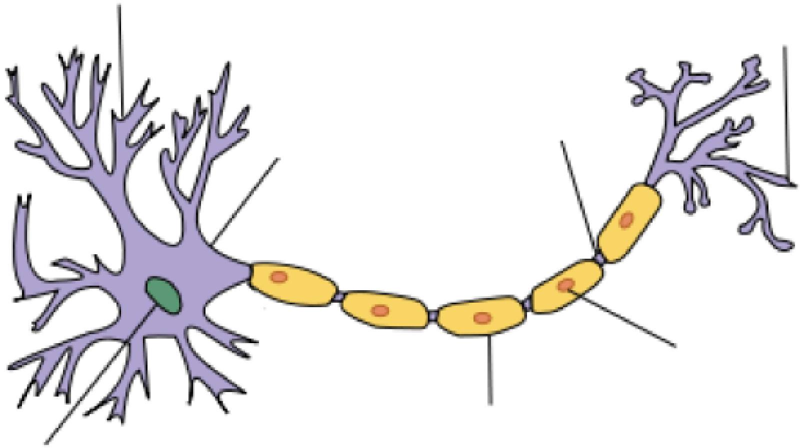
Input Feature extraction + Classification Output

What is Neural Networks?

- Neural Networks attempts to learn representations of data with multiple levels of abstraction.
- Neural Networks usually refers to a set of algorithms and computational models that are composed of multiple processing layers.
- These methods have significantly improved the state-of-the-art in many domains including, speech recognition, classification, pattern recognition, drug discovery, and genomics.

Human brains

- A brain is a set of densely connected neurons.
- Components of a neuron: dendrites, soma, axon, synapse
- Depending on the input signals, the neuron performs computations and decides to fire or not.



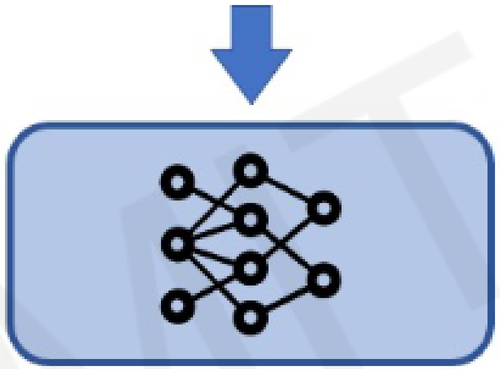
Applications

- Medical diagnosis
- Spam filtering
- Facial recognition
- Speech understanding
- Handwriting recognition

Success Stories

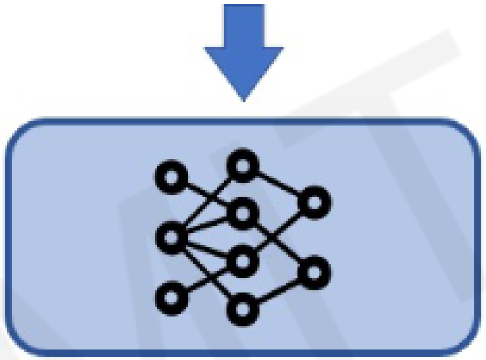
Generating Images from natural language

“A photo of an astronaut riding a horse.”



Generating Images from natural language

“Write code in TensorFlow to train a neural network.”



Certainly! Here is an example of how you can use TensorFlow to train a neural network for binary classification:

Copy code

```
import tensorflow as tf

# Load the data
(x_train, y_train), (x_test, y_test) = load_data()

# Define the model
model = tf.keras.Sequential([
    tf.keras.layers.Input(shape=x_train.shape[1:]),
    tf.keras.layers.Dense(16, activation='relu'),
    tf.keras.layers.Dense(1, activation='sigmoid')
])

# Compile the model
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=
['accuracy'])

# Train the model
model.fit(x_train, y_train, epochs=10, batch_size=32, validation_data=
(x_test, y_test))
```

This code assumes that you have a function `load_data()` that returns the training and test datasets as tuples of NumPy arrays. The input shape of the model is determined by the shape of the training data, and the model has two dense layers with 16 and 1 units, respectively, with ReLU and sigmoid activations. The model is then compiled with an Adam optimizer and a binary cross-entropy loss function, and is trained using the `fit()` method.

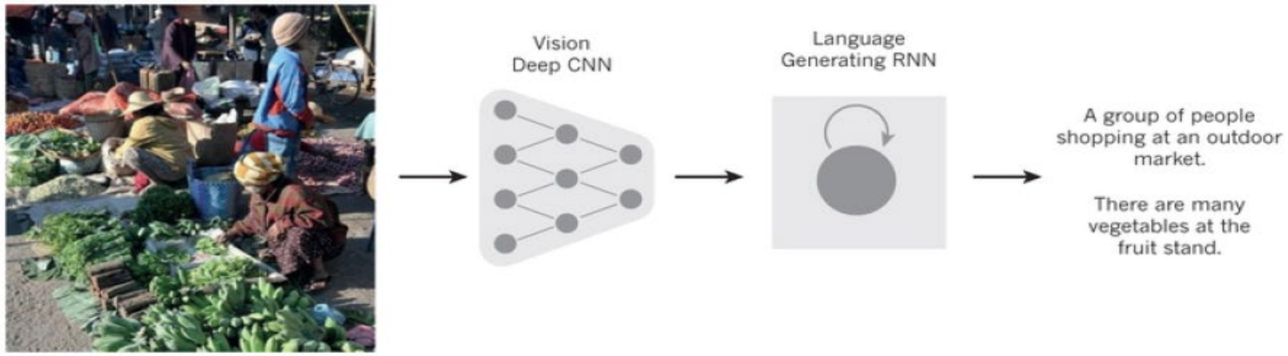
Video Generation

Games

Deep Learning Machine Teaches Itself Chess in 72 Hours, Plays at International Master Level.

An artificial intelligence machine plays chess by evaluating the board rather than using brute force to work out every possible move.

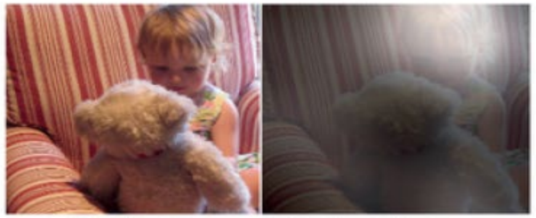
Caption Generation




A woman is throwing a **frisbee** in a park.


A **dog** is standing on a hardwood floor.

A **stop** sign is on a road with a mountain in the background


A little **girl** sitting on a bed with a teddy bear.


A group of **people** sitting on a boat in the water.

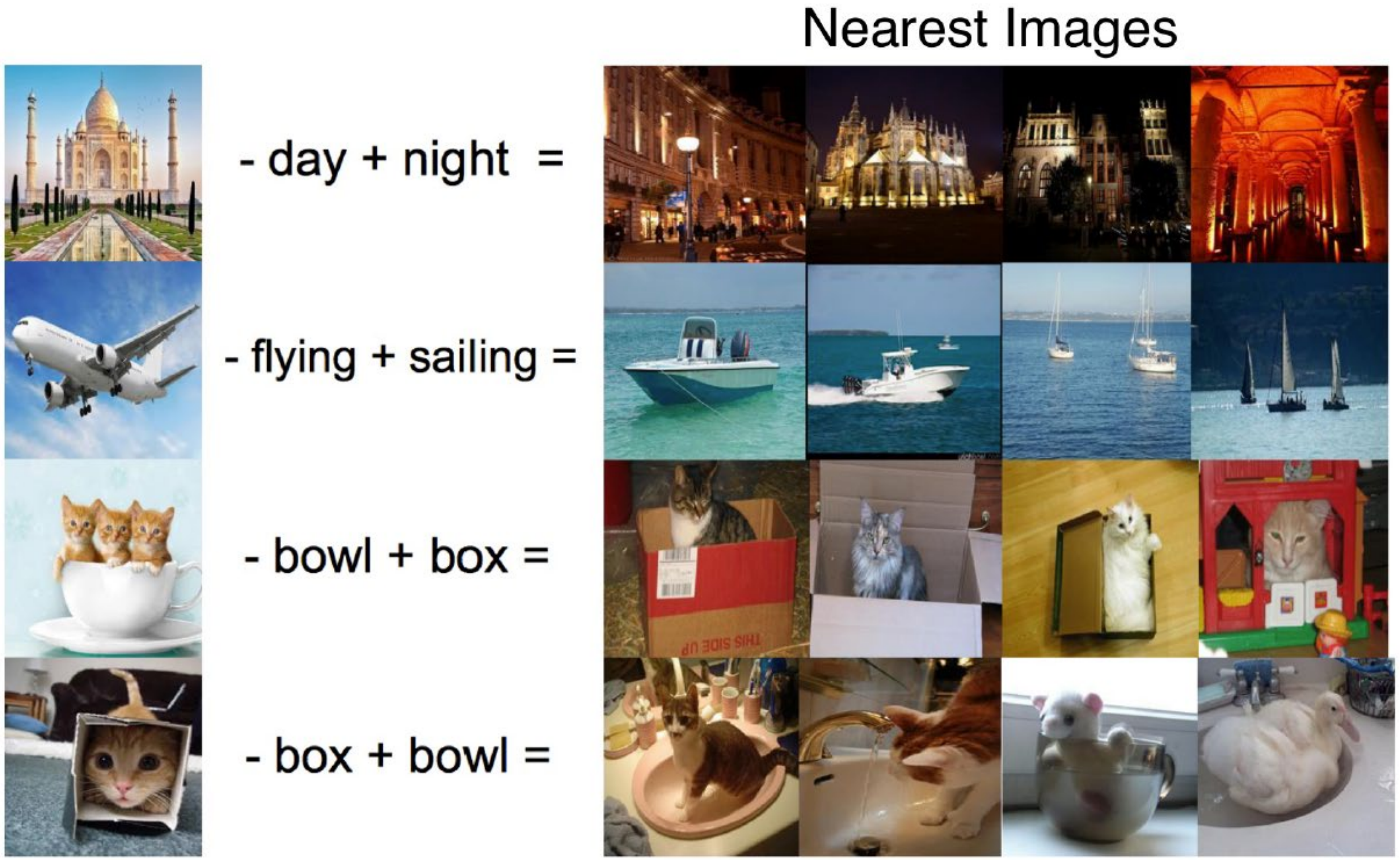

A giraffe standing in a forest with **trees** in the background.

Word embeding

Word2vec , Mikolov, 2013.

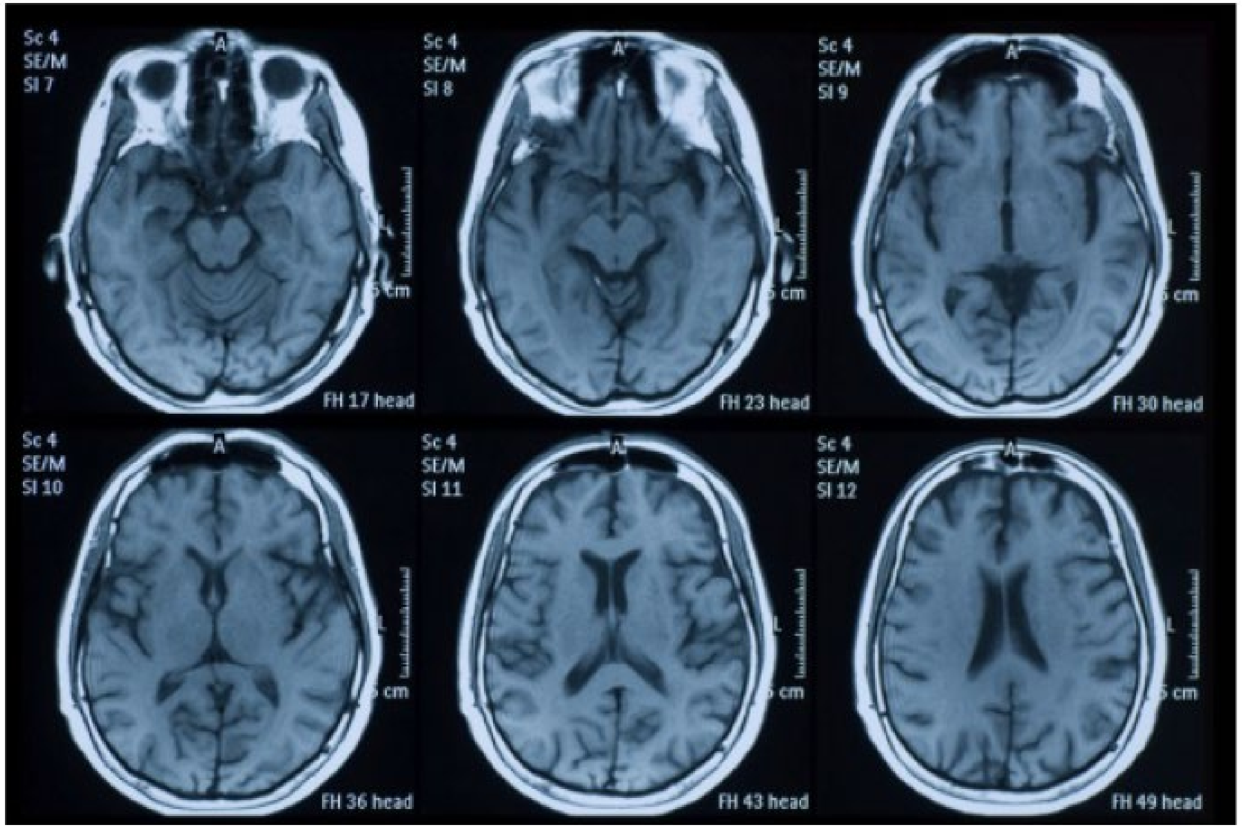
$$\text{king} - \text{man} + \text{woman} = \text{queen}$$

Word embeding



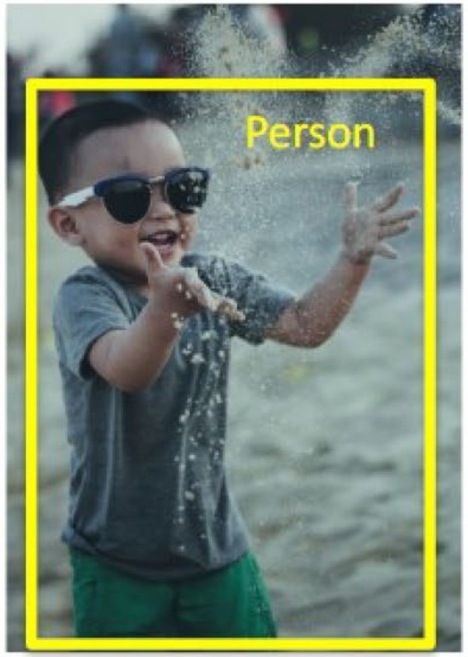
(Kiros, Salakhutdinov, Zemel, TACL 2015)

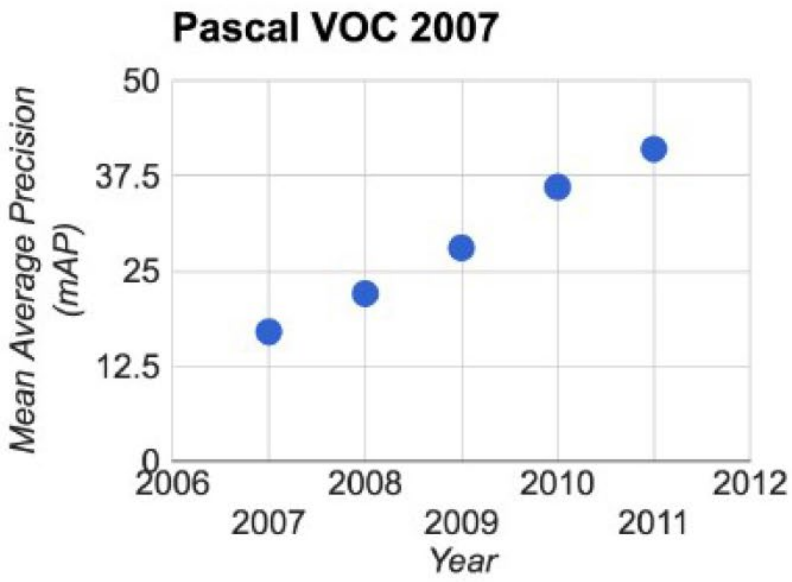
Medical Image processing



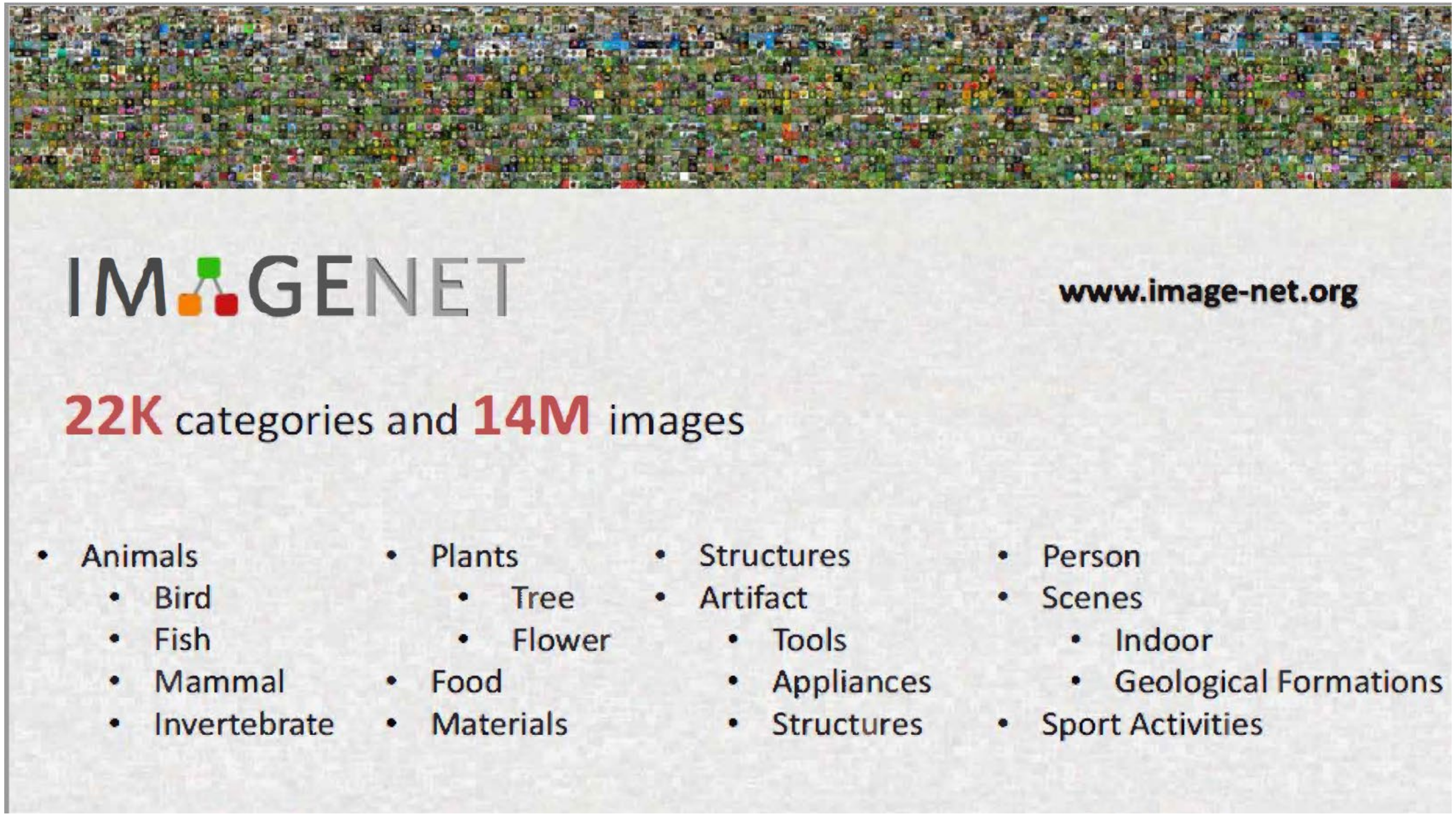
Success Stories

PASCAL Visual Object Challenge
(20 object categories)
[Everingham et al. 2006-2012]

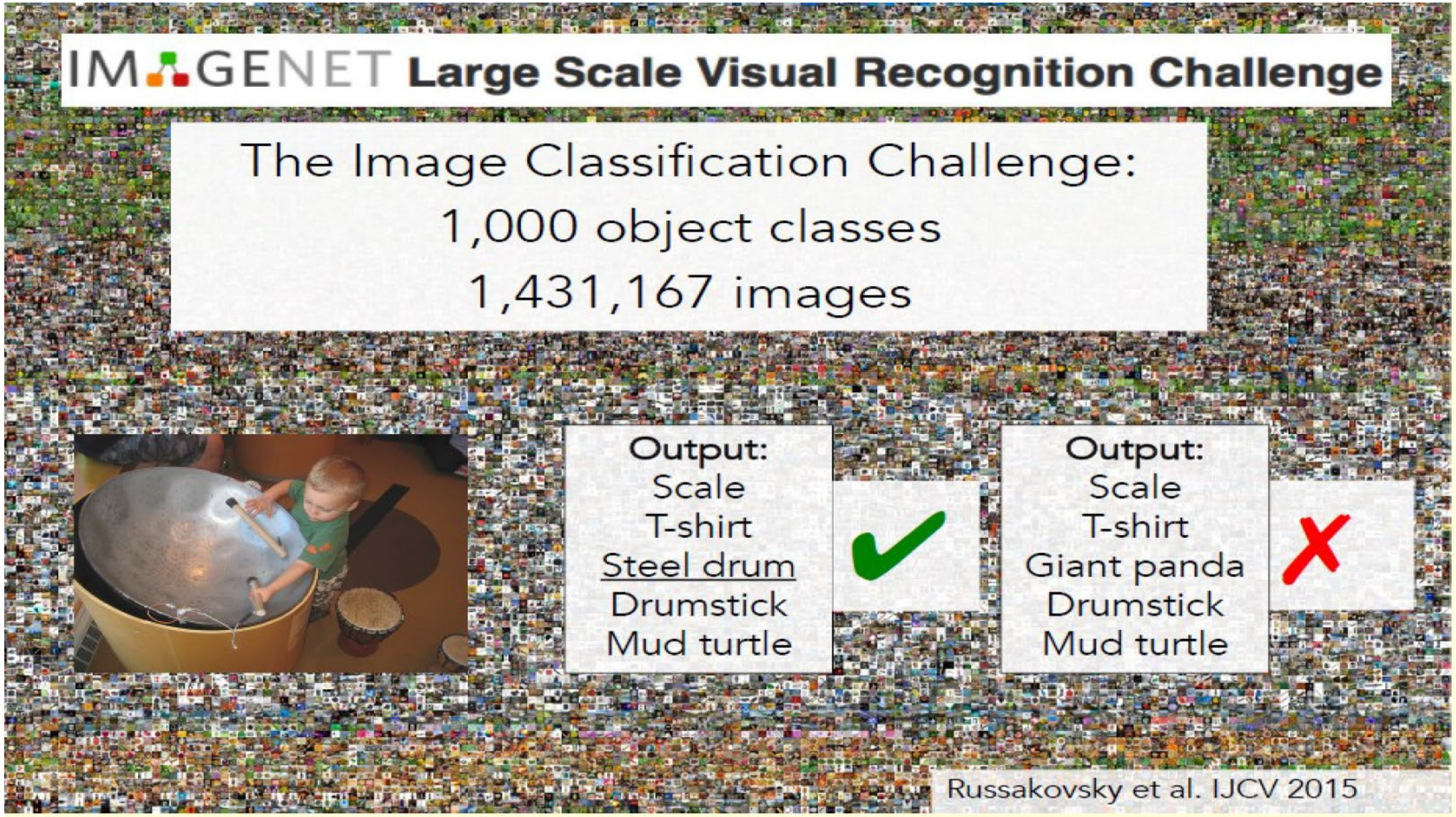




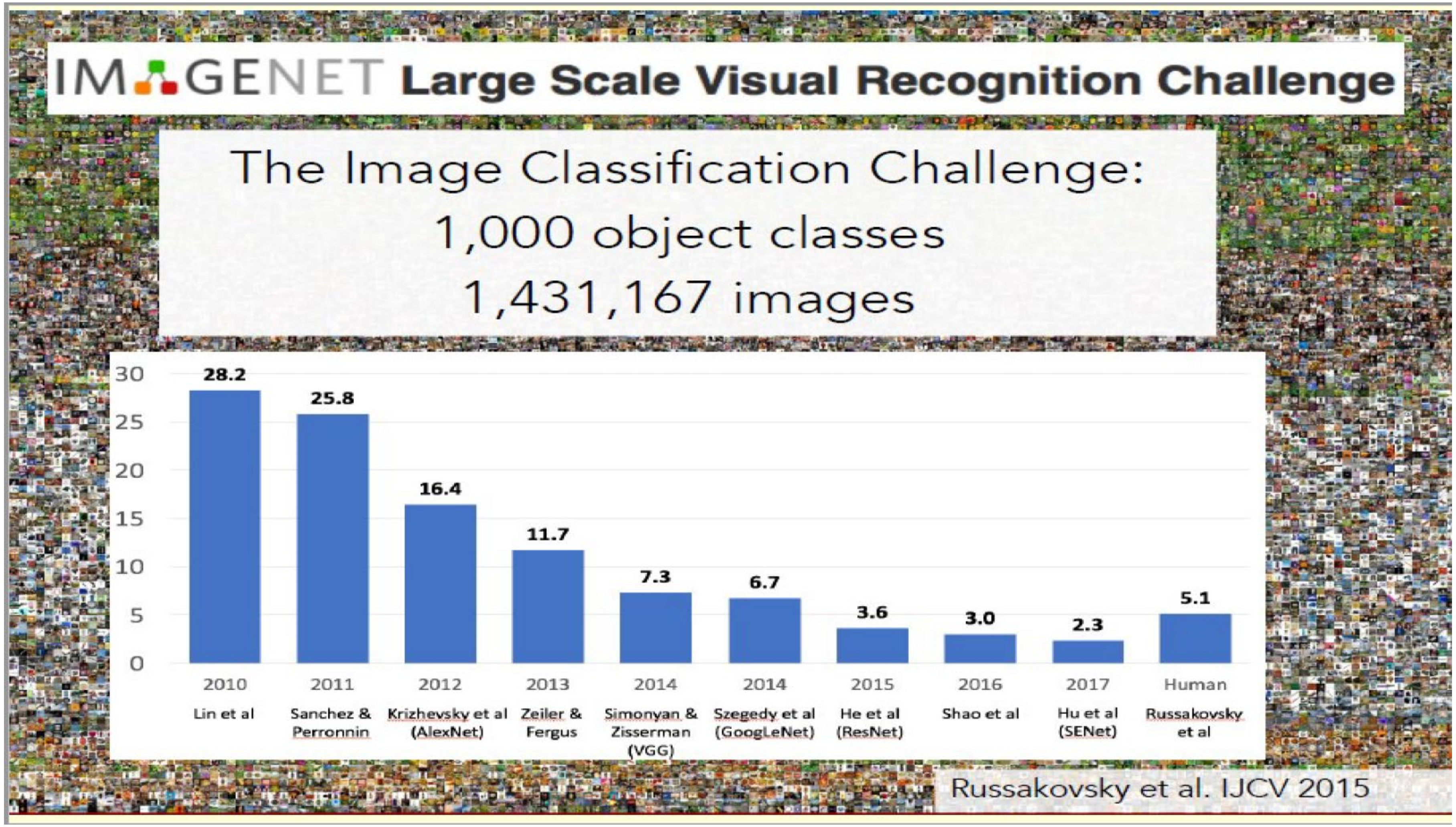
Success Stories



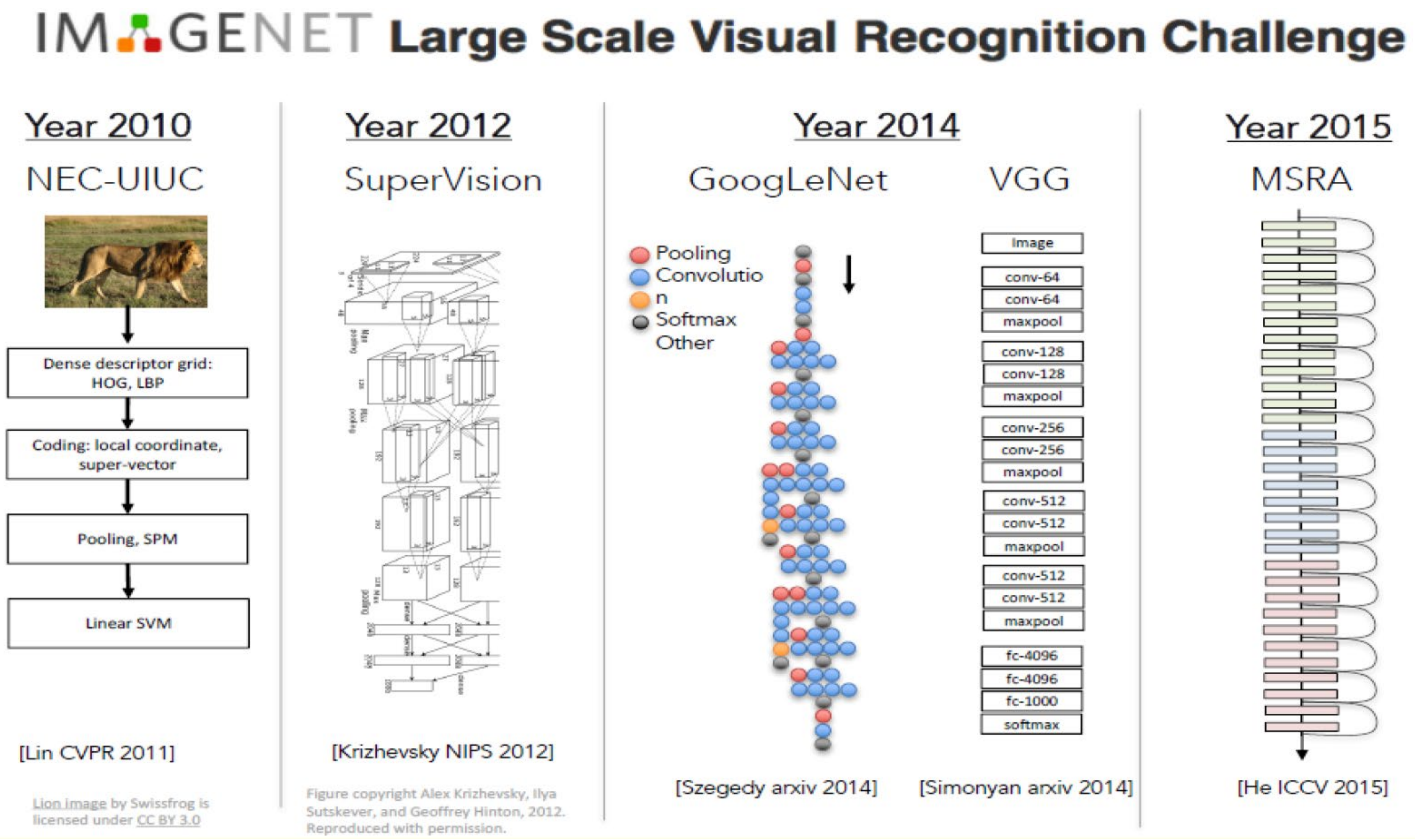
Success Stories



Success Stories



Success Stories



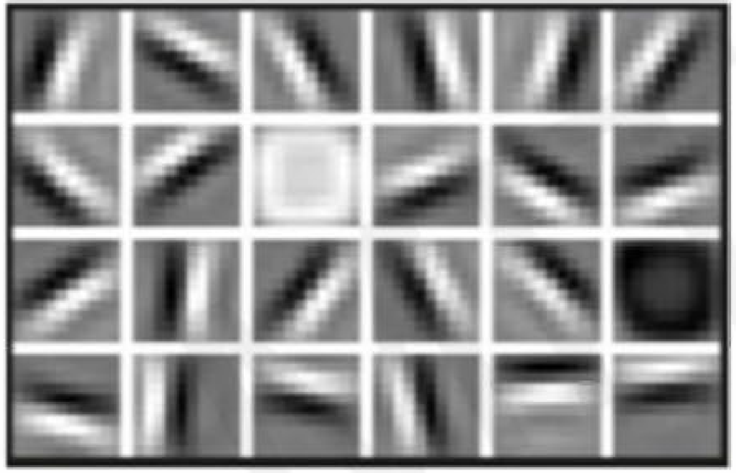
Why Deep Learning and Why now?

Why Neural Networks?

Hand engineered features are time consuming, brittle, and not scalable in practice

Can we learn the **underlying features** directly from data?

Low Level Features



Lines & Edges

Mid Level Features

Eyes & Nose & Ears

High Level Features

Facial Structure

Why Now?

1952 Stochastic Gradient Descent

1958 Perceptron
- Learnable Weights

⋮

1986 Backpropagation
- Multi-Layer Perceptron

1995 Deep Convolutional NN
- Digit Recognition

⋮

Neural Networks date back decades, so why the resurgence?

1. Big Data
- Larger Datasets
- Easier Collection & Storage

IMAGENET

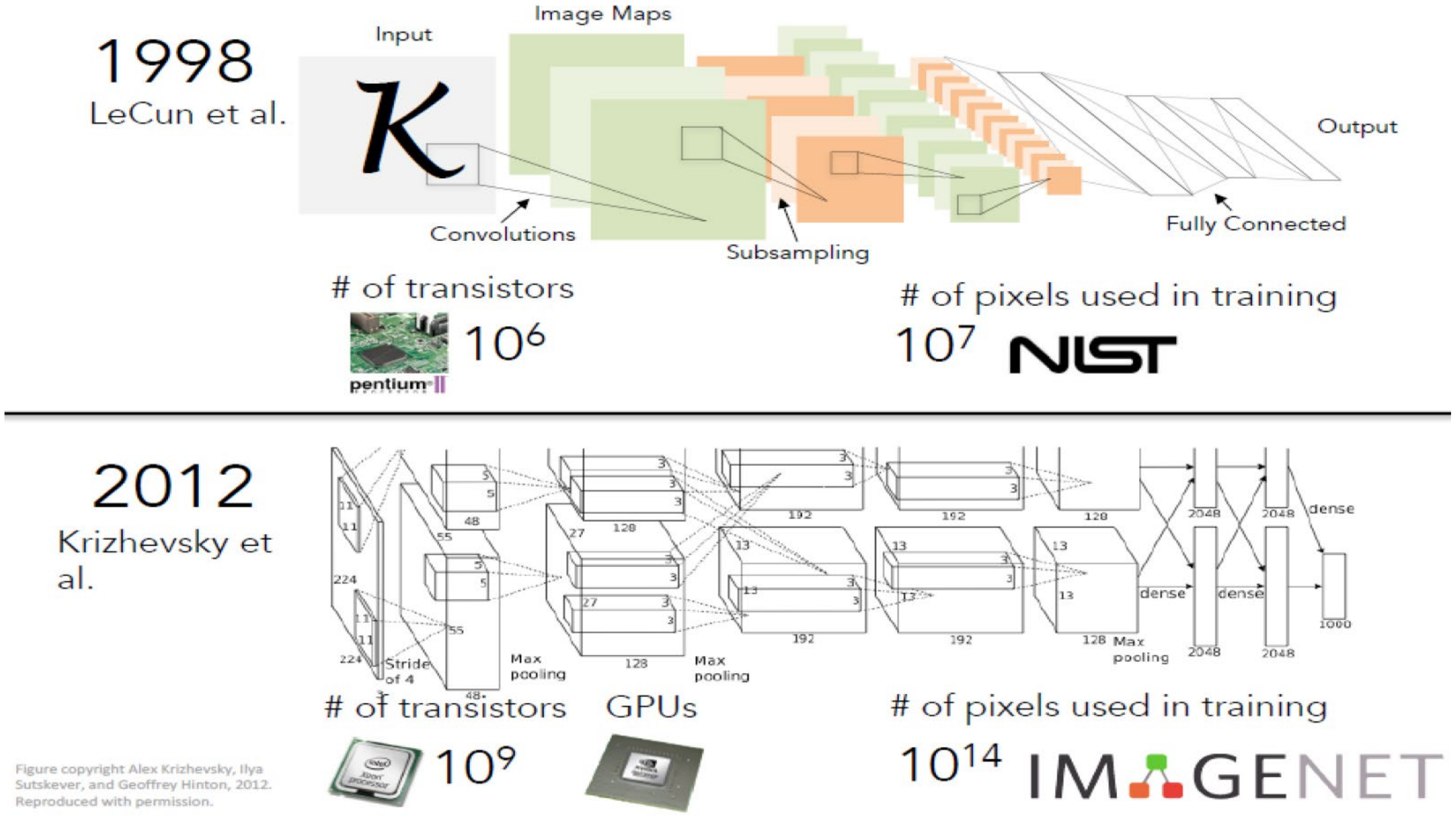
WIKIPEDIA
The Free Encyclopedia

2. Hardware
- Graphics Processing Units (GPUs)
- Massively Parallelizable

3. Software
- Improved Techniques
- New Models
- Toolboxes

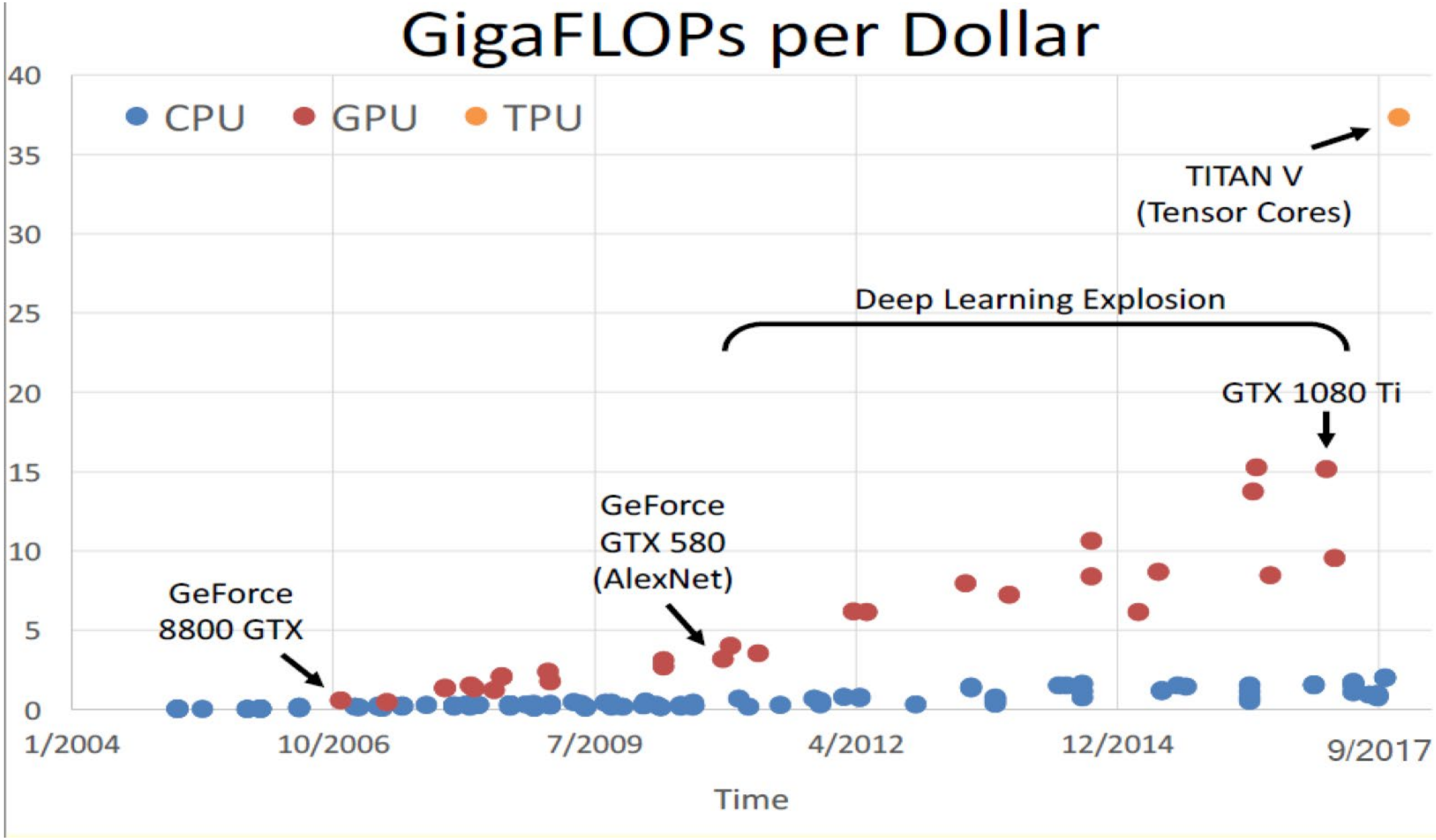
TensorFlow

Why now?



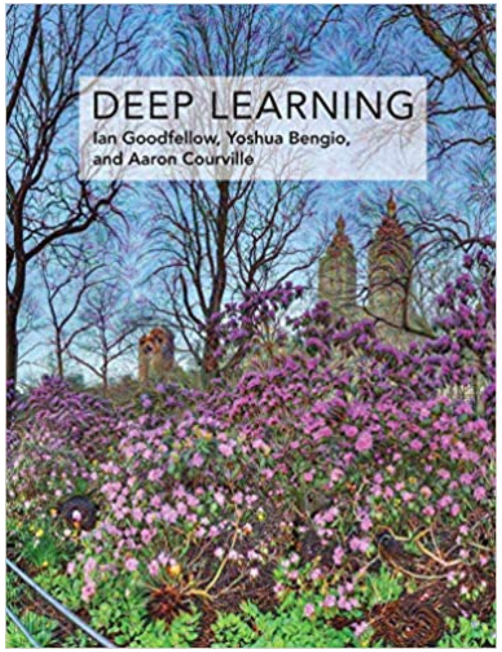
1998
LeCun et al.
Input
Image Maps
Output
Convolutions
Subsampling
Fully Connected
# of transistors
10^6
pentium II
# of pixels used in training
10^7
NIST
2012
Krizhevsky et al.
Stride of 4
Max pooling
dense
# of transistors
10^9
GPUs
# of pixels used in training
10^{14}
IMAGENET

Why now?


GigaFLOPs per Dollar
CPU
GPU
TPU
40
35
30
25
20
15
10
5
0
1/2004
10/2006
7/2009
4/2012
12/2014
9/2017
Time
TITAN V
(Tensor Cores)
Deep Learning Explosion
GTX 1080 Ti
GeForce
GTX 580
(AlexNet)
GeForce
8800 GTX

Text Books

- Deep learning, I. Goodfellow, j. Bengio MIT Press, 2016
- Neural Networks and deep learning, M. Nielsen Determination Press, 2015



Syllabus

L1-Introduction
L2-Neuron math model
L3-Perceptron
L4-Building and Applying NN
L5-Gradient descent
L6-Vectorization
L7-Overfitting
L8-Regularization I
L9-Regularization II
L10-Optimization Algorithms I (mini-batches)
L11-Optimization Algorithms II(exponentially weighted averages)
L12-Hyperparameter Tuning
L13-Batch Normalization
L14-Soiftmax
L15-Convolutional Neural Networks
L16-Padding, Strided convolution
L17-Simple Convolutional Network

Related Course

- Deep learning, Andrew Ng, Stanford University

- Intro to Deep Learning, MIT

- Neural Network for Machin Learning

Course administrations

Schedule

تاریخ	موضوع
یکشنبه, 14 بهمن 1403	L1-Introduction
سه شنبه, 16 بهمن 1403	L2-Neuron math model
یکشنبه, 21 بهمن 1403	L3-Perceptron
سه شنبه, 23 بهمن 1403	L4-Building and Applying NN
یکشنبه, 28 بهمن 1403	L5-Gradient descent
سه شنبه, 30 بهمن 1403	L6-Vectorization
یکشنبه, 5 اسفند 1403	L7-Overfitting
سه شنبه, 7 اسفند 1403	L8-Regularization I
یکشنبه, 12 اسفند 1403	L9-Regularization II
سه شنبه, 14 اسفند 1403	L10-Optimization Algorithms I (mini-batches)
یکشنبه, 19 اسفند 1403	L11-Optimization Algorithms II(exponentially weighted averages)
سه شنبه, 21 اسفند 1403	L12-Hyperparameter Tuning
یکشنبه, 26 اسفند 1403	L13-Batch Normalization
سه شنبه, 28 اسفند 1403	L14-Soiftmax
یکشنبه, 17 فروردین 1404	L15-Convolutional Neural Networks
سه شنبه, 19 فروردین 1404	L16-Padding, Strided convolution
یکشنبه, 24 فروردین 1404	L17-Simple Convolutional Network
سه شنبه, 26 فروردین 1404	L18- Genetic Algorithms
یکشنبه, 31 فروردین 1404	L19- Genetic Algorithms
سه شنبه, 2 اردیبهشت 1404	L1-Introduction
یکشنبه, 7 اردیبهشت 1404	L6-Particle Swarm Intelligence (PSO)
سه شنبه, 9 اردیبهشت 1404	L7-Particle Swarm Intelligence (PSO)
یکشنبه, 14 اردیبهشت 1404	L8-Discrete PSO
سه شنبه, 16 اردیبهشت 1404	L9-PSO Hyper-parameters tuning
یکشنبه, 21 اردیبهشت 1404	L10-PSO Applications
سه شنبه, 23 اردیبهشت 1404	L11-PSO Applications
یکشنبه, 28 اردیبهشت 1404	L17-Ant Colony Optimization (ACO)
سه شنبه, 30 اردیبهشت 1404	L18-Ant Colony Optimization (ACO)
یکشنبه, 4 خرداد 1404	L19-Ant Colony Optimization (ACO)
سه شنبه, 6 خرداد 1404	L20-ACO applications
یکشنبه, 11 خرداد 1404	L21-ACO applications
سه شنبه, 13 خرداد 1404	L22-ACO applications

Course Staff

- سمانه حسینی
- استادیار گروه هوش مصنوعی
- دانشگاه صنعتی اصفهان
- samane.hossayni@gmail.com
- samaneh.hoseini@iut.ac.ir

- Behnaz Aalipur
  Email: b.aalipur@ec.iut.ac.ir
  Telegram: @Behnaz_Aa
- A. Azhand
  Email: a.azhand@ec.iut.ac.ir
  Telegram: @arash_azhand
- M. Kafi
  Email: m.kafi@me.iut.ac.ir
  Telegram: @iamin_p
- A. Taheri
  Email: taheri.a@ec.iut.ac.ir
  Telegram: @AliTaheri2002
- Farzaneh Koohestani
  Email: farzanehkoohestani1999@gmail.com
  Telegram: farzaneh_koohestani

Assignments

Calss Rules

- تمرین ها به صورت انفرادی انجام میشود.
- زمان تحویل هر تمرین ۲ هفته بعد از پست آن تمرین است
- زمان تحویل تمرین ها مطابق با فایل زمانبندی انجام میشود و قابل تغییر نیست
- تمرین ها با ۸ روز تاخیر در کل تمرین ها قابل تحویل هستند
- در صورتی که ۸ روز شما به اتمام رسید، تمرینهای بعدی را میتوانید هر کدام را تا ۳ روز به ازای هر روز کسر ۱۰ درصد تحویل دهید.
- بعد از ۸ روز جواب تمرین توسط تی ای ها در سامانه بازگزاری میشود.
- بعد از پست پاسخ تمرین ها تحویل تمرین مجاز نیست.

Calss Rules

- نمره امتیازی برای هر تمرین عملی به صورتی انتخاب شود که رقابت بین گروه ها ایجاد شود.
- نمره امتیازی برای کل تکالیف بین ۰.۵ تا ۱ نمره خواهد بود
- تکالیف عملی توسط سیستم های چک تشابه بررسی می شوند و در صورت مشابهت بالا به هر دو نفر نمره ۰ تعلق می گیرد.
- تکالیف تئوری در صورت مشاهده تشابه بالا برای هر دو نفر نمره ۰ در نظر گرفته می شود.

Marking Scheme

- Assignments and Project: 3-5 marks
- Midterm: 5-7 marks
- Final: 5-7 marks
- Clicker Questions (probebly): 1 mark
- Attendance in class: 1-2 mark