Deep Learning

Lecture 1: Introduction to DL

National University of Singapore
CEG5304/EE6934
Instructor: Joey Zhou (A*STAR/NUS)

Lecturers

- Dr. Joey Zhou, for Part I
- Dr. Robby Tan, for part II

Teaching Assistants

Mr. Liu, Jiawei (email: e0692797@u.nus.edu) (Project 1 and 2)
Mr. Wu Zhangjie(email:e0823041@u.nus.edu) (Project 3 and 4)

Q&A Google doc:

https://docs.google.com/document/d/159iXIH0E_acPCtPe6FHb5Er3NkOTqn6Ihx74BT_XCpU/edit?usp=sharing

Acknowledgements

For various materials from:

COS 495 Introduction to Deep Learning, Princeton University
EE 2211 Introduction to Machine Learning, NTU
11-785 Introduction to Deep Learning, Carnegie Mellon University
MIT 6.S191 Introduction to Deep Learning, MIT
C19 Machine Learning, Oxford
CS276 Information Retrieval & Web Search , Stanford University
CS213N Convolutional Neural Networks for Visual Recognition, Stanford University
CS294-129 Designing, Visualizing & Understanding Deep Neural Networks, UC Berkeley
CS498 Introduction to Deep Learning, UIUC
CSC321 Introduction to Neural Networks & Machine Learning, University of Toronto
CS 898: Deep Learning and Its Applications
Introduction to Neural Networks & Deep Learning, UVA
etc.

Module Learning Outcomes

Achieve an understanding of basic **deep learning methods;** particularly **Deep Convolutional Neural Network** models;

Be informed of recent progress within the deep learning field;

Able to design, build and implement deep learning models;

Apply deep learning methods to solve practical problems in areas such as image processing, natural language processing and computer vision.

Pre-requisites

- **EE5907 Pattern Recognition**
- **Proficiency in Python**, high-level familiarity with C/C++
  - assignments will be in Python with some deep learning libraries in C++.
- Calculus, Linear Algebra
- Probability Theory & Statistics

Textbook and materials

- Deep Learning:

  http://www.deeplearningbook.org/

PYTHON is a **general-purpose programming language** which becomes a powerful environment for scientific work with several libraries: `numpy`, `scipy`, `matplotlib`.

PYTHON is almost like pseudocode and allows expression of powerful ideas in very few lines of code.

- Other software frameworks: Pytorch, Tensorflow, Caffe, Torch, Marvin, …

Some Python Resources

Installing scikit-learn (Ref: [Book2] Andreas C. Muller and Sarah Guido, “Introduction to Machine Learning with Python: A Guide for Data Scientists”, O’Reilly Media, Inc., 2017)

scikit-learn depends on two other Python packages, *NumPy* and *SciPy*. For plotting and interactive development, you should also install matplotlib, IPython, and the Jupyter Notebook. We recommend using the following prepackaged Python distributions, which provides the necessary packages:

Anaconda

A Python distribution made for large-scale data processing, predictive analytics, and scientific computing. Anaconda comes with NumPy, SciPy, matplotlib, pandas, IPython, Jupyter Notebook, and scikit-learn. Available on Mac OS, Windows, and Linux, it is a very convenient solution and is the one we suggest for people without an existing installation of the scientific Python packages. Anaconda now also includes the commercial Intel MKL library for free. Using MKL (which is done automatically when Anaconda is installed) can give significant speed improvements for many algorithms in scikit-learn.

Some tutorials that might be useful:

A quickstart tutorial on NumPy: https://numpy.org/devdocs/user/quickstart.html

Some community tutorials on Pandas: https://pandas.pydata.org/pandas-docs/stable/getting_started/tutorials.html

Scikit-learn tutorials: https://scikit-learn.org/stable/tutorial/index.html

Code of Conduct.

Students are encouraged to **actively participate**.

Your grade depends on what you do, not what others do..

Don’t share your solutions/codes… what you submit should be your own work..

Academic Integrity:

Student’s academic work & behavior are held to highest academic integrity standards. Cheating, plagiarism, unauthorized collaboration, and ‘helping others commit these acts’ are examples of academic misconduct, which can result in disciplinary action.

Any violation to academic honesty will be reported to the University's disciplinary panel.

Topics (PART I- Dr. Joey Zhou)

Week 1: Introduction: History of Deep Learning. Machine Learning Settings.

Week 2: Machine Learning Basics: Data representation. Linear Regression/Classification. K-NN. SVM.

Week 3: Machine Learning Basics II: Linear Regression/Classification, SVM, Nonlinear Classification, Overfitting, Regularization, Hypermeters.

Week 5: Fully Connected Neural Networks: Learning Visual Features. Training Neural Networks. Loss Optimization, Gradient Descent. Regularization..

Week 6: Autoencoder & CNN : Variants of Autoencoder CNN Pipeline. Adversarial examples/training.

Reeding Week: Why Deep? Why Small?: Deeper vs wider? Model Compression for DNN.

Topics (PART II- Dr. Robby Tan)

Week 8: Advanced CNN Models: Case study of AlexNet, ResNet, modern MLP, etc.

Week 9: Deep generative models: Introduction to generative probabilistic models and variational Bayesian. Variational auto-encoder models and their application generative adversarial networks and their application.

Week 10: Recurrent neural networks and Transformer: RNN, Transformer, image captioning.

Week 11: Deep reinforcement learning: Markov decision process. Deep reinforcement learning, policy optimization techniques application of deep RL.

Week 12: Bayesian Neural Networks: Bayesian deep learning, applications

Week 13: Few Shot Learning And Imbalanced Classes: few shot, imbalanced classes, out of distribution, etc.

Week 14: Review

Grading

- CEG5304
- Projects #1: 25%
- Projects #2: 25%
- Projects #3&4: 50% (Robby)

- EE6934
- Projects #1: 25%
- Projects #2: 25%
- Projects #3 and Essay: 50% (Robby)

Project #1 (10 Jan 2023 – 29 Jan 2023 6pm)
Project #2 (31 Jan 2023 – 6 Mar 2023 6pm)
Project #3 (TBD)
Project #4/Essay (TBD)

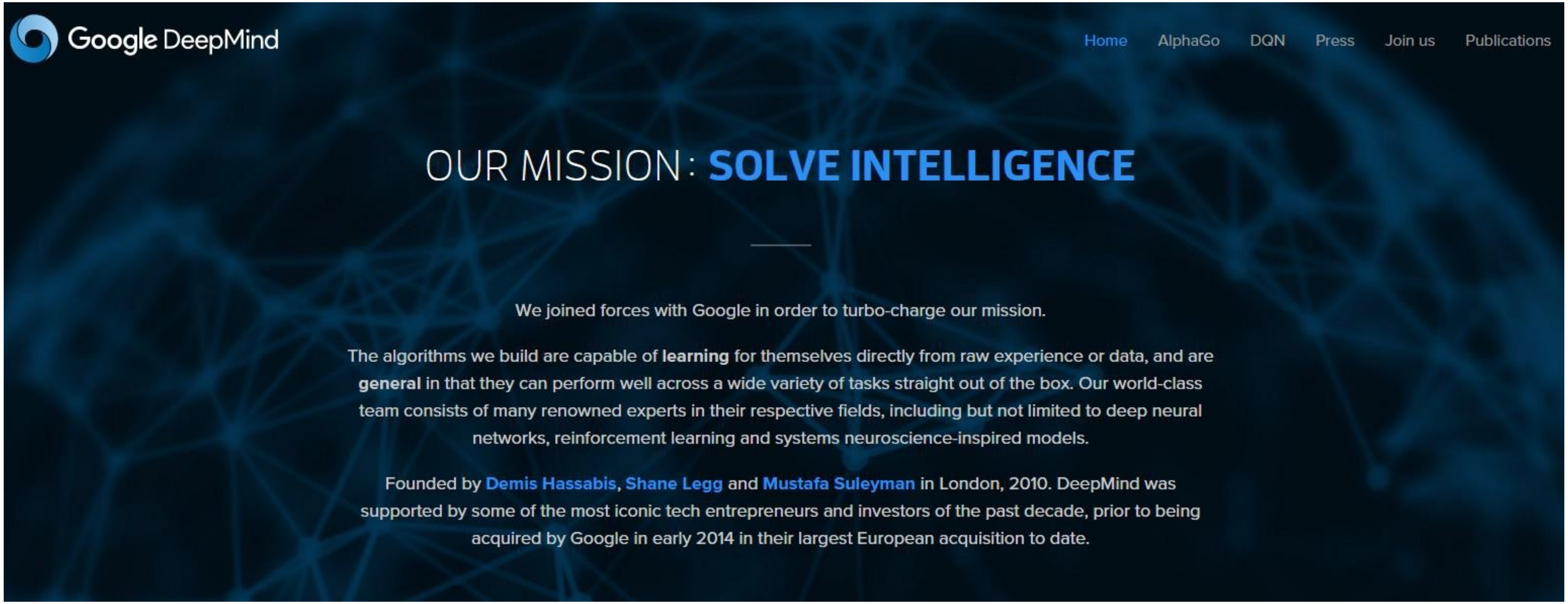
Deadlines are strict. Late submission will be deducted 10 points (out of 100) for every 24 hours.

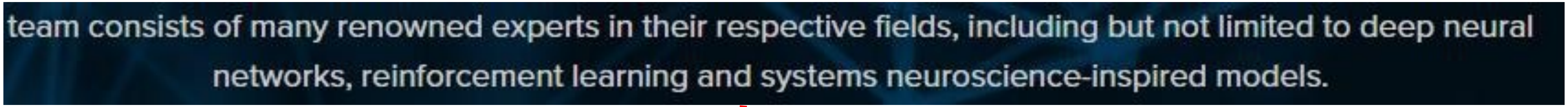
What is deep learning?

- Let us review this concept from industry side first.

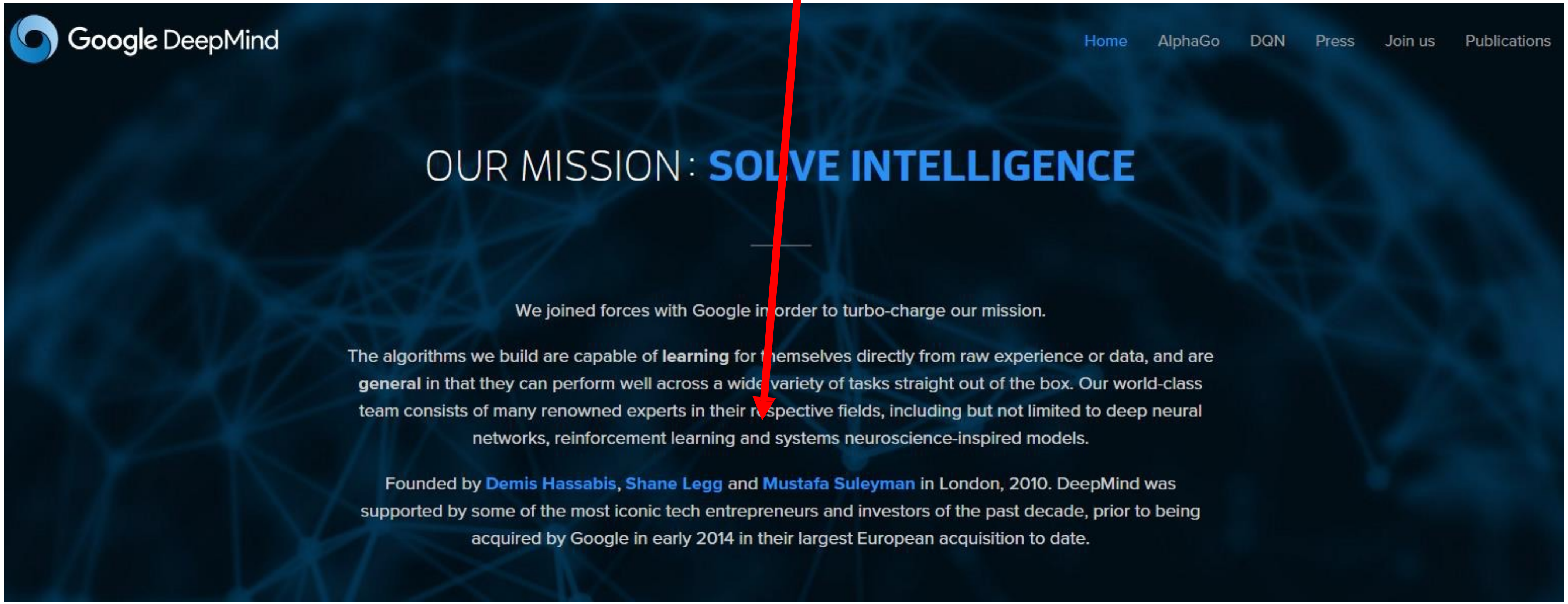
Industry

- Google



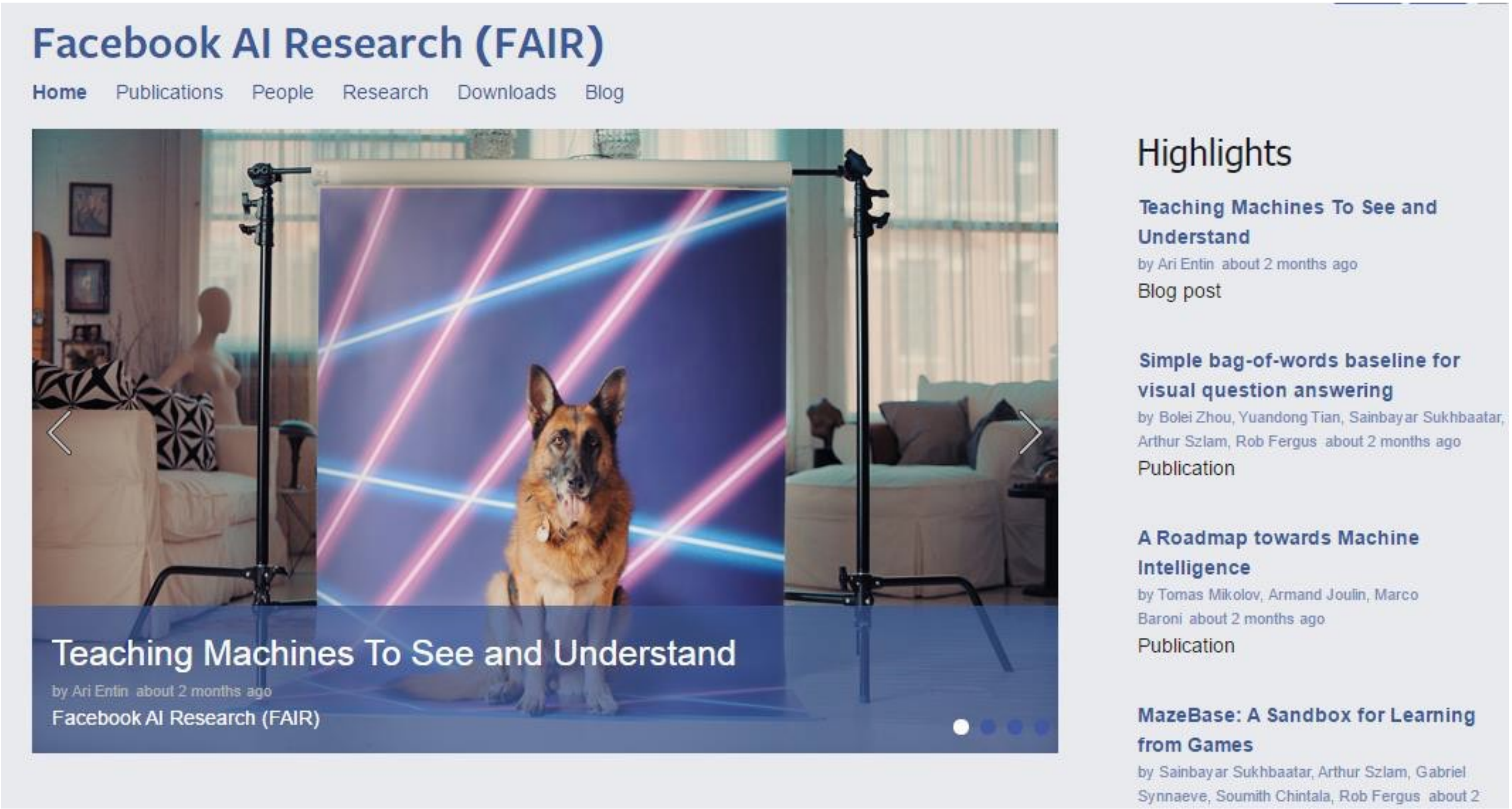

team consists of many renowned experts in their respective fields, including but not limited to deep neural
networks, reinforcement learning and systems neuroscience-inspired models.

- Google


Google DeepMind
Home
AlphaGo
DQN
Press
Join us
Publications
OUR MISSION: SOLVE INTELLIGENCE
We joined forces with Google in order to turbo-charge our mission.
The algorithms we build are capable of learning for themselves directly from raw experience or data, and are general in that they can perform well across a wide variety of tasks straight out of the box. Our world-class team consists of many renowned experts in their respective fields, including but not limited to deep neural networks, reinforcement learning and systems neuroscience-inspired models.
Founded by Demis Hassabis, Shane Legg and Mustafa Suleyman in London, 2010. DeepMind was supported by some of the most iconic tech entrepreneurs and investors of the past decade, prior to being acquired by Google in early 2014 in their largest European acquisition to date.

Industry

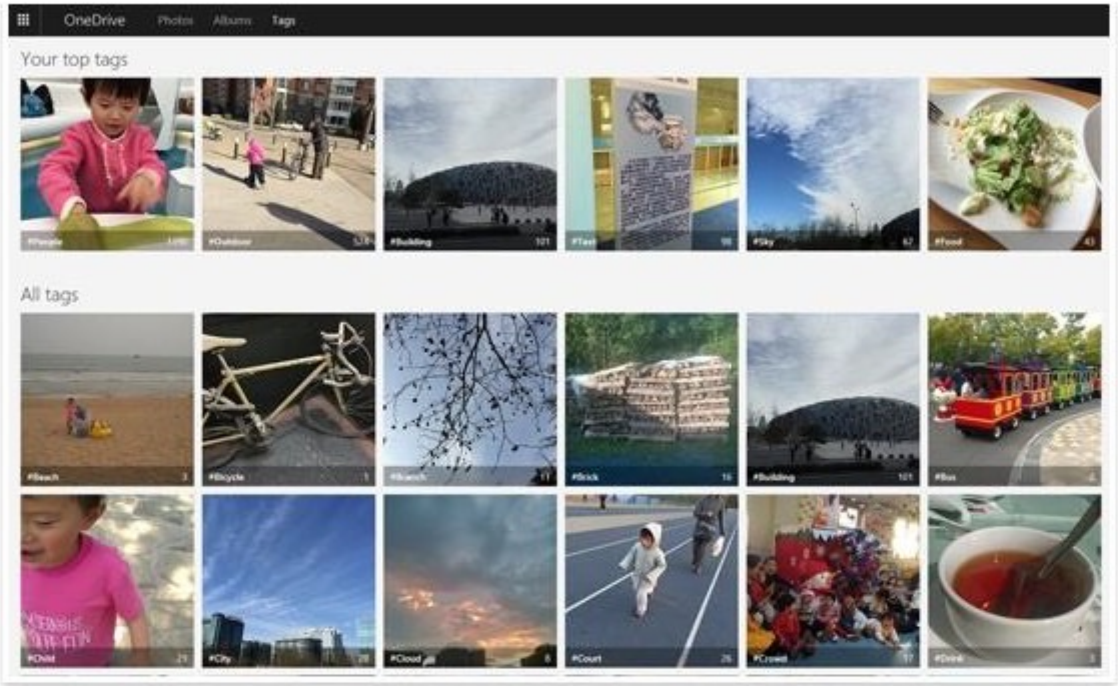
- Facebook



Industry

- Microsoft

Microsoft Researchers' Algorithm Sets ImageNet Challenge Milestone



Industry

- Elon Musk

Elon Musk @elonmusk · 3m
Announcing formation of @open_ai ...
openai.com/blog/introduci...

OpenAI

OpenAI
OpenAI is a non-profit artificial intelligence group.
openai.com

Forbes / Tech

Top 20 Stocks for 2016

DEC 11, 2015 @ 05:04 PM 4,715 VIEWS

Elon Musk And Peter Thiel Launch OpenAI, A Non-Profit Artificial Intelligence Research Company

Industry

- Toyota

SEARCH The New York Times

TECHNOLOGY

Toyota Invests $1 Billion in Artificial Intelligence

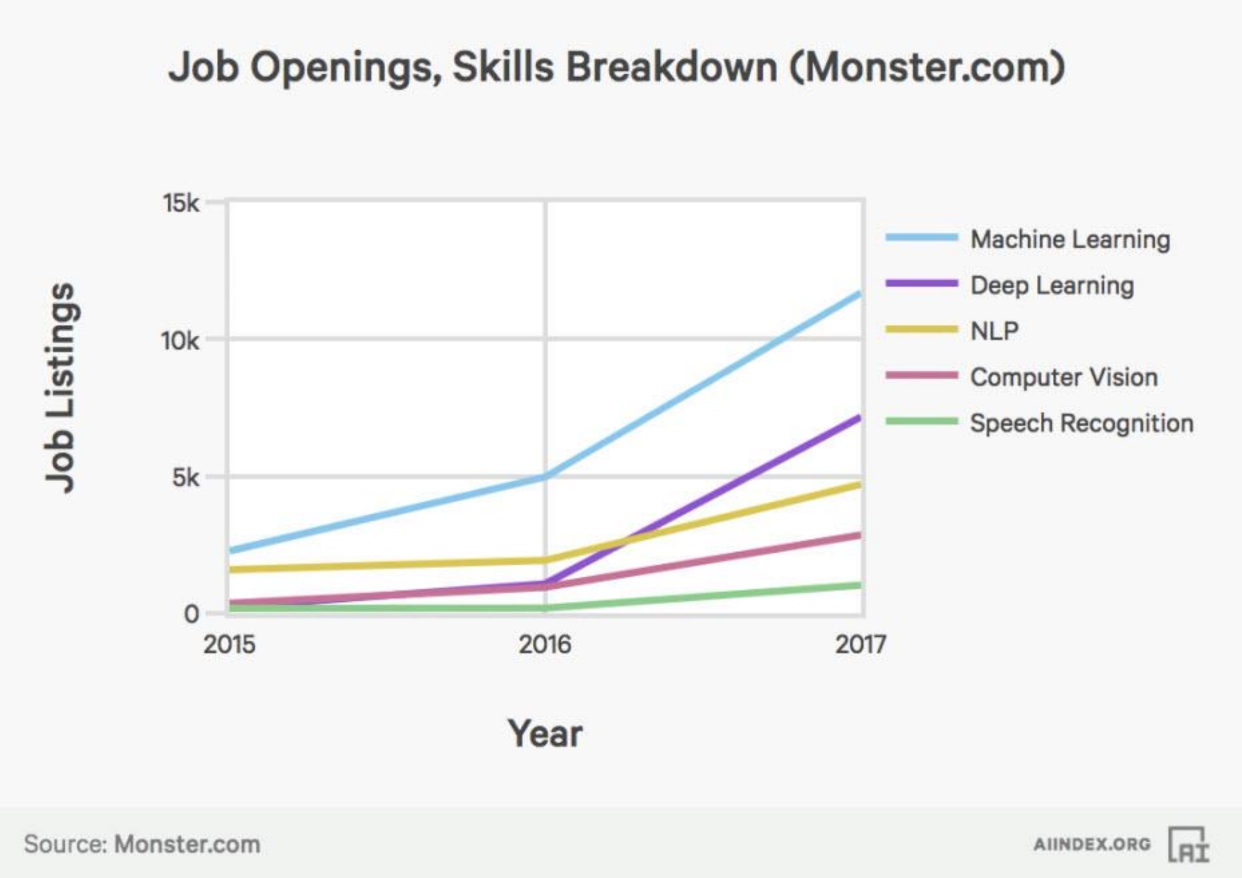
By JOHN MARKOFF NOV. 6, 2015

Gill Pratt, a roboticist who will oversee Toyota's new research laboratory in the United States, at a news conference Friday in Tokyo. Yuya Shino/Reuters

AI job titles with the highest salaries

Industry

Salary in Company

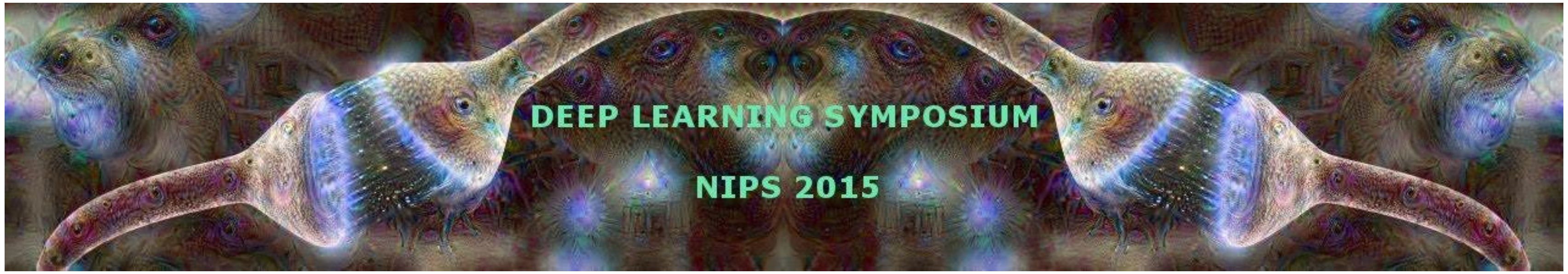


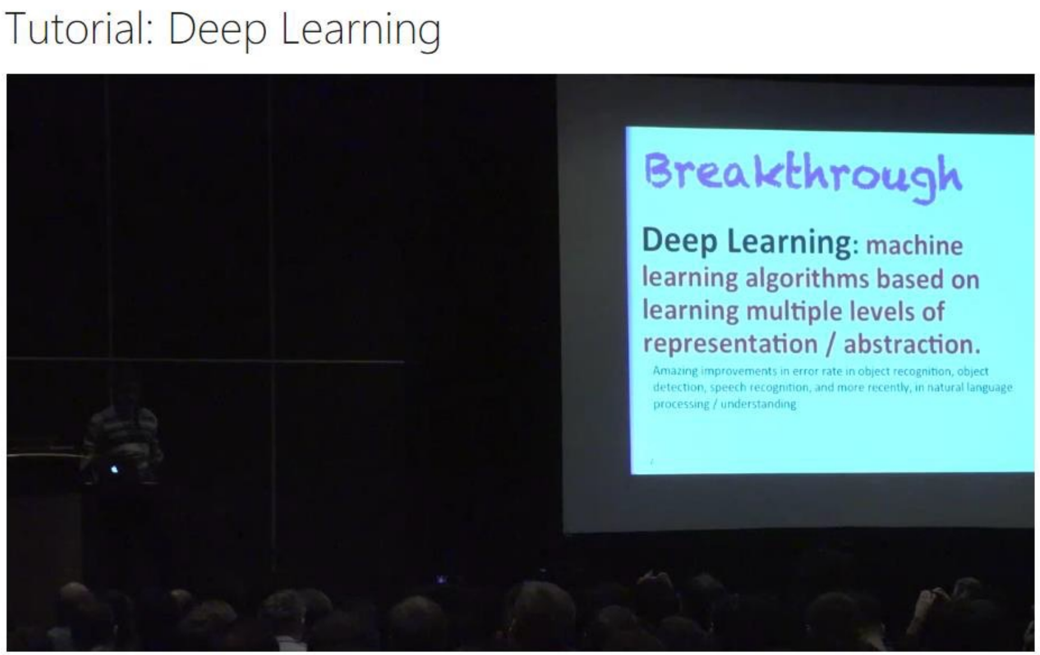
Rank	Job title	Average salary
1.	Machine learning engineer	$142,858.57
2.	Data scientist	$126,927.41
3.	Computer vision engineer	$126,399.81
4.	Data warehouse architect	$126,008.25
5.	Algorithm engineer	$109,313.51

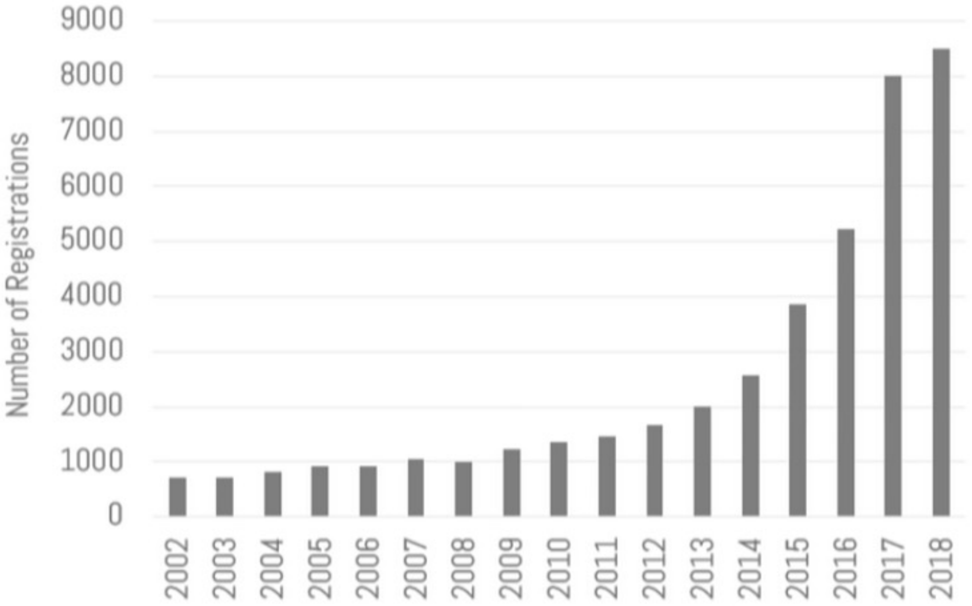
indeed

Academy

- NeurIPS 2018: ~8000 attendees, double the number of NIPS







Academy

- Science special issue
- Nature invited review

REVIEW

Deep learning

Yann LeCun[1,2], Yoshua Bengio[3] & Geoffrey Hinton[4,5]

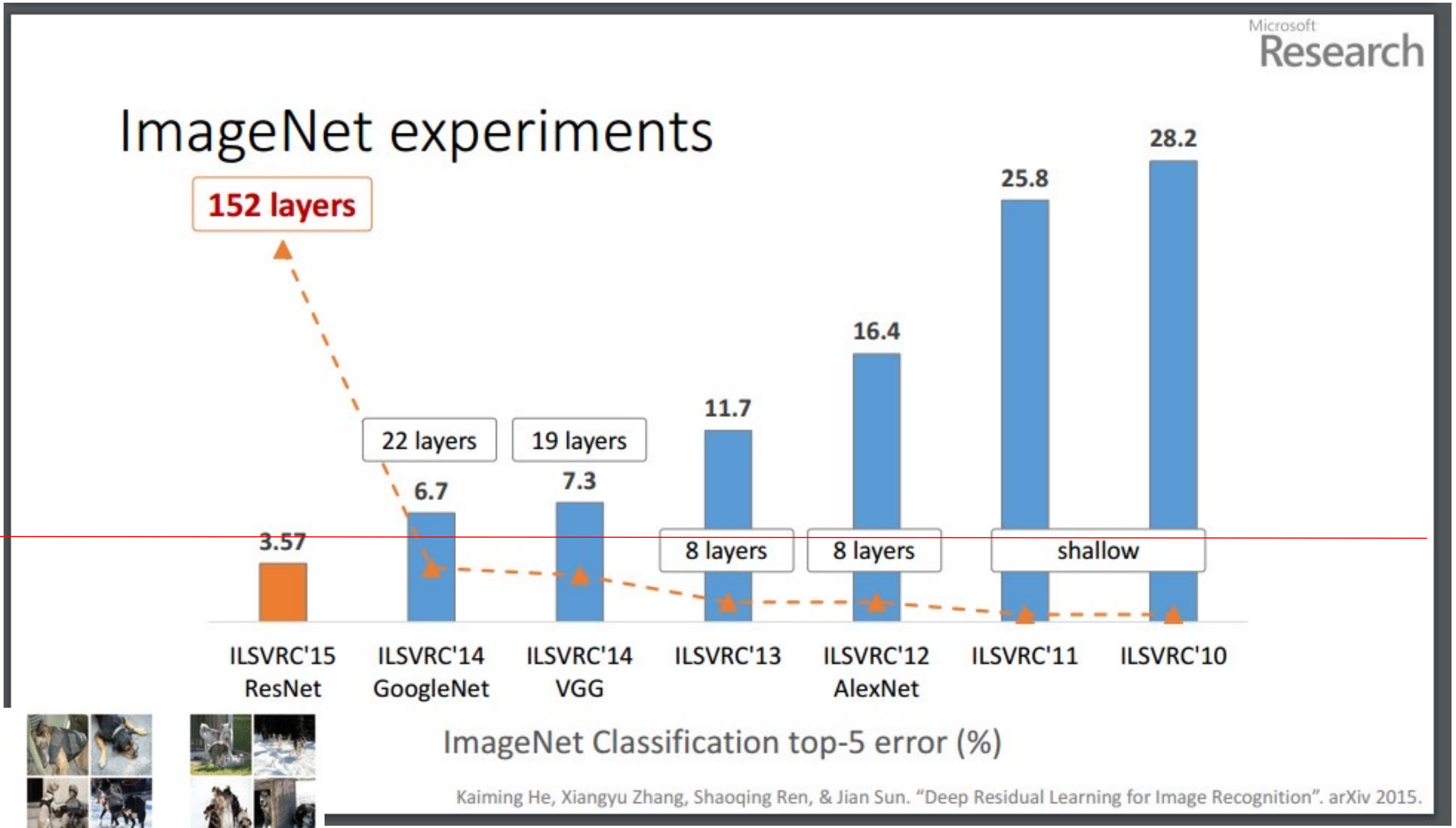
What is deep learning?

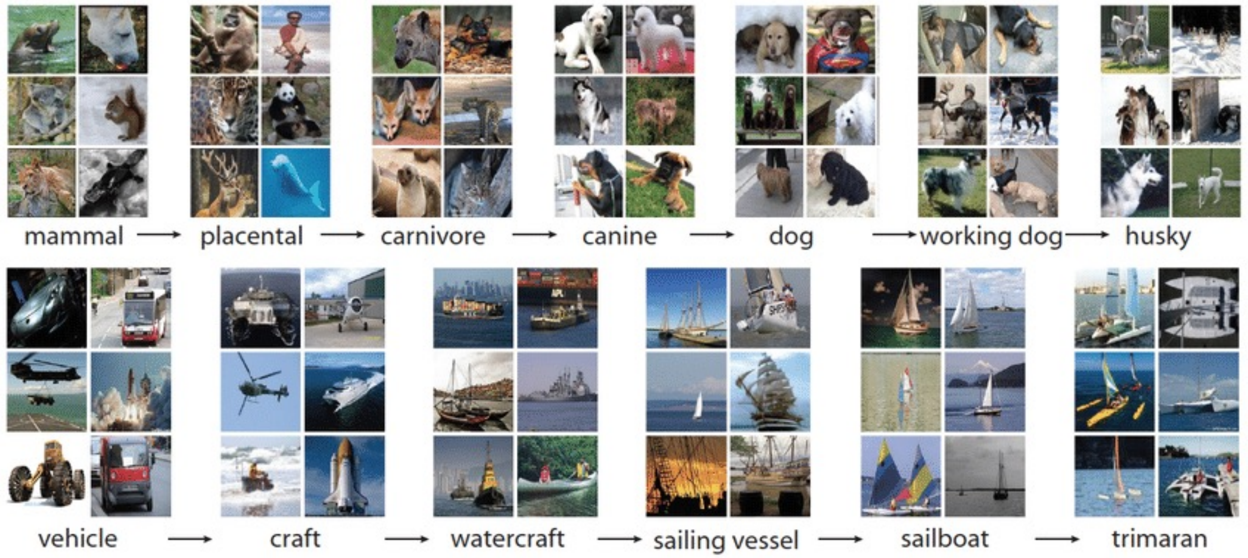
- Longer answer: machine learning framework that shows impressive performance on many Artificial Intelligence tasks

Image

- Image classification
  - 1000 classes

Human performance: ~5%

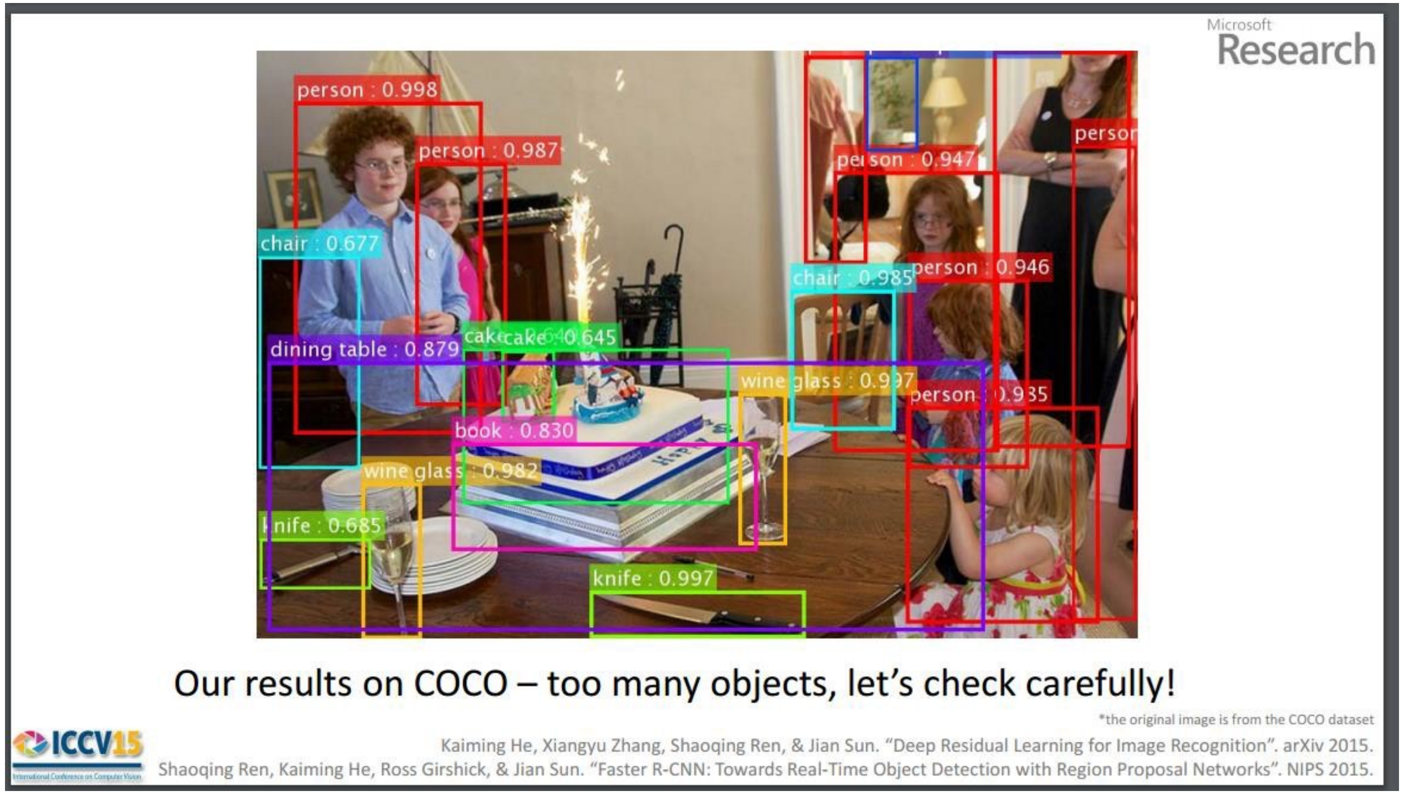




Slides from Kaiming He, MSRA

Image

- Object location



Slides from Kaiming He, MSRA

Image

- Image captioning

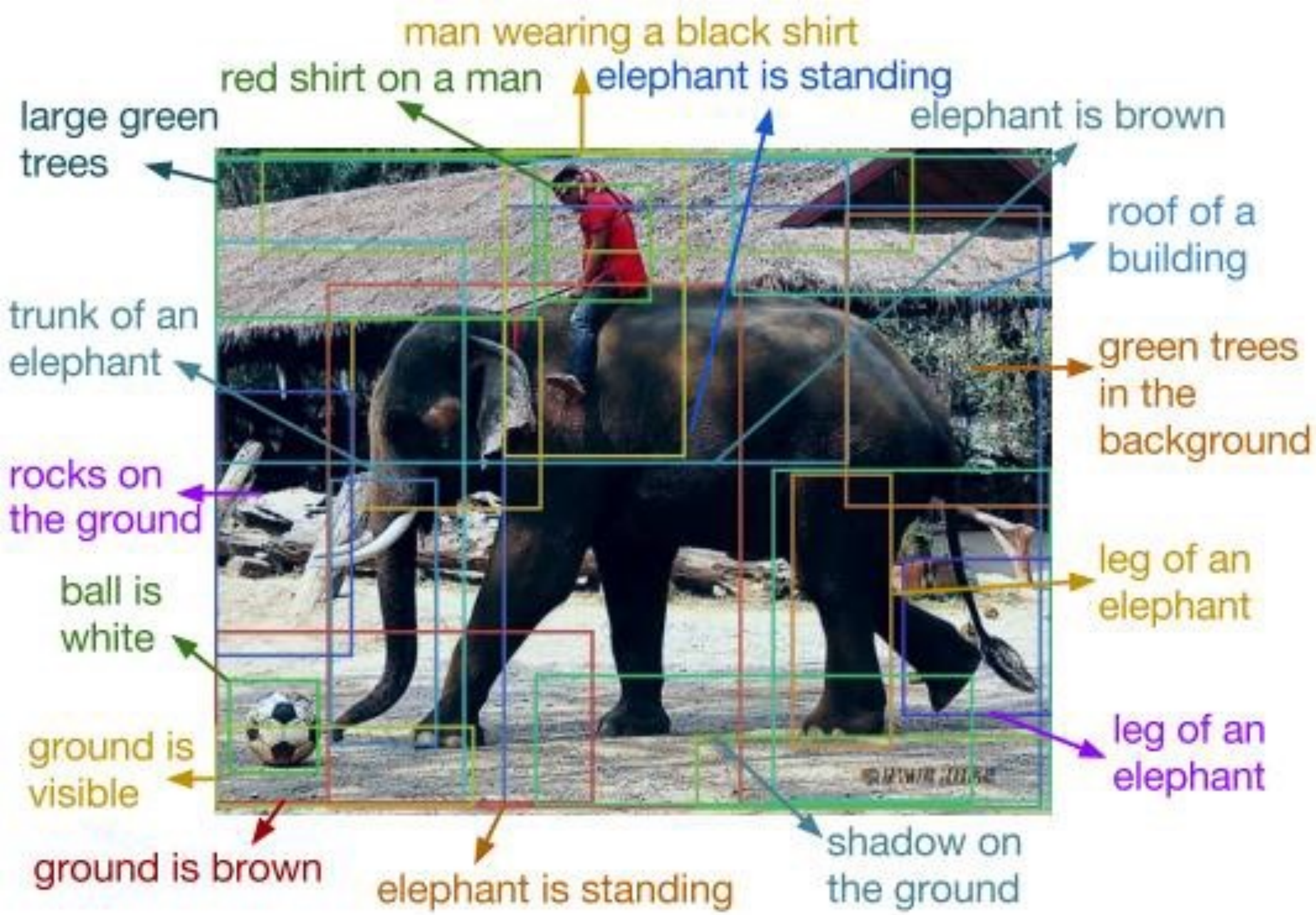


Figure from the paper "DenseCap: Fully Convolutional Localization Networks for Dense Captioning", by Justin Johnson, Andrej Karpathy, Li Fei-Fei

Text

- Question & Answer

I: Jane went to the hallway.
I: Mary walked to the bathroom.
I: Sandra went to the garden.
I: Daniel went back to the garden.
I: Sandra took the milk there.
Q: Where is the milk?
A: garden

I: The answer is far from obvious.
Q: In French?
A: La réponse est loin d’être évidente.

Figures from the paper “Ask Me Anything: Dynamic Memory Networks for Natural Language Processing ”, by Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Richard Socher

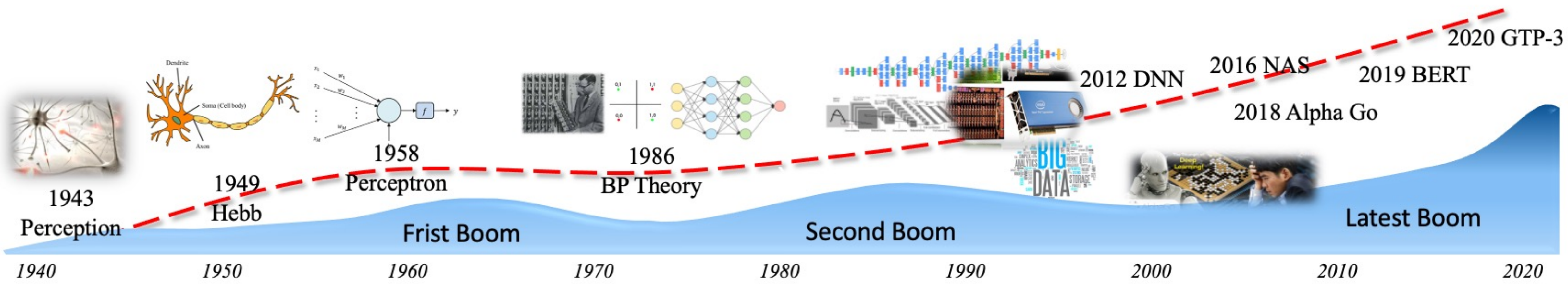
Game

[Google DeepMind's Deep Q-learning playing Atari Breakout]
From the paper “Playing Atari with Deep Reinforcement Learning”,
by Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller

Game


nature
THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE
At last — a computer program that can beat a champion Go player PAGE 484
ALL SYSTEMS GO
CONSERVATION
SONGBIRDS À LA CARTE
Illegal harvest of millions of Mediterranean birds
PAGE 452
RESEARCH ETHICS
SAFEGUARD TRANSPARENCY
Don't let openness backfire on individuals
PAGE 459
POPULAR SCIENCE
WHEN GENES GOT 'SELFISH'
Dawkins's calling card forty years on
PAGE 462
NATURE.COM/NATURE
28 January 2016 £10
Vol. 529, No. 7587

AI Historic Review



Neural Networks date back decades,

Biological Inspiration …modeling the human brain..

At basic level the brain is composed of **neurons**:

- a **neuron** receives input from other **neurons** (thousands) via its synapses
- inputs are approximately summed
- when input exceeds a threshold the **neuron** sends an electrical spike that travels that travels from the body, down the axon, to the next **neuron**(s)..

Learning in the Brain..

Brains learn

Altering strength between **neurons** …creating/deleting connections

Hebb's Postulate (***Hebbian Learning***) 1949

When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.

If neuron x_i repeatedly triggers neuron y, the synaptic knob connecting x_i to y gets larger

mathematical model: $w_i = w_i + \eta x_i y$

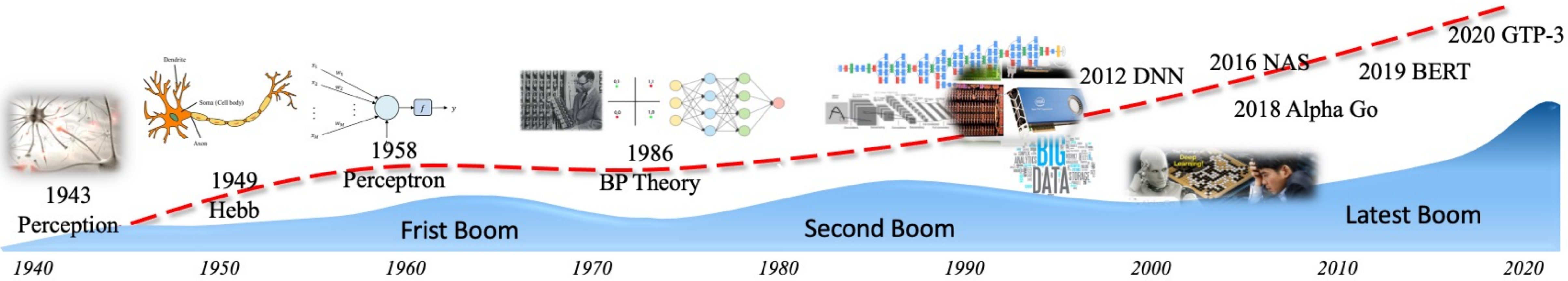
weight of i^{th} neuron's input to output neuron y

unstable....

⇨simple formula is actually the basis of many learning algorithms in machine/deep learning.

The weight between two neurons increases if the two neurons activate simultaneously, and reduces if they activate separately. Nodes that tend to be either both positive or both negative at the same time have strong positive weights, while those that tend to be opposite have strong negative weights.

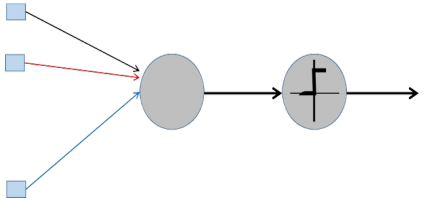
AI Historic Review



Perceptron

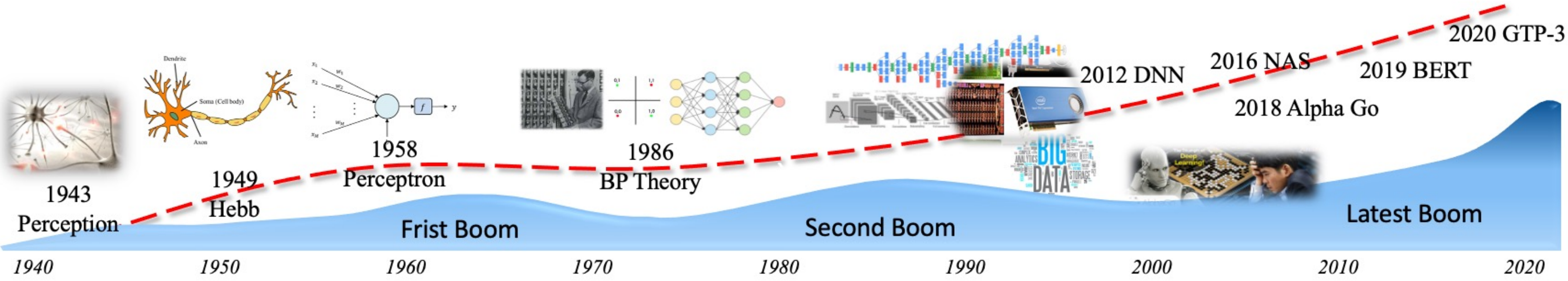
The perceptron (Rosenblatt, 1957) is very similar, except that the inputs are real:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \sum_i w_i x_i + b \geq 0 \\ 0 & \text{otherwise} \end{cases}$$



This model was originally motivated by biology, with w_i being synaptic weights and x_i and f firing rates.

AI Historic Review



Neural Networks date back decades,

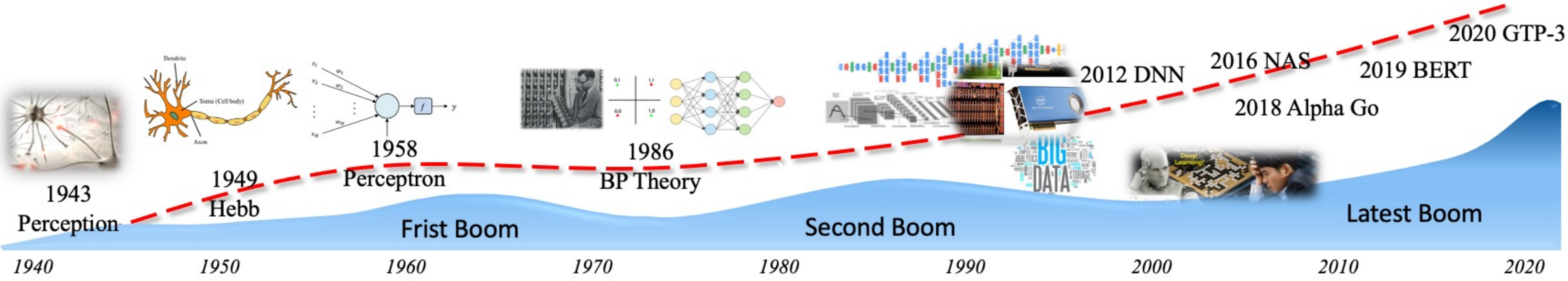
Backpropagation algorithm for learning multiple layers of non-linear features introduced developed over 1970's & 80's

… Werbos, Parker, LeCun, Rumelhart et. al.

Backpropagation held great promise but by 1990's interest

- seemed unable to make good use of multiple hidden layers except in "time- delay" and convolutional nets
- didn't work well in recurrent networks

AI Historic Review



Backpropagation …what went wrong?

Requires labeled training data…

– but almost all data is **unlabeled**… Learning time does not scale well..

– **slow** in networks with multiple hidden layers..

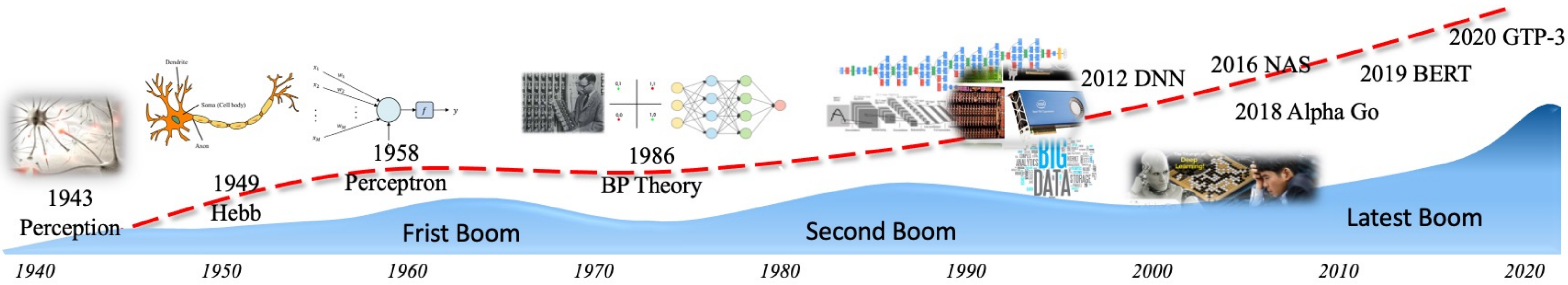
Gets stuck in "**poor local optima**"..

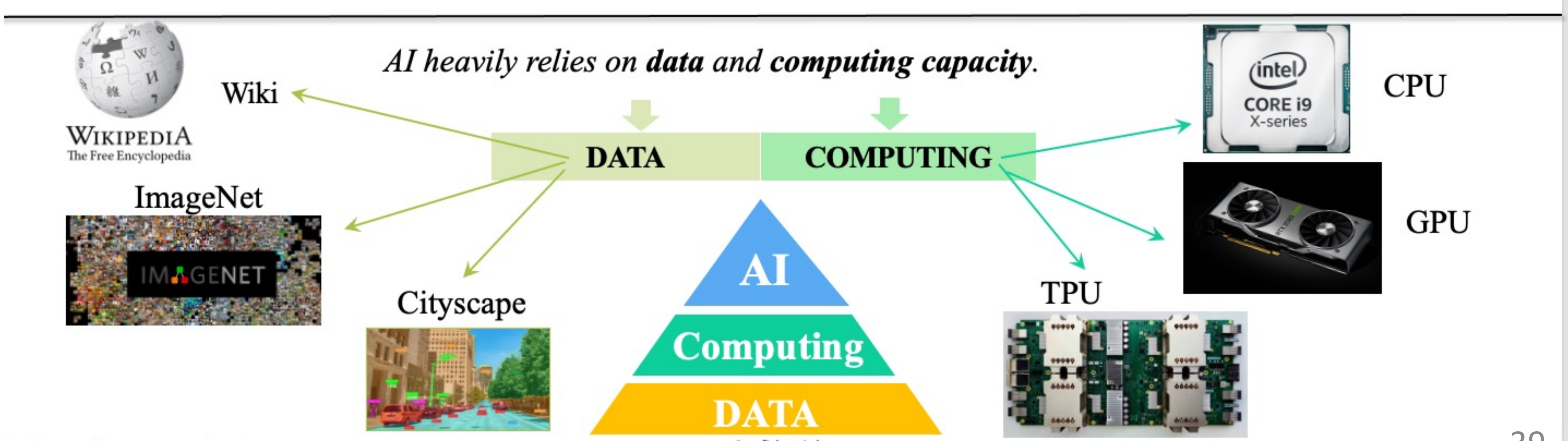
– often “**satisfactory**" but not optimal for Deep Nets

Questions behind the scene

- Return of artificial neural network
  - What's different
  - Why get great performance
- Future development
  - The road to general-purpose AI?

AI Historic Review





*Ref: https://www.eetimes./ais-limits-send-scientists-back-to-the-brain

Deep Learning.. brief history

Now…

1952 Stochastic Gradient Descent

1958 Perceptron
- Learnable Weights

1986 Backpropagation
- Multi-Layer Perceptron

1995 Deep Convolutional NN
- Digit Recognition

2017

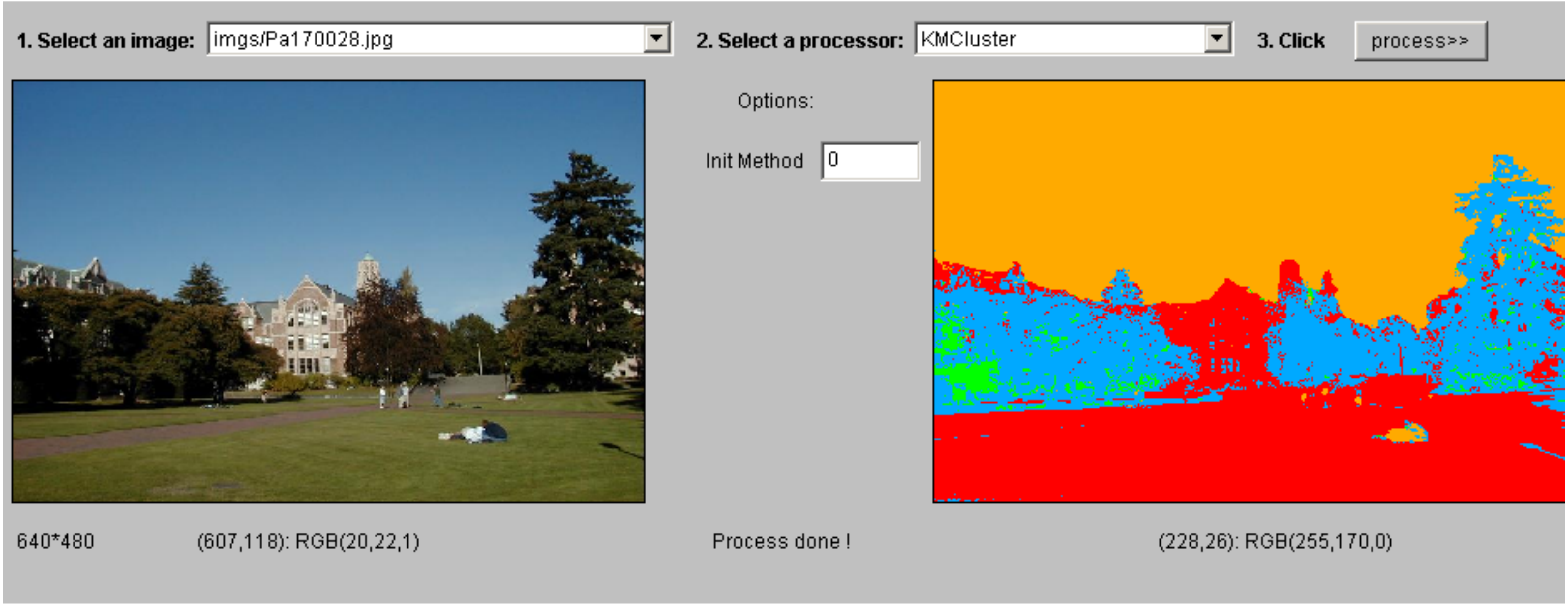
Deep Learning … almost everywhere

- Object classification
- Object detection, segmentation, pose estimation
- Image captioning, question answering
- Machine translation
- Speech recognition
- Robotics

Some strongholds

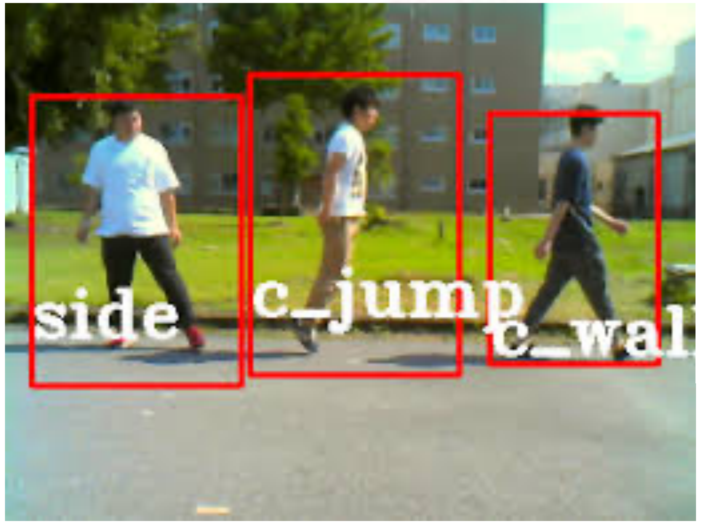
- Action classification, action detection
- Object retrieval
- Object tracking

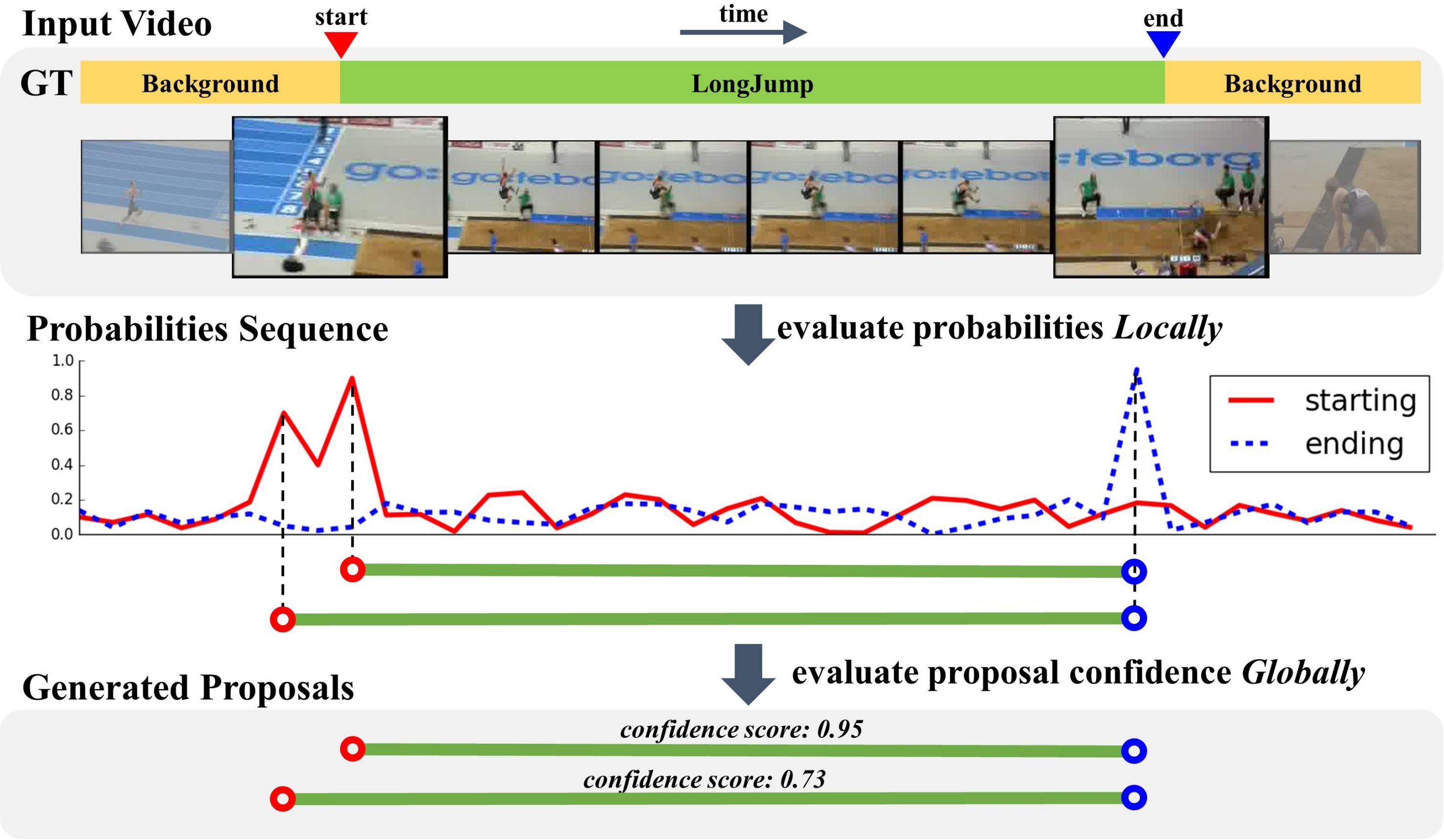
Segmentation



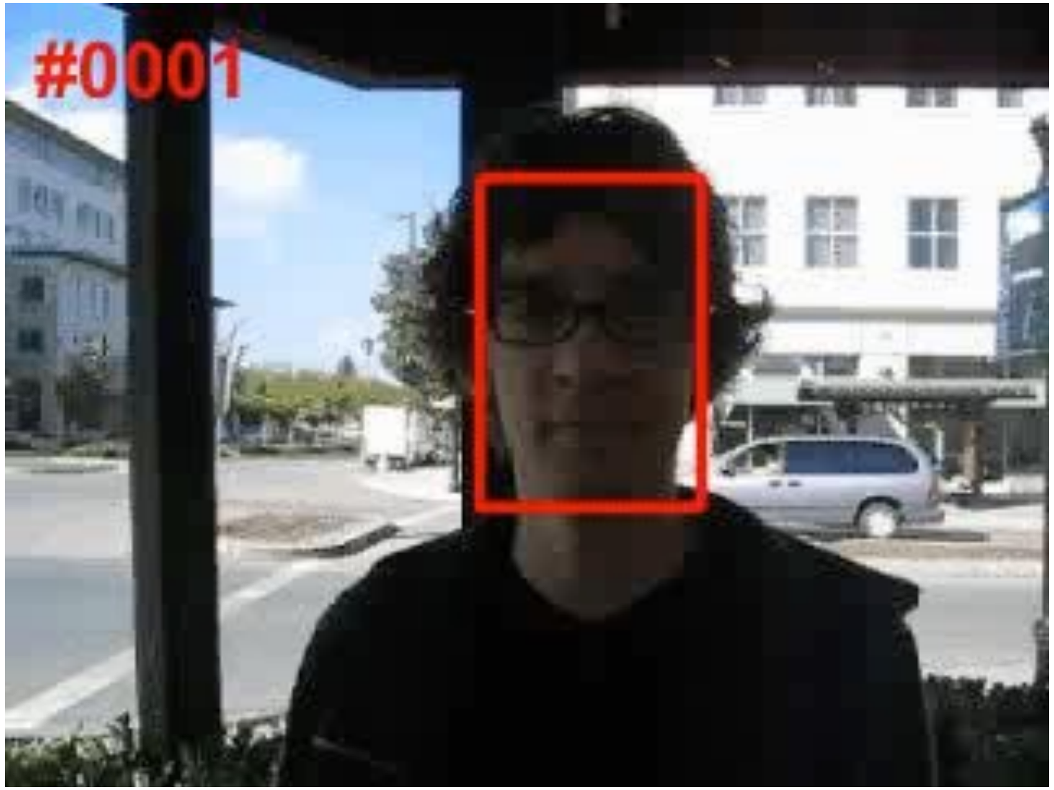
Pose Estimation

Action Detection



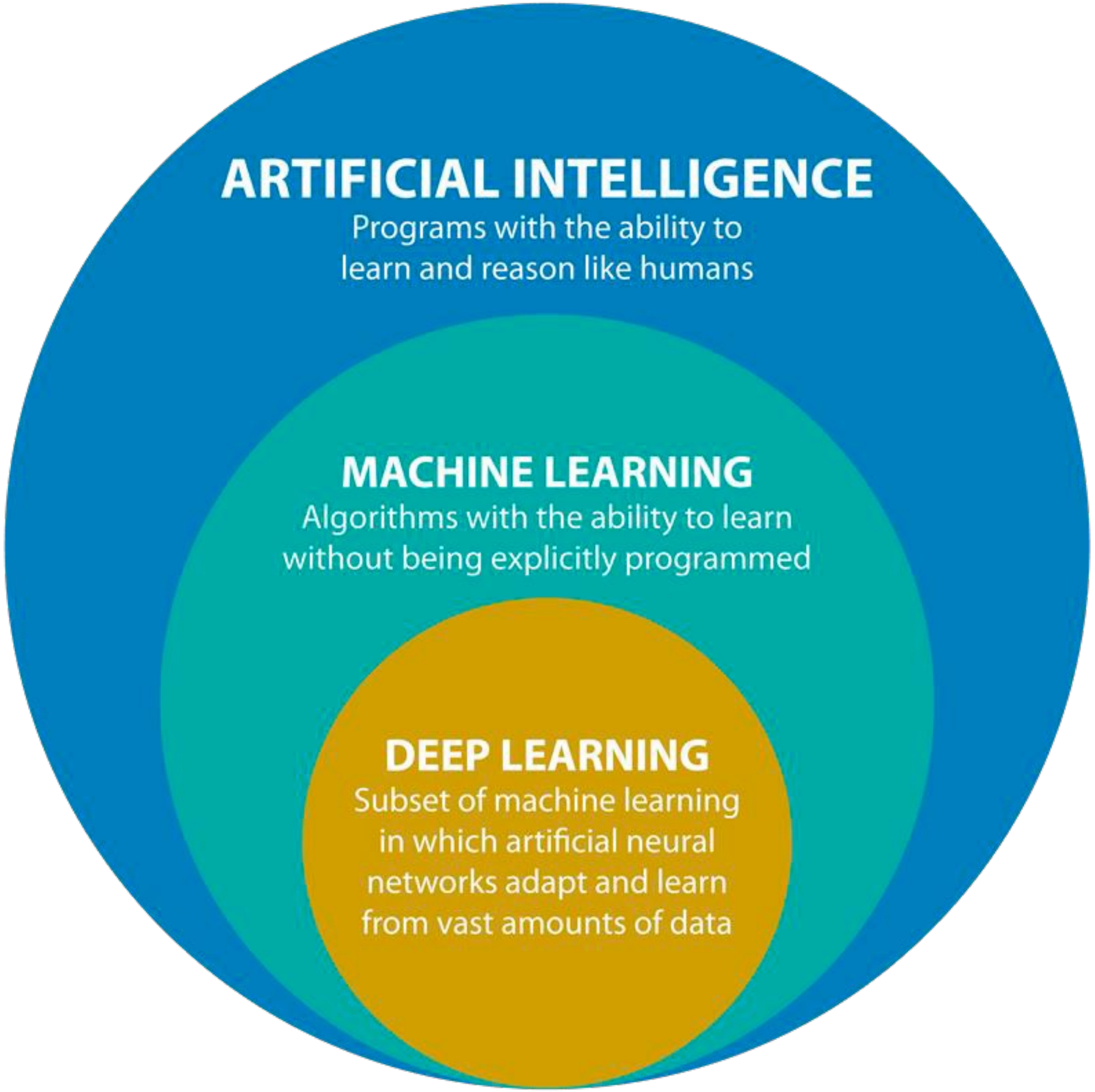


Tracking



Short Break- Come back in 10 mins

What is the difference between DL (Deep Learning), ML (Machine Learning) and AI (Artificial Intelligence)?



Artificial Intelligence is the broader concept of machines being able to carry out tasks in a way that we would consider "smart". And, Machine Learning is a current application of AI based around the idea that we should really just be able to give machines access to data and let them learn for themselves.

What is Machine Learning?

- **Machine learning**
  - ➢ is a subfield of computer science
  - ➢ that is concerned with building algorithms which,
  - ➢ rely on a collection of examples of some phenomenon.
    - ✓ These examples can come from nature, be handcrafted by humans or generated by another algorithm.

Early Definitions

- **Arthur Samuel** (1959): A field of study that gives computers the ability to learn without being explicitly programmed.

- **Tom Mitchell** (1998): A computer program is said to *learn from experience E* with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

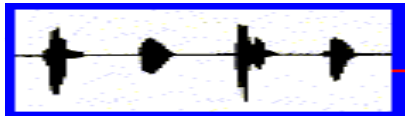
How do we create computer programs that improve with experience?"

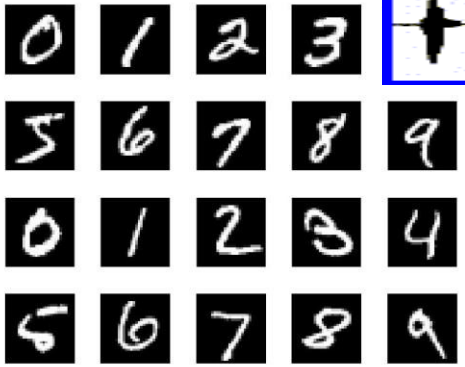
Tom Mitchell

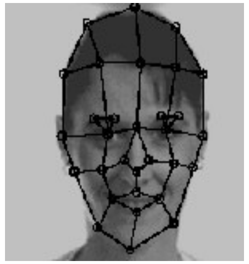
http://videolectures.net/mlas06_mitchell_itm/

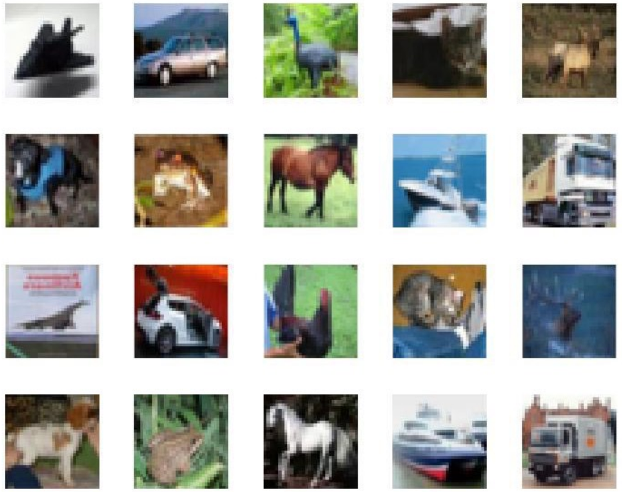
Application Examples

- Speech recognition
- Face recognition
- Handwriting recognition
- Object recognition
- Housing price prediction
- Etc, etc.









Types of Learning

- Supervised Learning
- Unsupervised Learning
- Semi-Supervised Learning
- Reinforcement Learning

	Supervised Learning	Unsupervised Learning
Discrete	Classification	Clustering
Continuous	Regression	Dimensionality Reduction

Supervised Learning

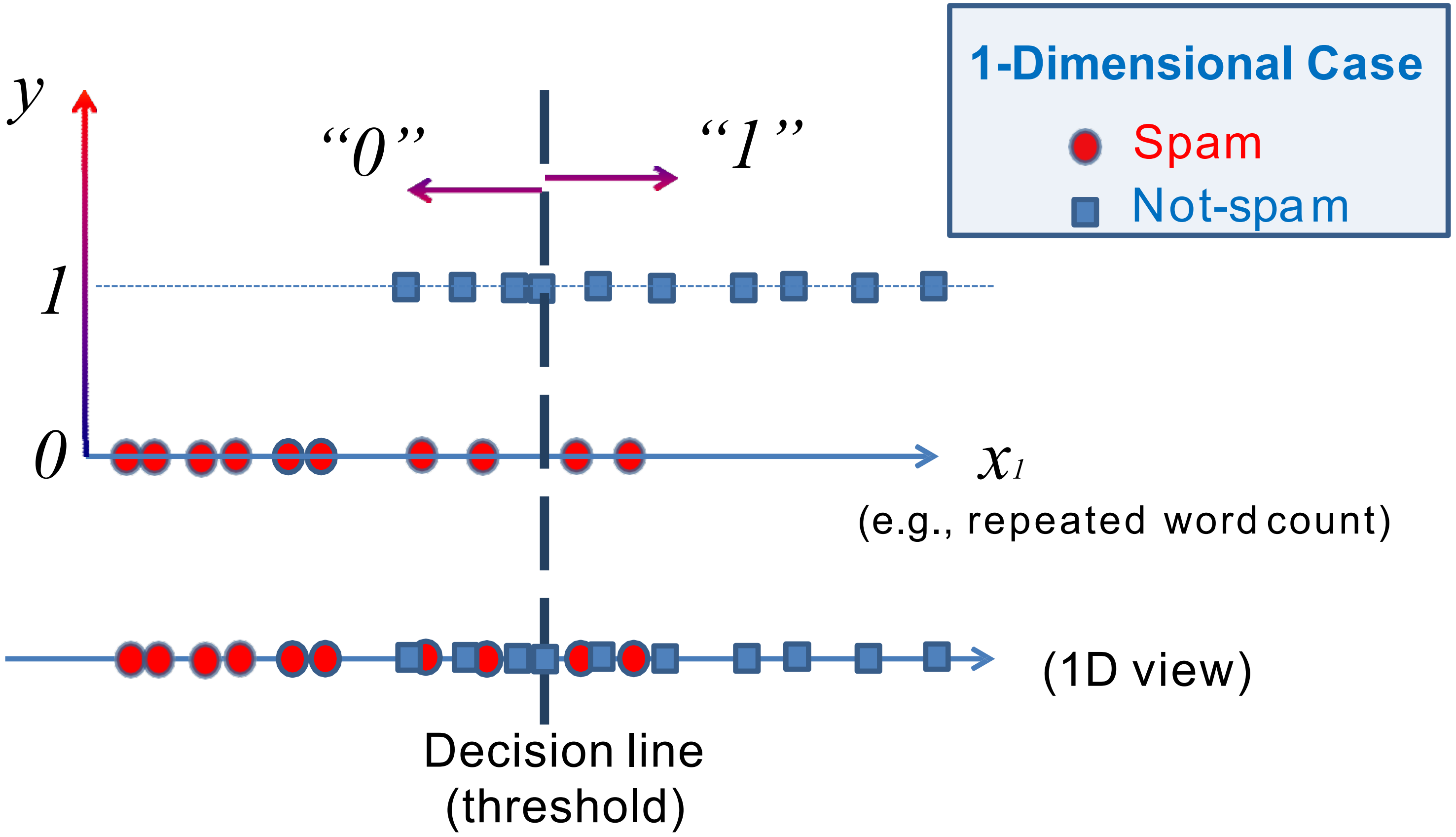
- In **supervised learning**, the **dataset** is the collection of **labeled examples** $\{(\mathbf{x}_i, y_i)\}_{i=1}^{m}$,

$$\mathbf{x}_i = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix}_i \text{ or } \mathbf{x}_i = [x_1, x_2, \ldots, x_d]_i^T, \quad i = 1, \ldots, m$$

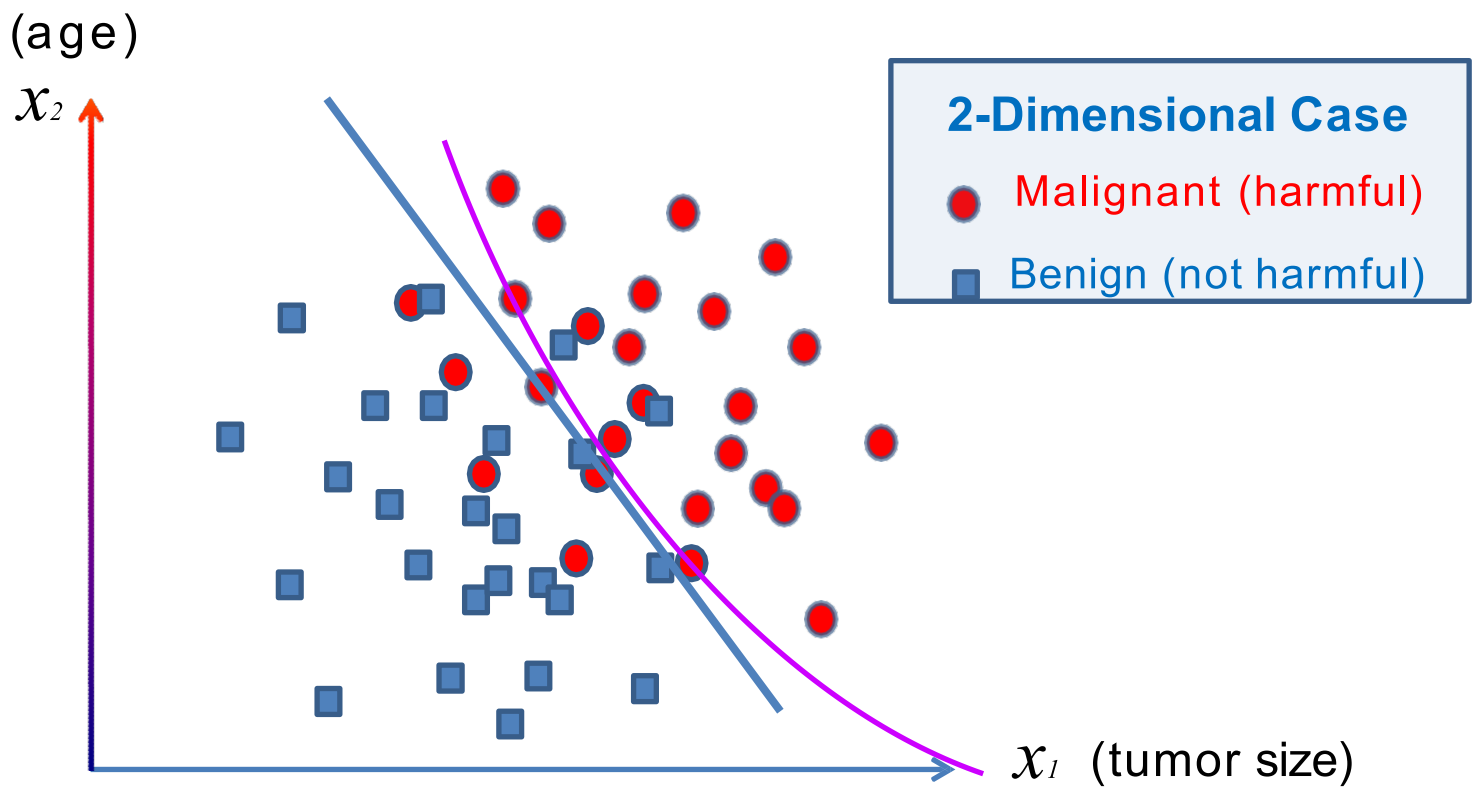
  - Each element $\mathbf{x}_i$ among m is called a **feature vector**.
    - A feature vector is a vector in which each dimension $j = 1, \ldots, d$ contains a value that describes the example somehow.

  - The **label** y_i can be either an element belonging to a finite set of **classes** $\{1, 2, \ldots, C\}$, or a **real** number.

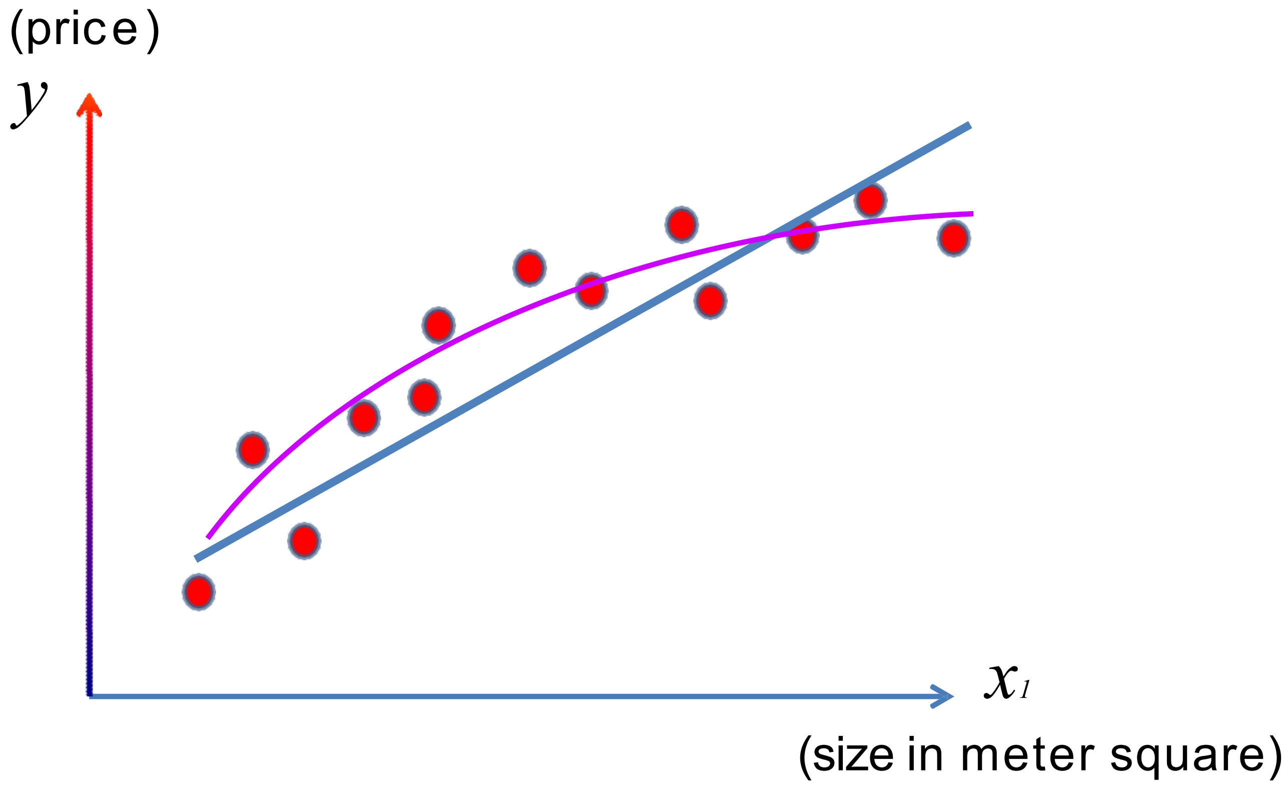
- For instance, if your examples are email messages and your problem is spam detection, then you have two classes {*spam*, *not-spam*}.
- **Classification**: predict discrete valued output (e.g., 0 or 1)



Classification: Breast Cancer (malignant, benign)



Regression: predict continuous valued output (e.g., house price prediction)



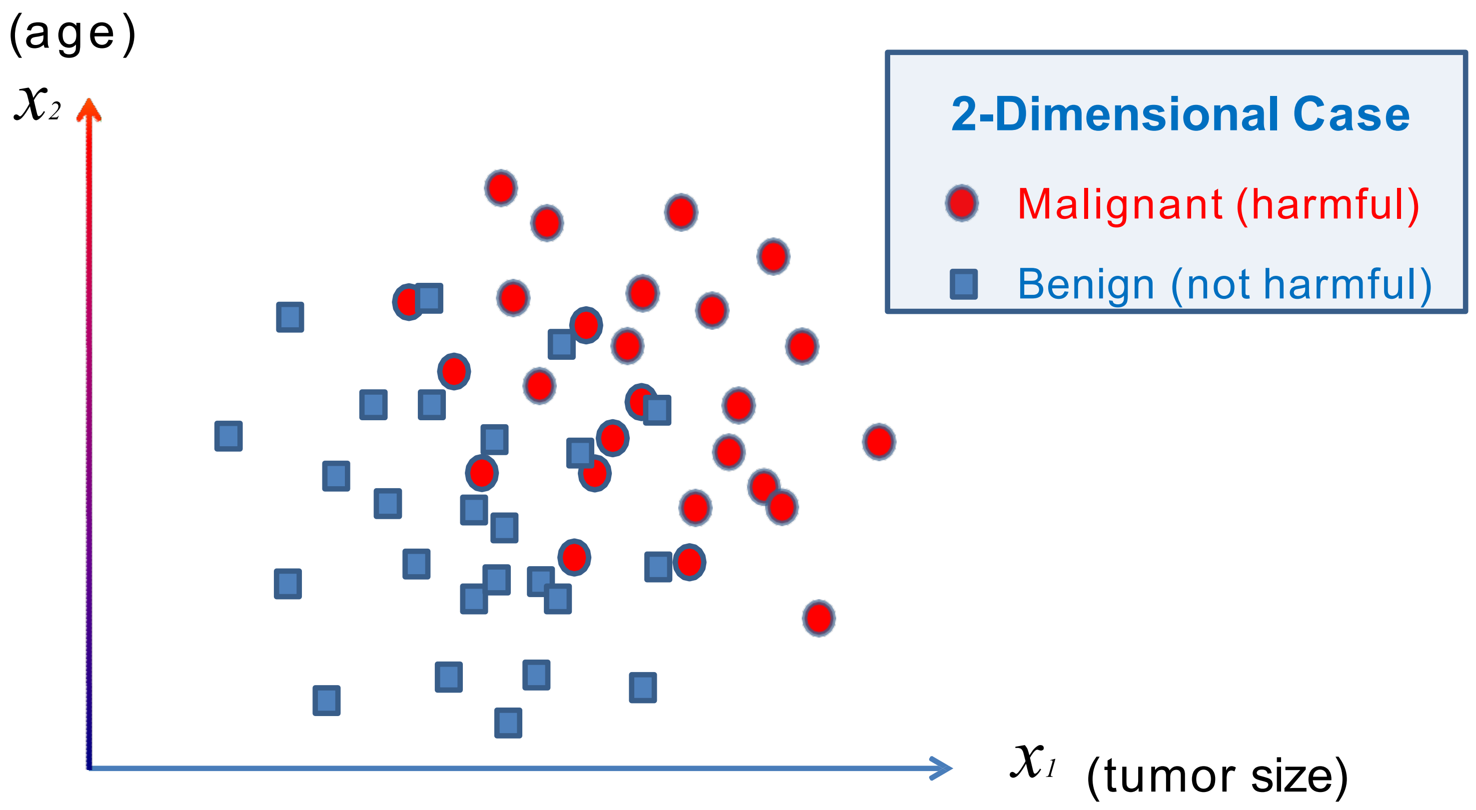
Un-Supervised Learning

- In **unsupervised learning**, the dataset is a collection of **unlabeled examples** $\{\mathbf{x}_i\}_{i=1}^{m}$

- Again, $\mathbf{x}$ is a feature vector, and the goal of an **unsupervised learning algorithm** is to create a **model** that takes a feature vector $\mathbf{x}$ as input and either transforms it into another vector or into a value that can be used to solve a practical problem.

Pictorial summary

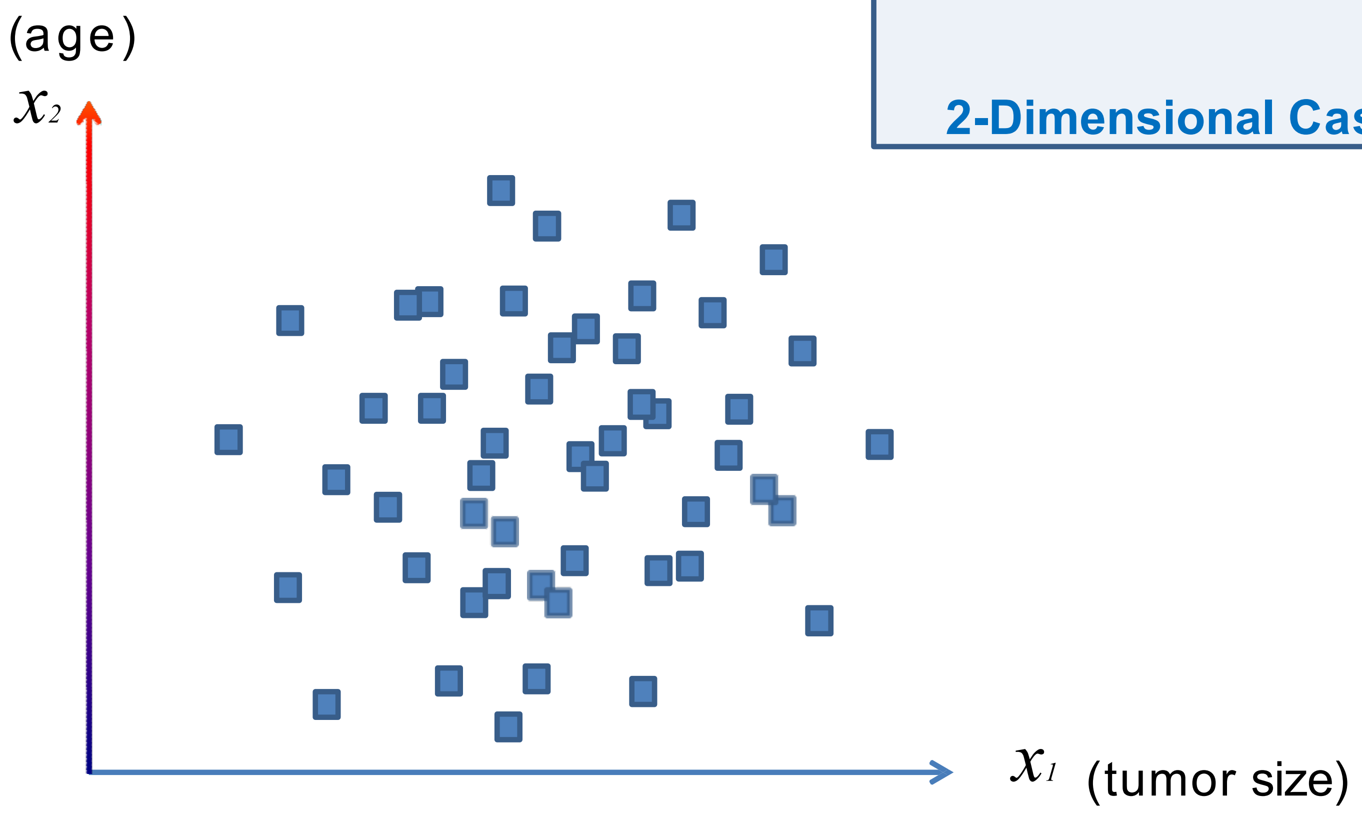
- **Supervised Learning**



Pictorial summary

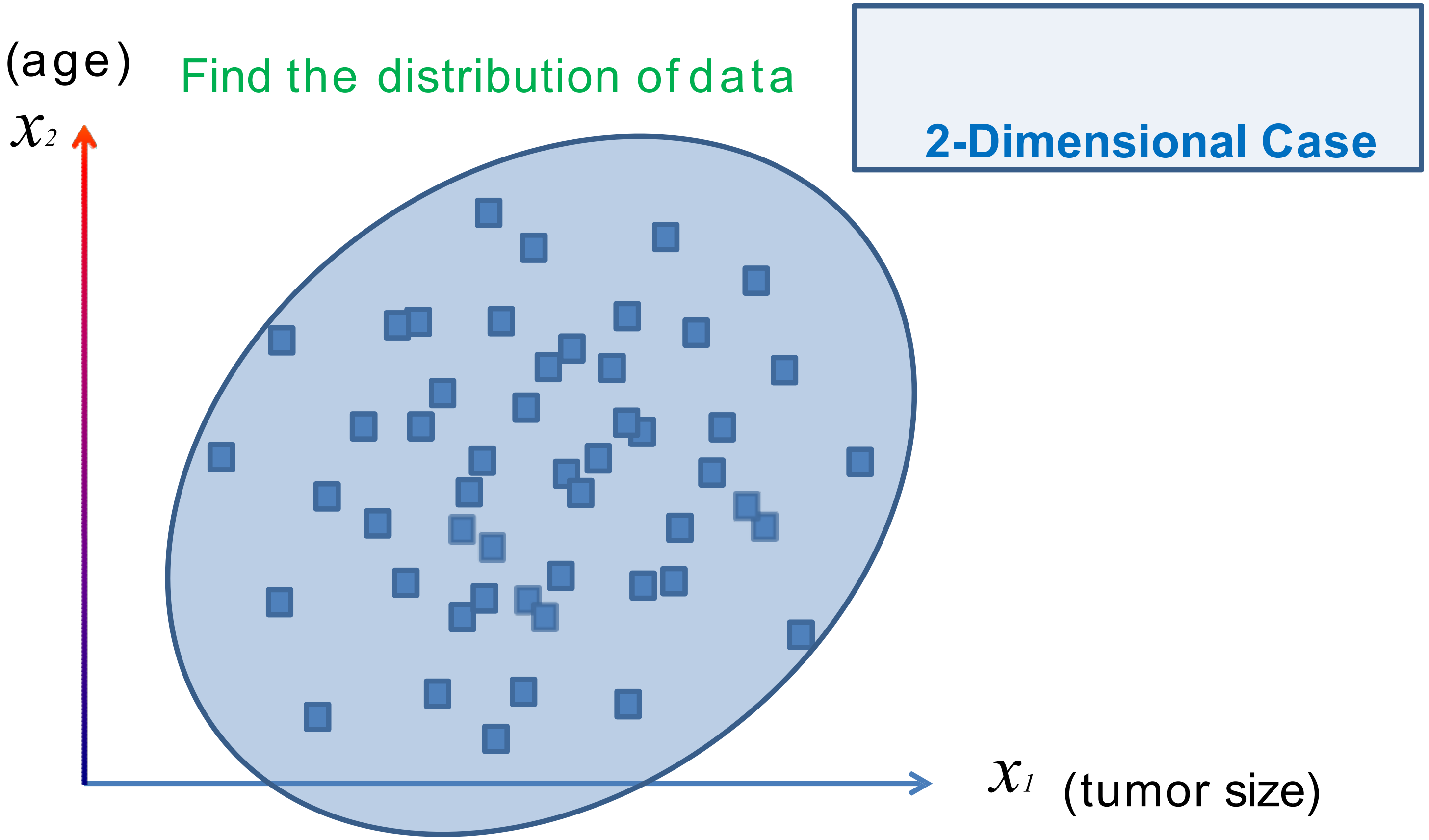
- **Un-Supervised Learning**

2-Dimensional Case



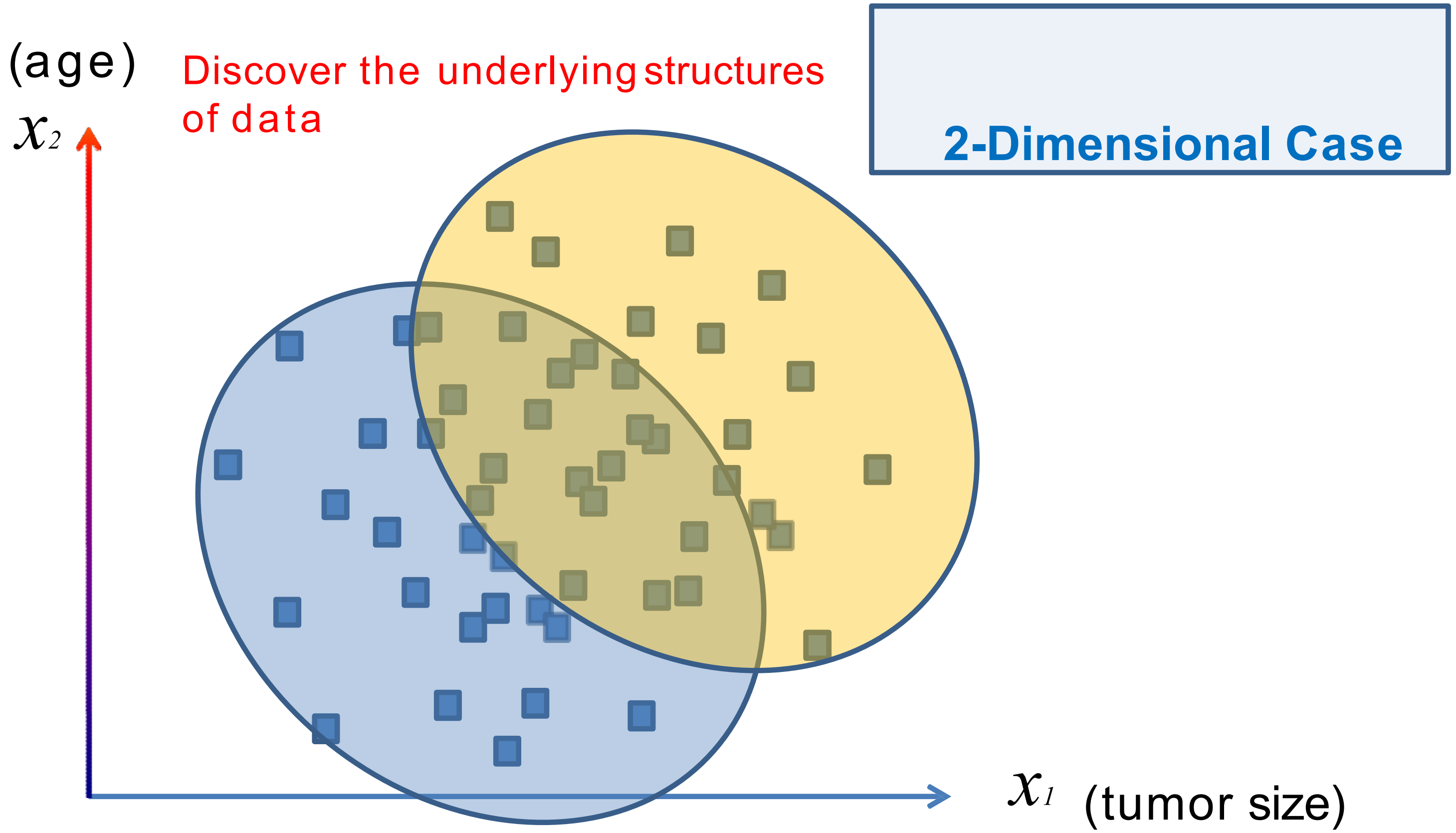
Pictorial summary

- **Un-Supervised Learning**

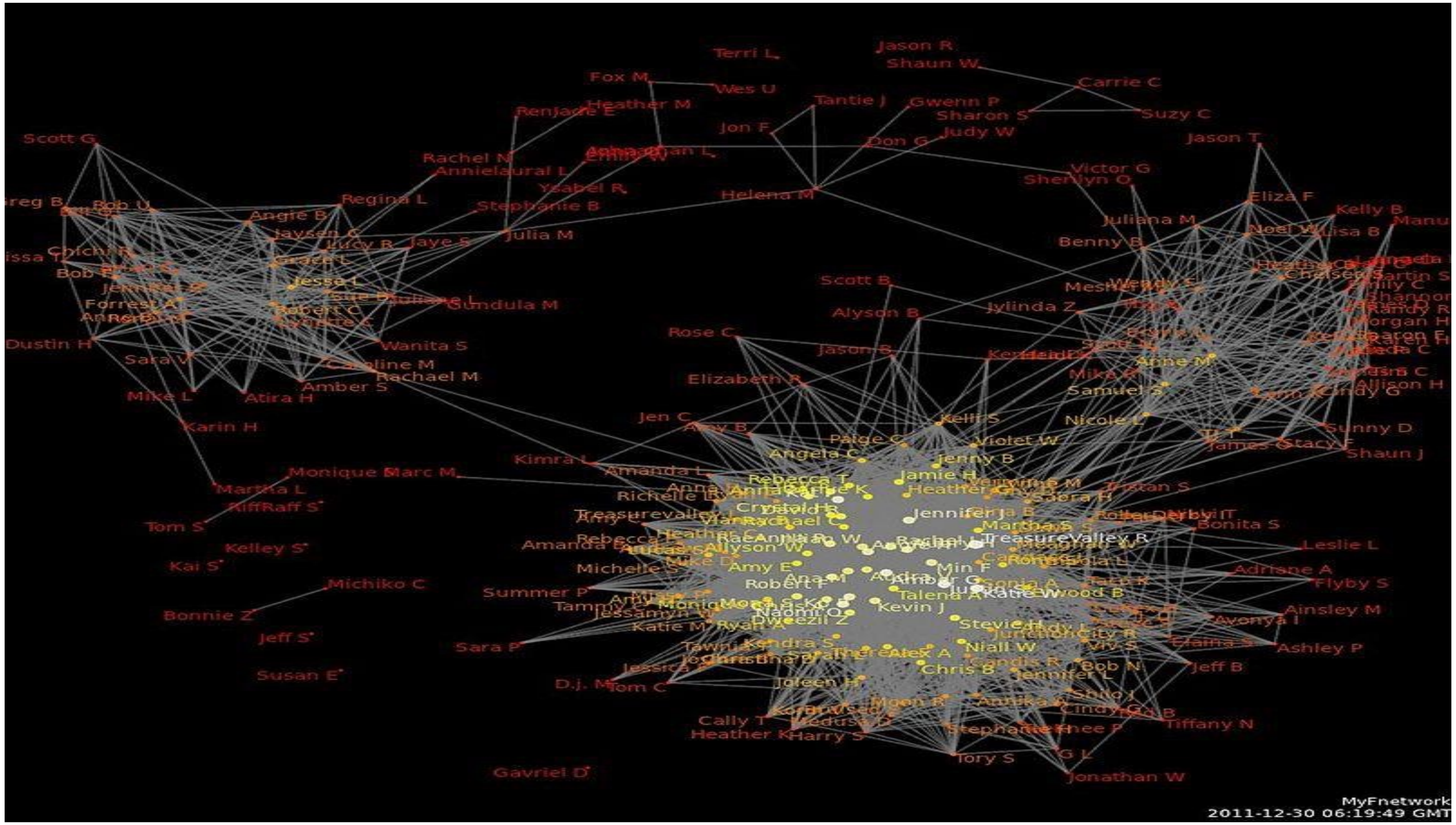


Pictorial summary

- **Un-Supervised Learning: Clustering**

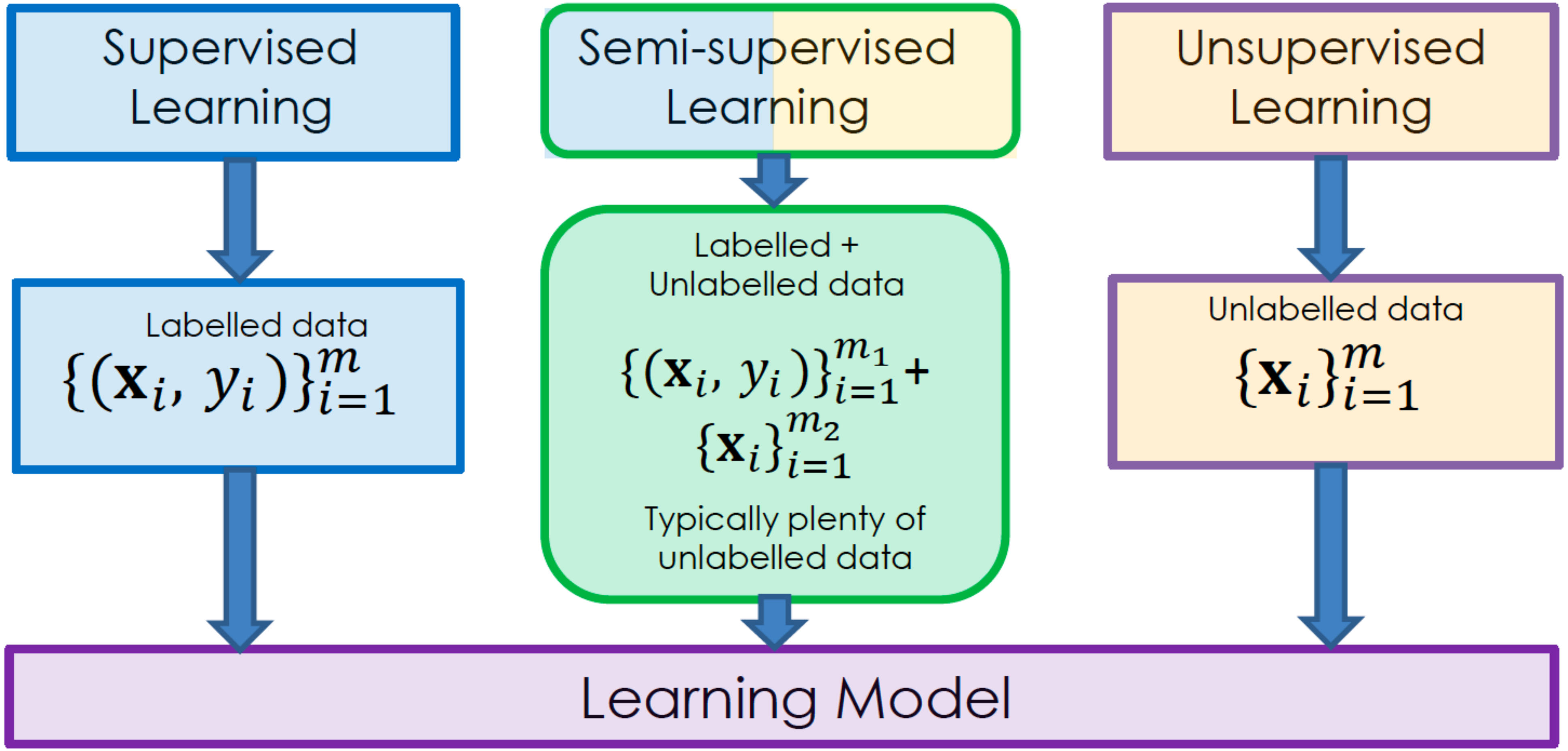


Example: Social Network

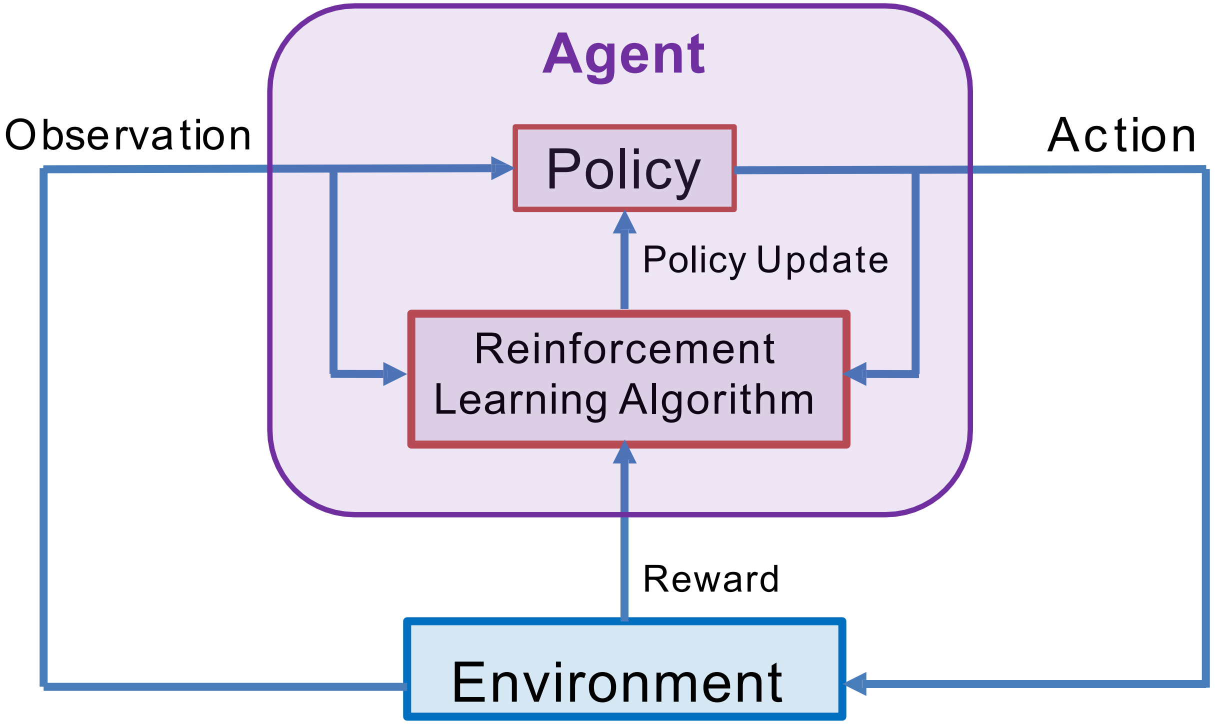


https://en.wikipedia.org/wiki/Social_network_analysis#/media/File:Kencf0618FacebookNetwork.jpg

Semi-Supervised Learning



Reinforcement Learning



Reinforcement Learning

- A **policy** is a function (similar to the model in supervised learning) that takes the feature vector of a state as input and outputs an optimal action to execute in that state.

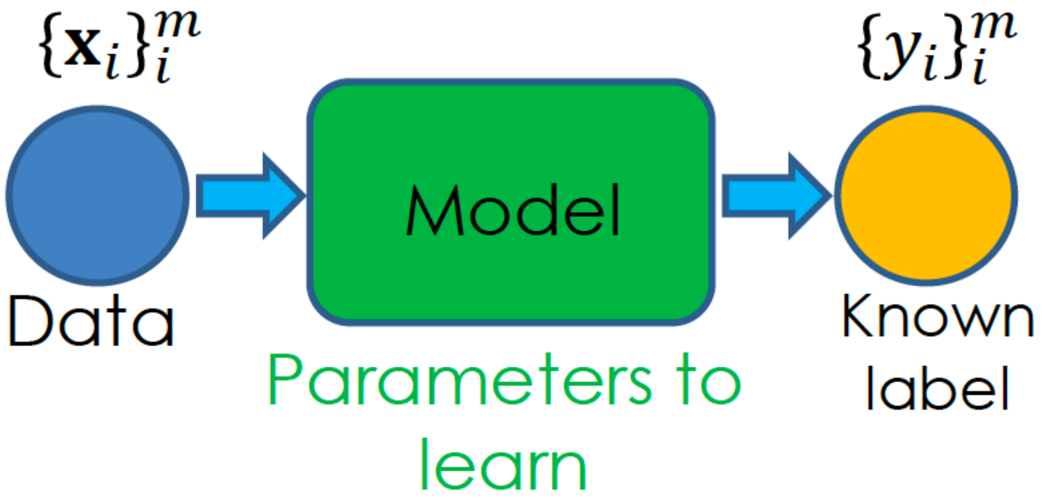
- The **action** is optimal if it maximizes the *expected average reward*.

* Reinforcement learning will detailed in the part III in this course.

How Supervised Learning Works

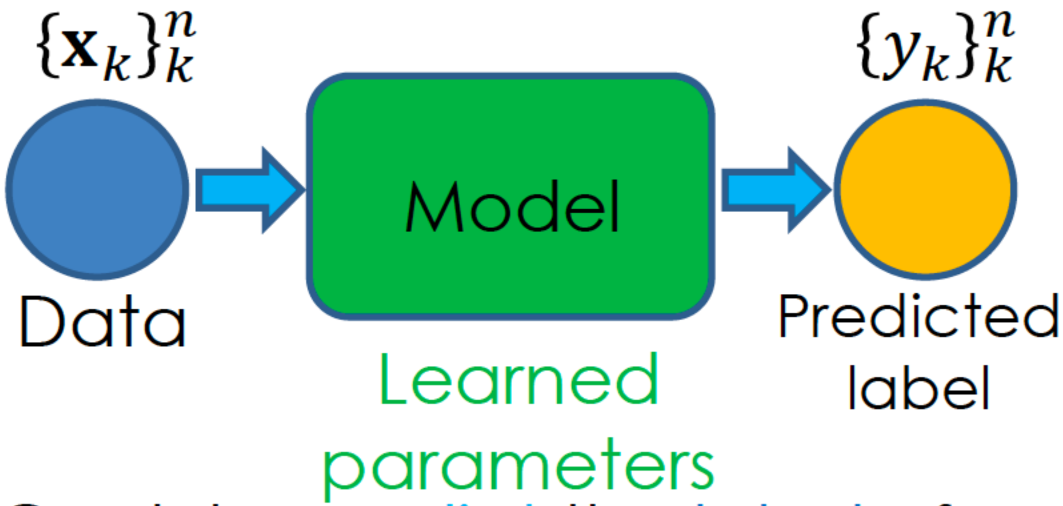
The pipeline:

Training



Goal: to learn the model's parameters from the given data and labels $\{\mathbf{x}_i, y_i\}_i^m$

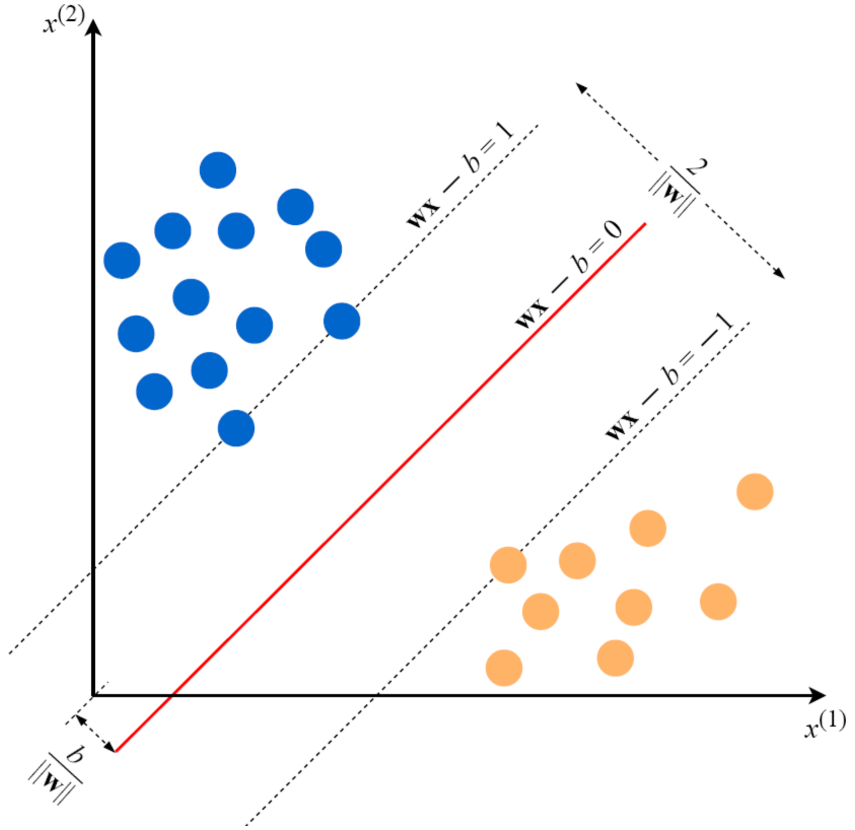
Test



Goal: to predict the label of novel data $\{\mathbf{x}_k\}_k^n$ using the learned parameters

Example:

$$Minimize\ \|\mathbf{w}\|$$
$$subject\ to\ y_i(\mathbf{w}\mathbf{x}_i - b) \geq 1$$
$$for\ i = 1, \ldots, m$$



* Constraint Optimization is out of scope of this course.

Inductive vs. Deductive Reasoning

Main task of Machine Learning: **to make inferences**

Type of inferences

Example

Inductive

- To reach **probable conclusions**.
- All needed information is unavailable or unknown, causing uncertainty in the conclusions

Deductive

- To reach **logical conclusions** deterministically: all information that can lead to the correct conclusion is available

Probability and Statistics are required → Statistical Machine Learning (as opposed to logic machine learning)

Basic properties

- $0 \le p(x) \le 1$
- $\int p(x) \mathrm{d}x = 1$
- $\sum_x p(x) = 1$

Basic rules

Product rule

- Independent variables:
  $p(a,b) = p(a)p(b)$
- Dependent variables:
- $p(a,b) = p(a|b)\,p(b)$
  $= p(b|a)\,p(a)$

Sum rule

- Dependent variables:
  $p(a) = \sum_b p(a,b)$
  (Marginalization)

Slides courtesy: Professor Robby Tan

Inductive Reasoning

Note: humans use inductive reasoning all the time and not in a formal way like using probability/statistics.

A cartoon comment on inductive reasoning

Ref: Gardener, Martin (March 1979). "MATHEMATICAL GAMES: On the fabric of inductive logic, and some probability paradoxes" (PDF). *Scientific American*. **234**

Determine whether each of the following is “inductive” or “deductive” reasoning?

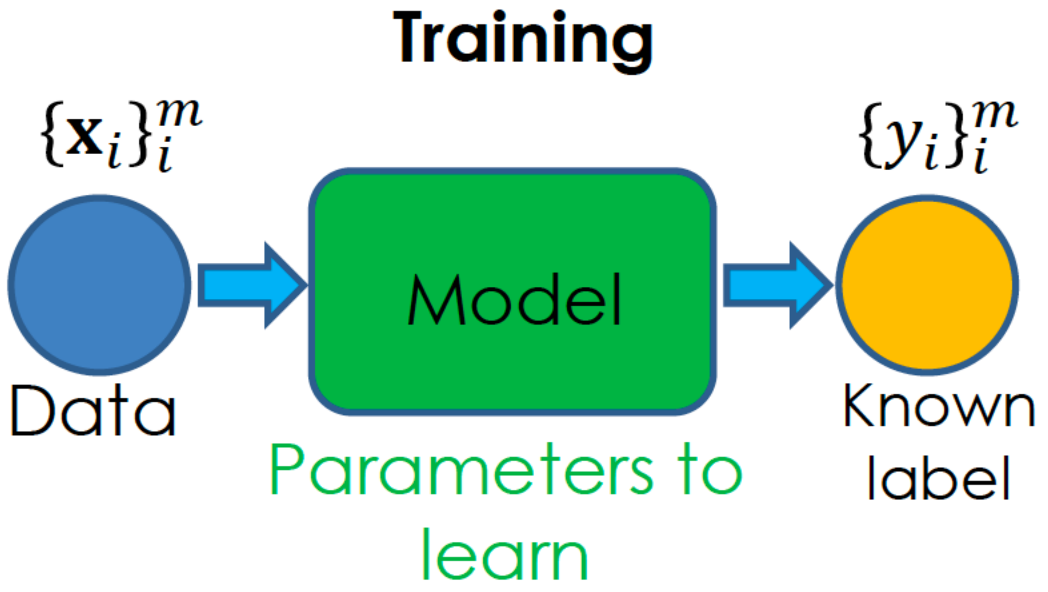
- (a) The first coin I pulled from the bag is a penny. The second and the third coins from the bag are also pennies. Therefore, all the coins in the bag are pennies.

- (b) All men are mortal. Harold is a man. Therefore, Harold is mortal.

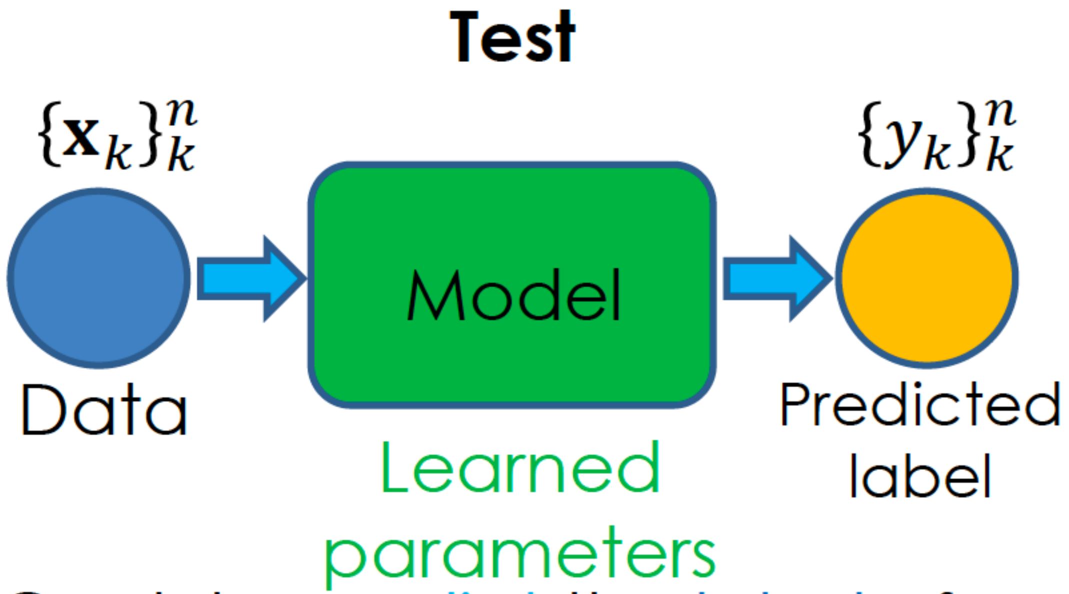
How Supervised Learning Works

What is the underlying Assumption?

The pipeline:



Goal: to learn the model's parameters from the given data and labels $\{\mathbf{x}_i, y_i\}_i^m$



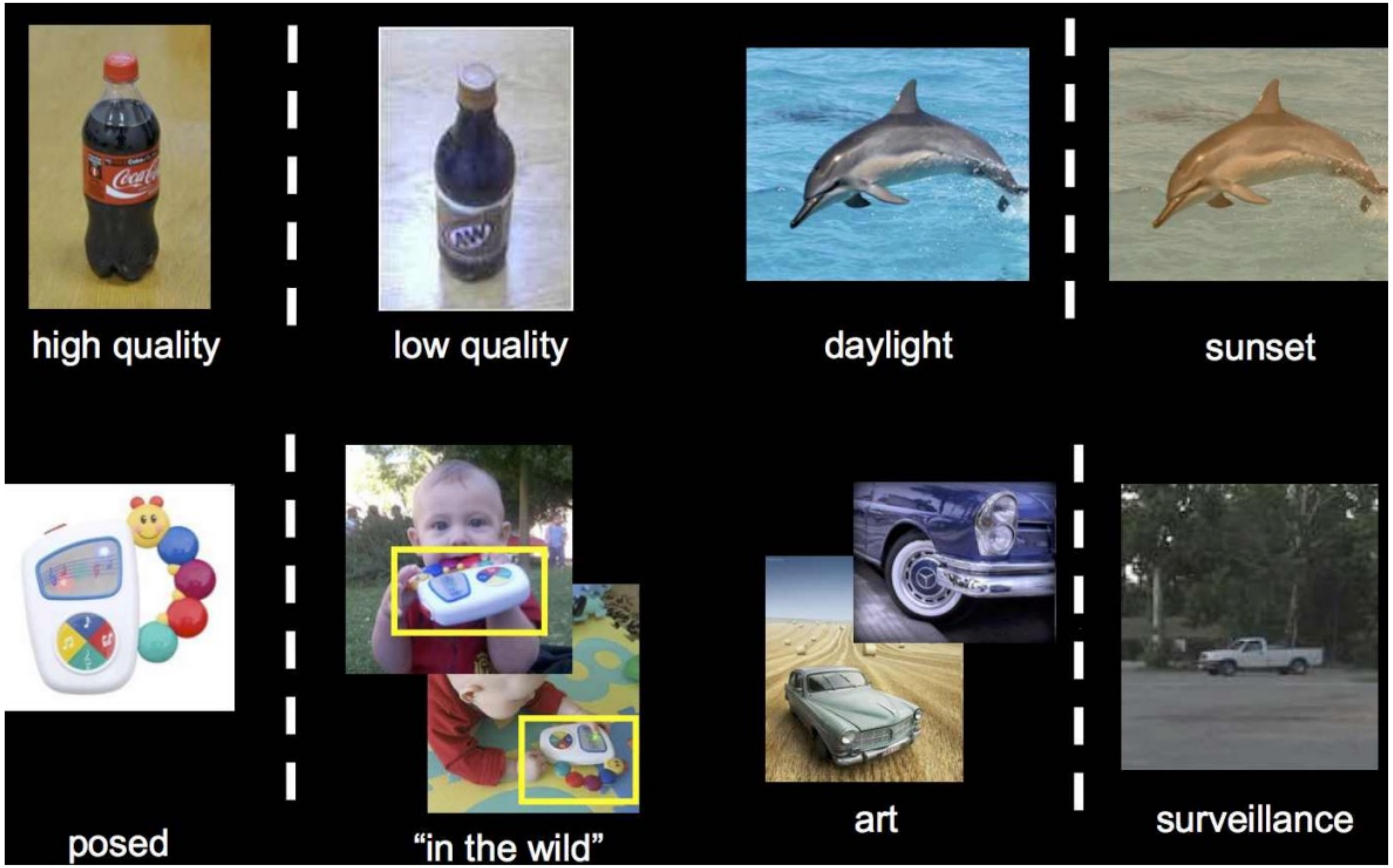
Goal: to predict the label of novel data $\{\mathbf{x}_k\}_k^n$ using the learned parameters

Transfer Learning

Real-life challenges

1. Deep learning methods are data-hungry
2. >50K data items needed for training
3. The distributions of the source and target data must be the same
4. Labeled data in the target domain may be limited
5. This problem is typically addressed with **transfer learning**

Hard to predict what will change in the new domain



[Xu,Saenko,Tsang, Domain Transfer Tutorial - CVPR'12]

Natural Language Processing

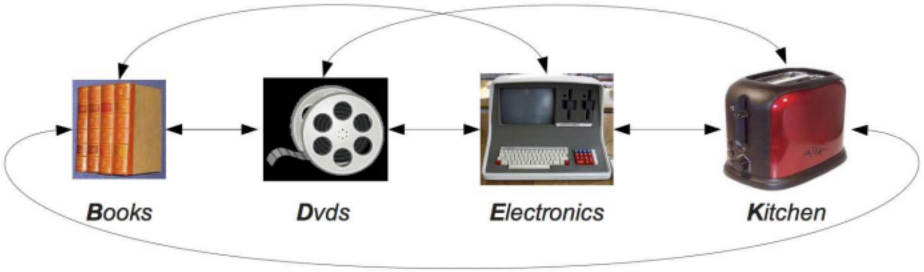
Text are represented by "words" (Bag of Words)

- Part of Speech Tagging: Adapt a tagger learned from medical papers to a journal (Wall Street Journal) - Newsgroup

Biomedical	WSJ
the **signal** *required* to	**investment** *required*
stimulatory **signal** *from*	**buyouts** *from* buyers
essential **signal** *for*	to **jail** *for* violating

- Spam detection: Adapt a classifier from one mailbox to another

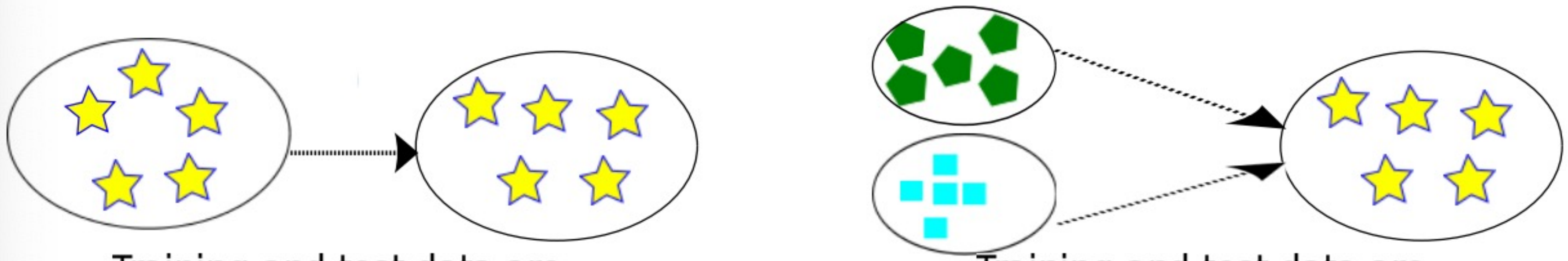
- Sentiment analysis:



Domain Adaptation for sentiment analysis - ex [Pan-IJCAI'13 tutorial]

	Electronics	**Video games**
	(1) Compact; easy to operate; very good picture quality; looks sharp!	(2) A very good game! It is action packed and full of excitement. I am very much hooked on this game.
	(3) I purchased this unit from Circuit City and I was very excited about the quality of the picture. It is really nice and sharp.	(4) Very realistic shooting action and good plots. We played this and were hooked.
	(5) It is also quite blurry in very dark settings. I will never_buy HP again.	(6) It is so boring. I am extremely unhappy and will probably never_buy UbiSoft again.

- Source specific: *compact, sharp, blurry*.
- Target specific: *hooked, realistic, boring*.
- Domain independent: *good, excited, nice*, *never_buy, unhappy*.



Training and test data are from the same domain

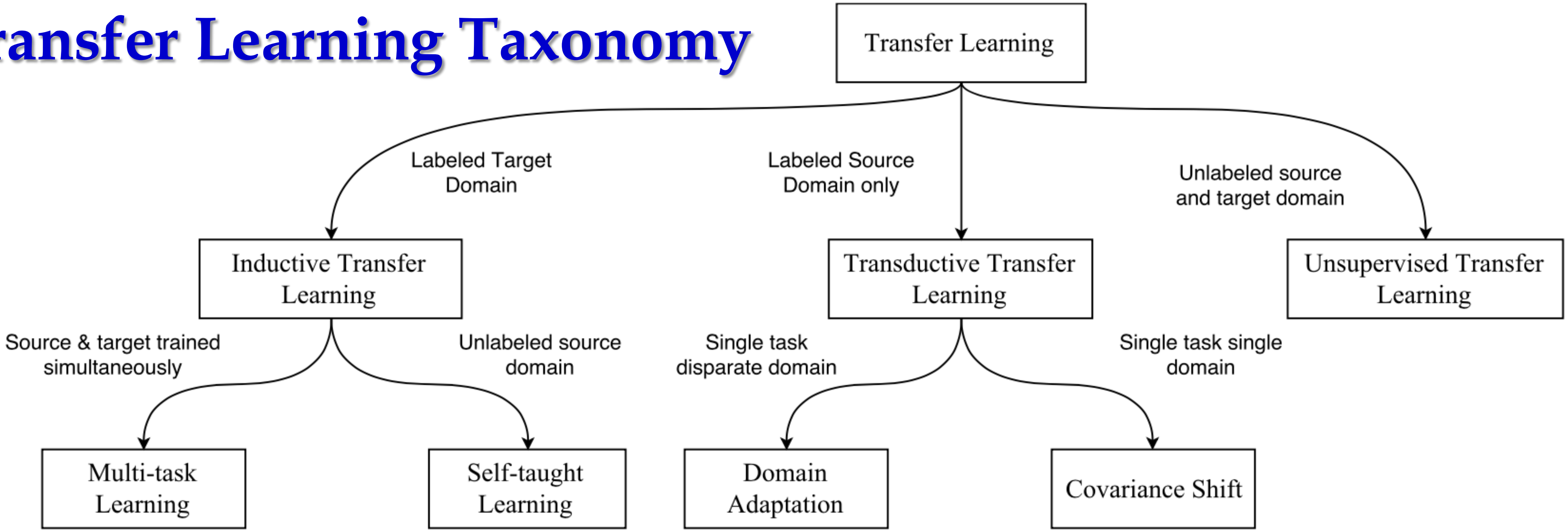
Training and test data are from different domains

Imagenet: 1million

my dataset: 1,000

- Domains are modeled as probability distributions over an instance space
- Tasks associated to a domain (classification, regression, clustering, ...)
- **Objective:** From source to target
  $\Rightarrow$ Improve a target predictive function in the target domain using knowledge from the source domain

Transfer Learning Taxonomy



Transductive transfer

- No labeled target domain data available
- Focus of most transfer research
- E.g. Domain adaptation

Inductive transfer

- Labeled target domain data available
- Goal: improve performance on the target task by training on other task(s)
  - Jointly training on >1 task (multi-task learning)
  - Pre-training (e.g. word embeddings)

Source: https://arxiv.org/pdf/1802.05934.pdf

Transfer Learning .. questions to consider

What to transfer

- Instances?
- Model?
- Features?

How to transfer

- Weight instances
- Unify features
- Map model

When to transfer .. In which situations?

- Faster to transfer or to retrain?

Transfer Learning .. ConvNets are good ..

Even if dataset $\mathcal{T}$ is not large, *can train a CNN for it!!!*

First pre-train a network on dataset $\mathcal{S}$.

Approaches:

fine-tuning..

CNN as feature extractor

Transfer Learning .. using h_s initialization

Assume parameters of $\mathcal{S}$ are a good start, near final local optimum.

Use them as initial parameters for the new **CNN** for target dataset.

$$w^{t=0}{}_{\mathcal{T},l} = w_{\mathcal{S},l} \quad \text{for some layers } l = 1, 2, \ldots$$

This is a good solution when dataset $\mathcal{T}$ is relatively small..

e.g. for ImageNet $\mathcal{S}$ with approximately 1 million images and
a dataset $\mathcal{T}$ with more than a few thousand images →should be OK

what layers to initialize and how?

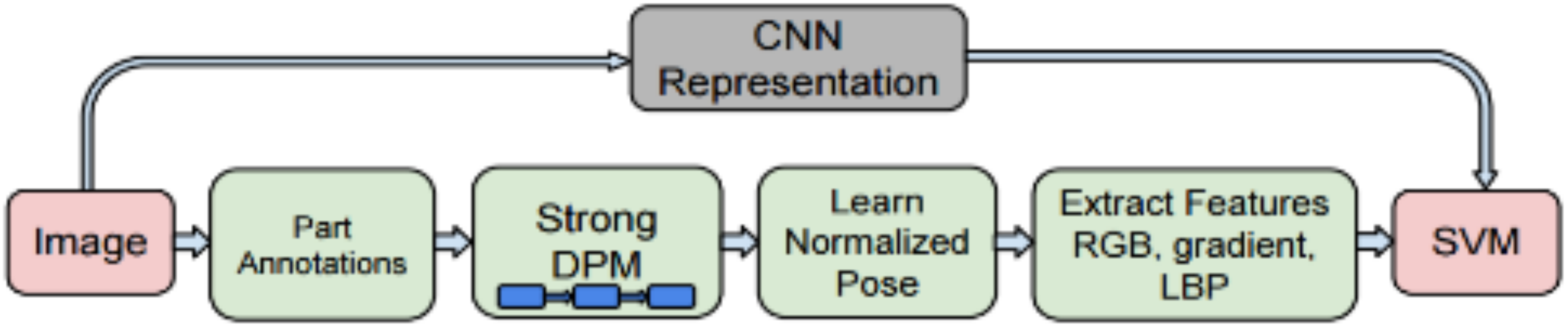
When to finetune your model?

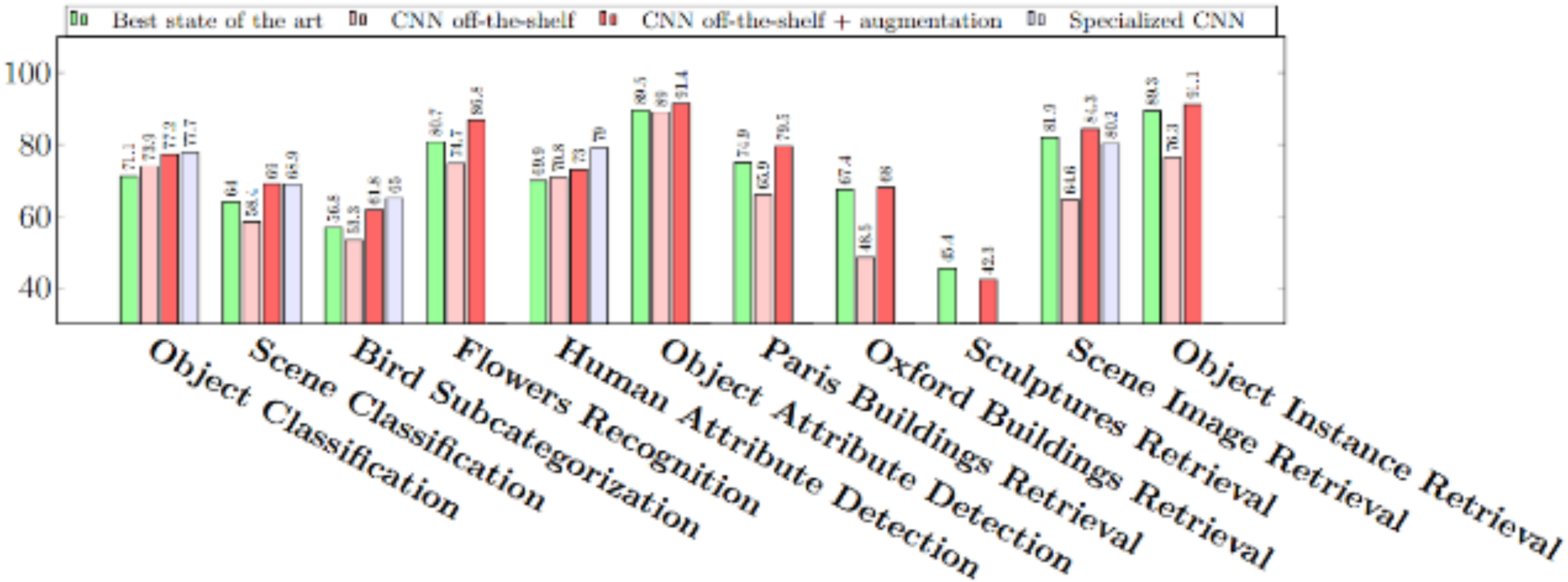
New dataset is small and similar to original dataset.
  train a linear classifier on the CNN codes

New dataset is large and similar to the original dataset
  fine-tune through the full network

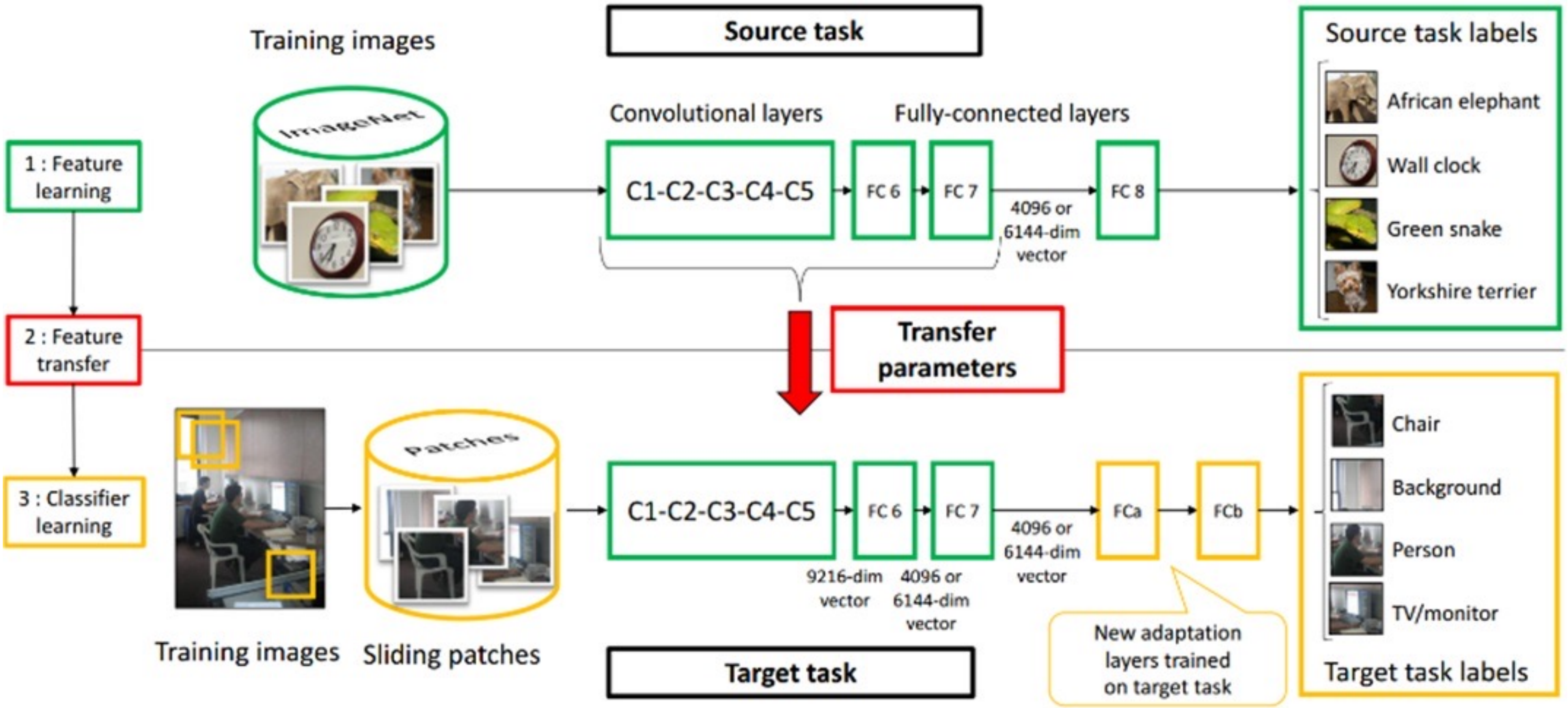
New dataset is small but very different from the original dataset
  SVM classifier from activations somewhere earlier in the network

New dataset is large and very different from the original dataset
  fine-tune through the entire network



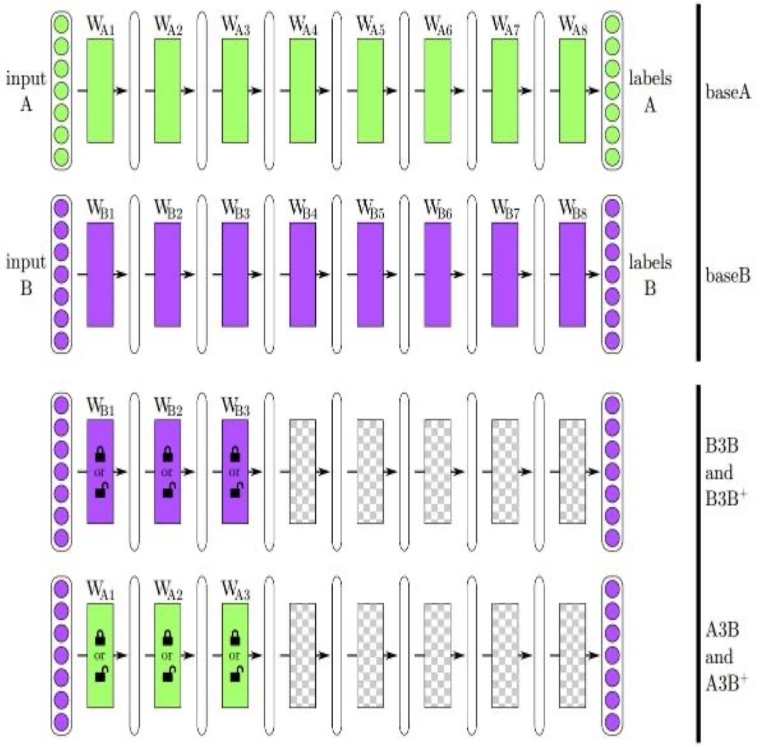


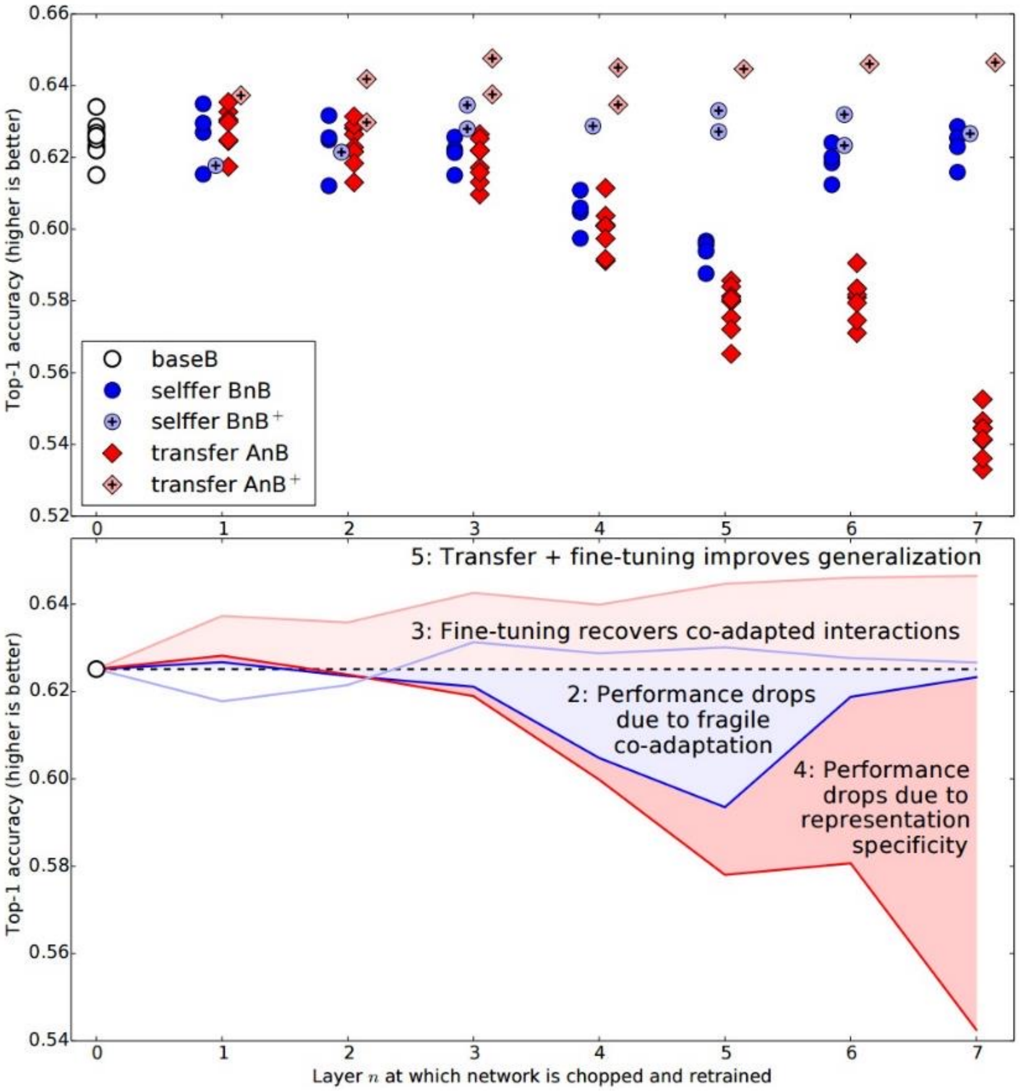
CNN Features off-the-shelf: an Astounding Baseline for Recognition
[Razavian et al. 2014]



Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks [Oquab et al. CVPR 2014]

How transferable are features in CNN?





How transferable are features in deep neural networks [Yosinski NIPS 2014]