# Introduction to Deep Learning

Chia-Po Wei

Department of Electrical Engineering
National Sun Yat-sen University

# Outline

- Timeline
- Supervised Learning
  - Convolutional Neural Networks (CNN)
- Unsupervised Learning
  - Generative Adversarial Networks (GAN)
- Reinforcement Learning

# What is Deep Learning?

### Artificial Intelligence

Any technique that enables computers to mimic human behavior

### Machine Learning

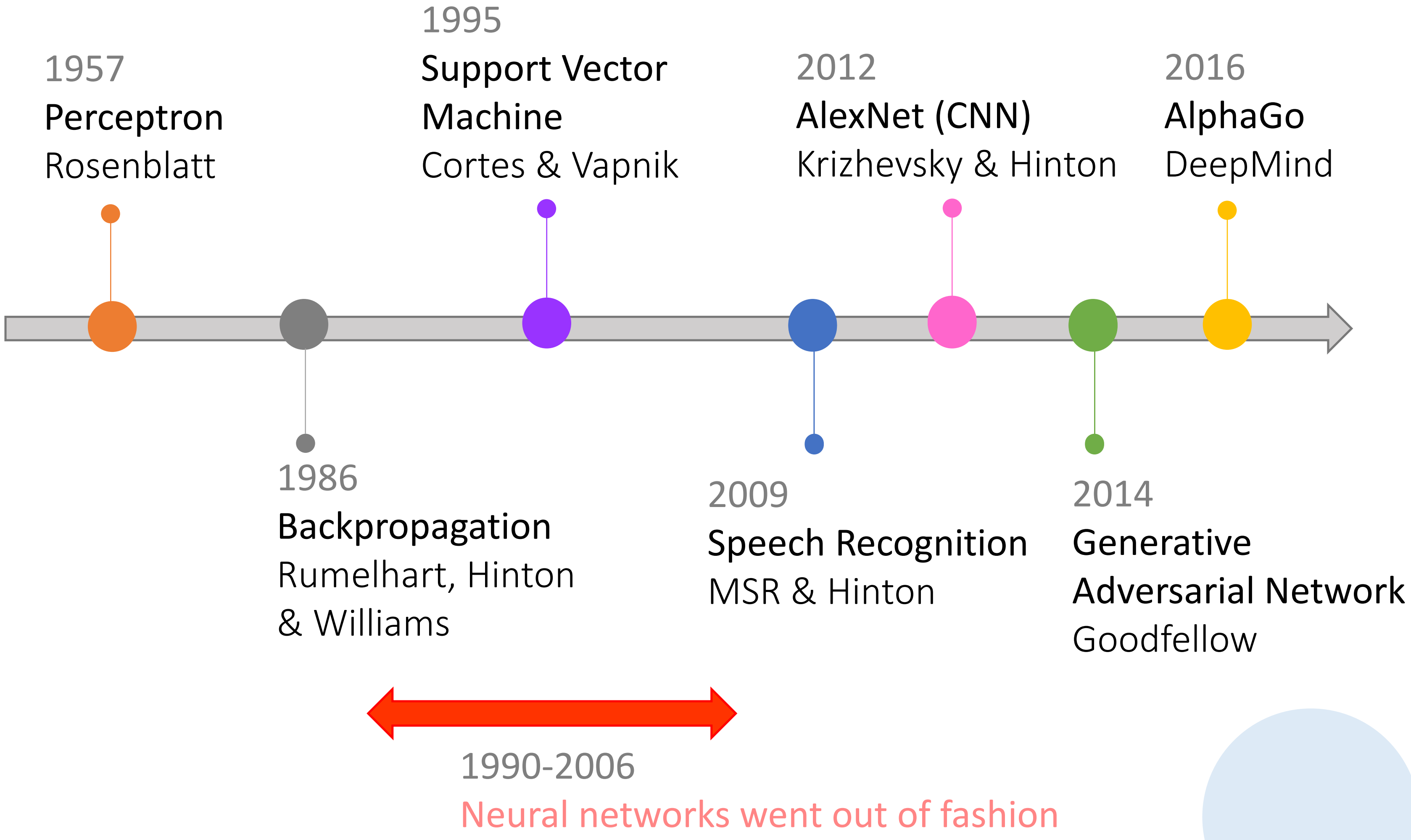
Ability to learn without explicitly being programmed

### Deep Learning

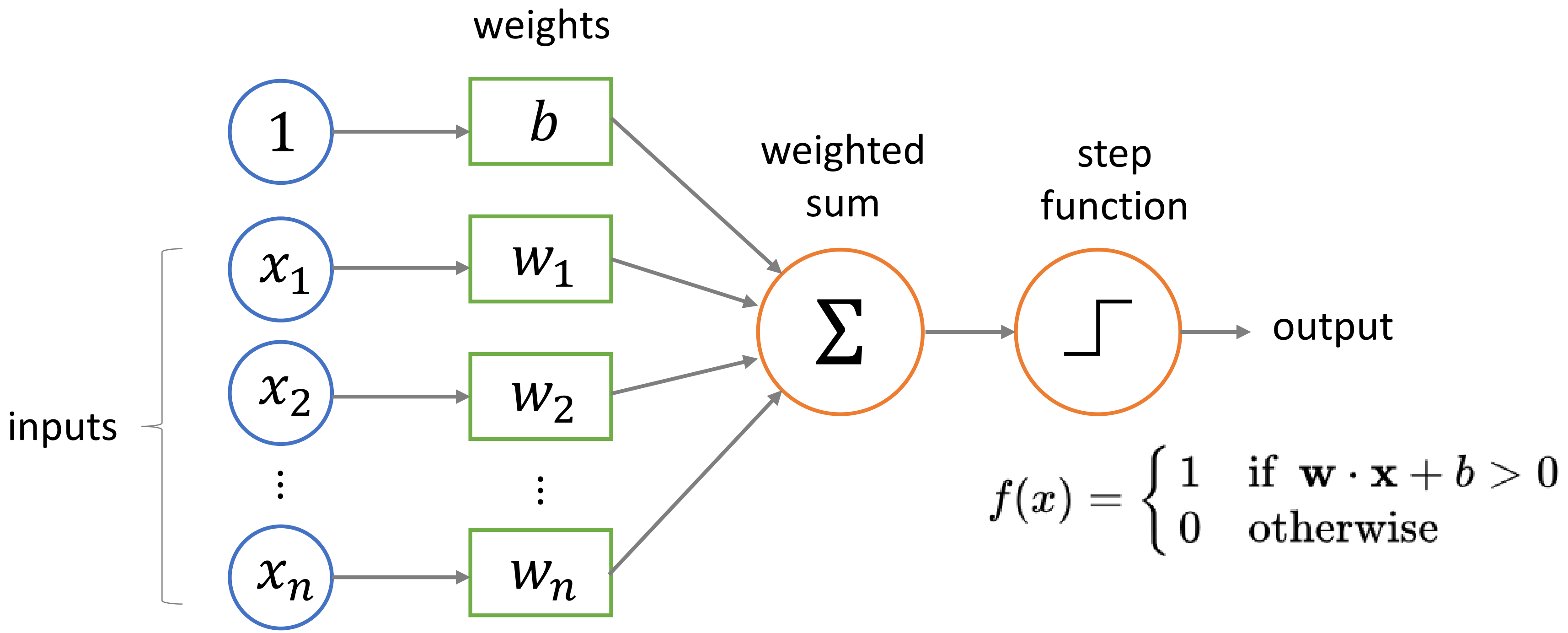
Learn underlying features in data using neural networks

[slide credit: Alexander Amini]

# Timeline

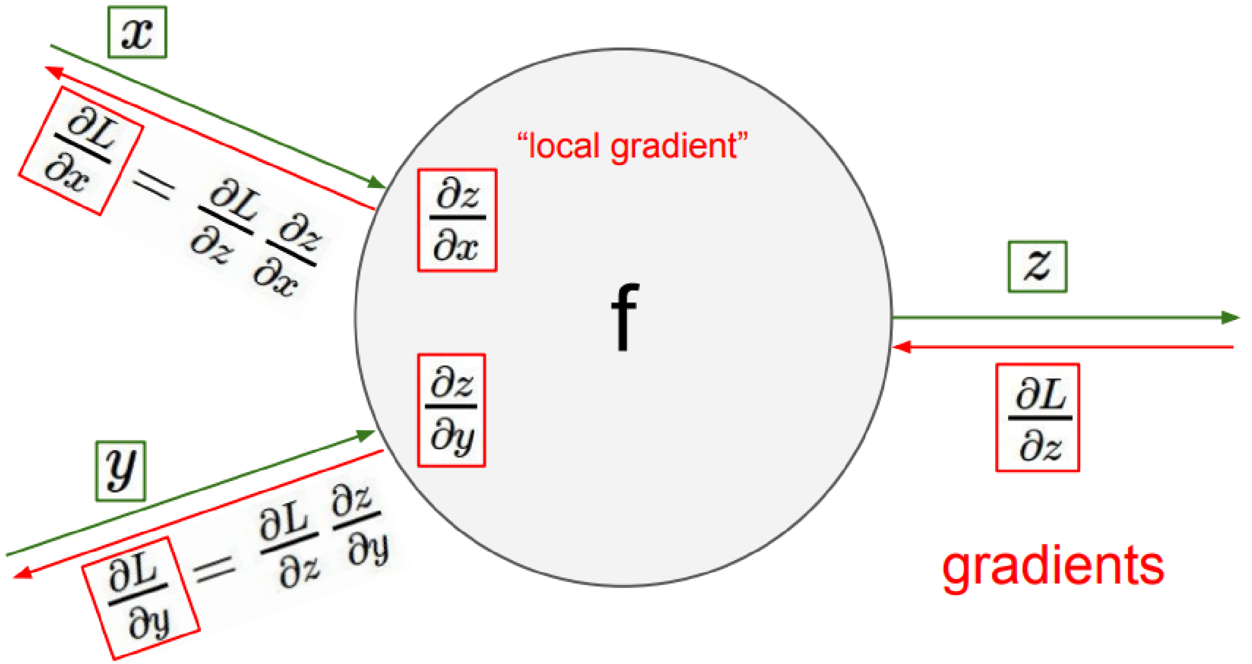

1957
Perceptron
Rosenblatt
1986
Backpropagation
Rumelhart, Hinton & Williams
1995
Support Vector Machine
Cortes & Vapnik
2009
Speech Recognition
MSR & Hinton
2012
AlexNet (CNN)
Krizhevsky & Hinton
2014
Generative Adversarial Network
Goodfellow
2016
AlphaGo
DeepMind
1990-2006
Neural networks went out of fashion

# Perceptron (Rosenblatt 1957)




- The perceptron algorithm is guaranteed to converge only if the data are **linearly separable**.

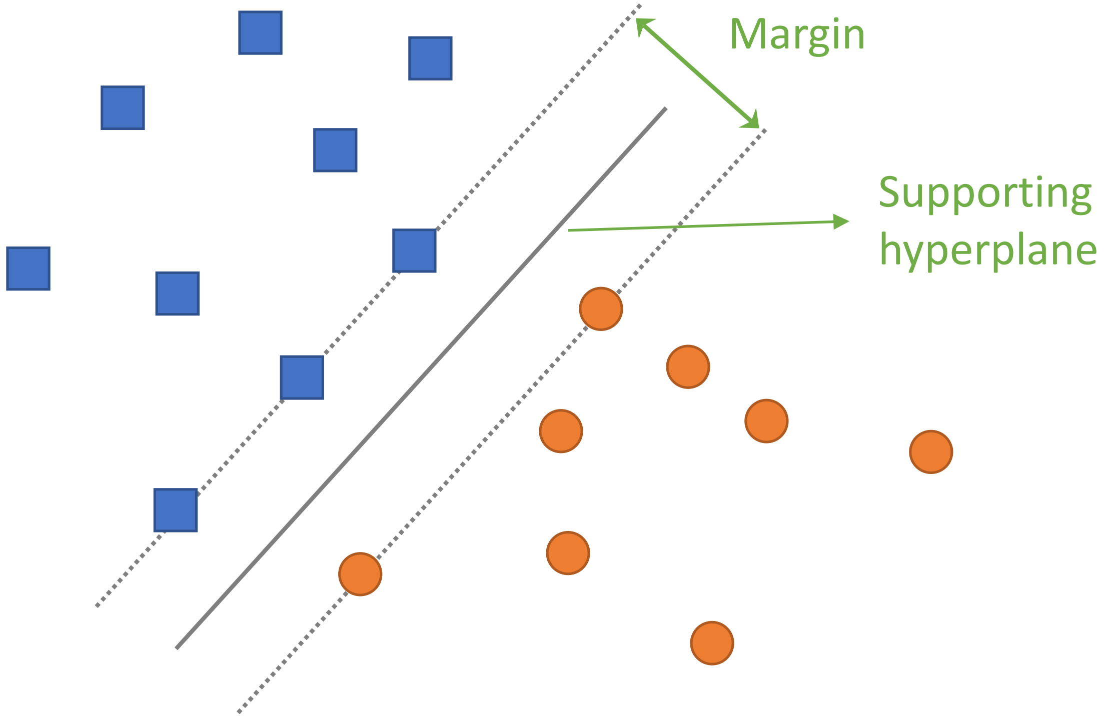
# Backpropagation (Rumelhart et al. 1986)




- Backpropagation calculates a gradient that is needed for updating the weights of the neural network.

[slide credit: Stanford CS231n]

# Support Vector Machine (Cortes & Vapnik 1995)




- The support vector machine (SVM) is a maximal margin classifier.
- Both SVMs and perceptrons are **shallow** (only one hidden layer).

# Resurgence of Neural Networks

- Neural networks went out of fashion between 1990-2006.
  1. Neural networks were difficult to train.
  2. Neural networks did not outperform other approaches.

- Why the resurgence?
  1. Big data: Large open datasets
  2. Hardware: GPU & parallelization
  3. Software: Open source framework

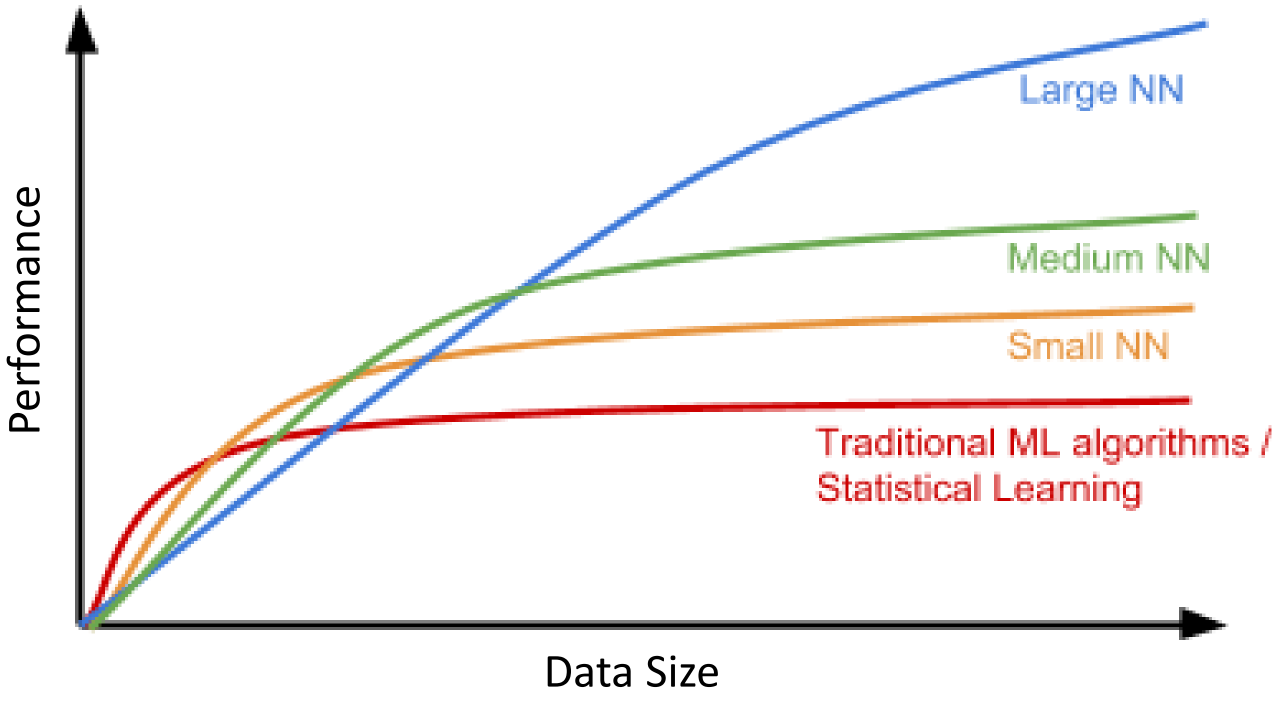

TensorFlow

PyTorch

IMAGENET

Caffe2

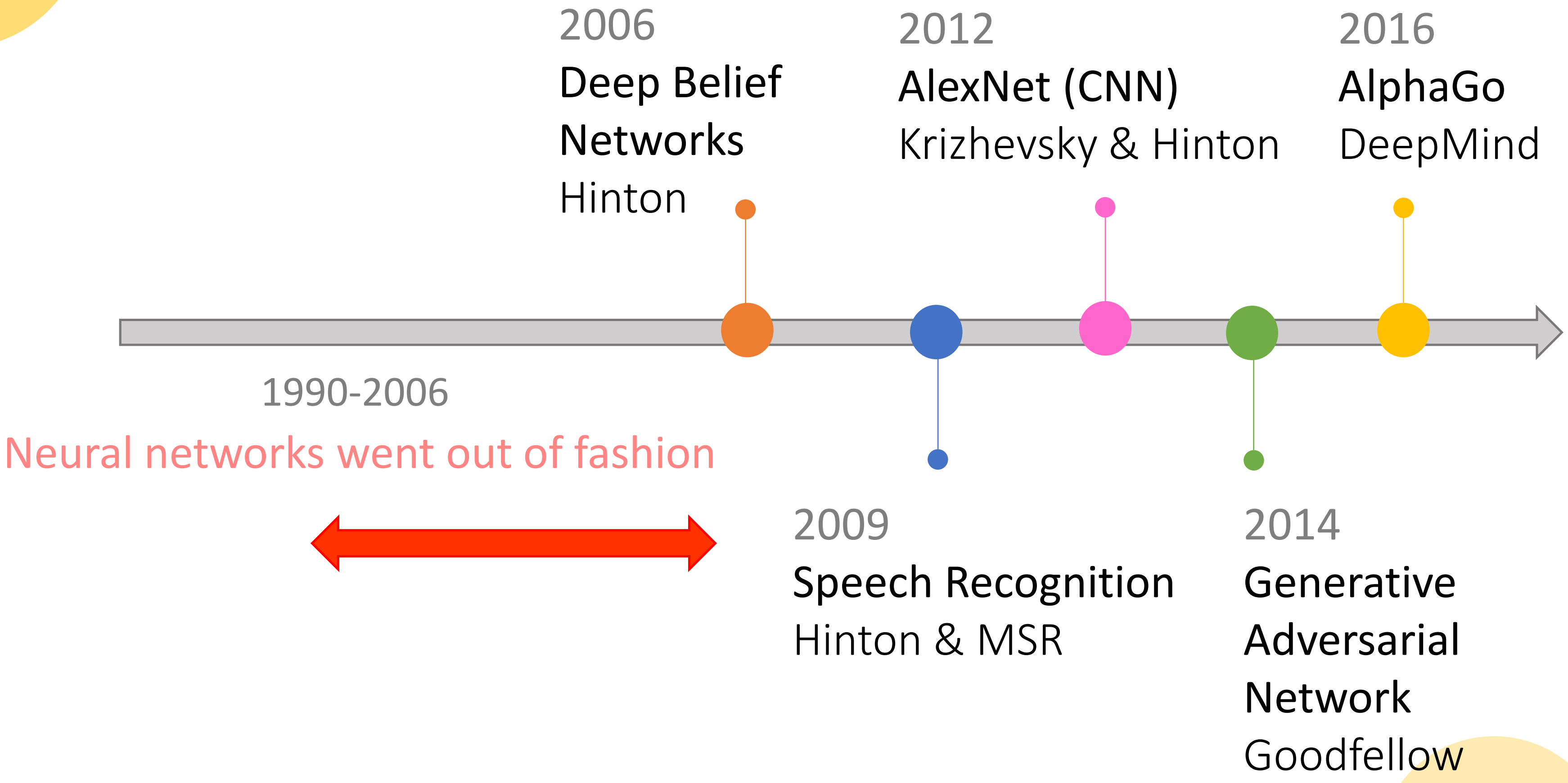
# Data Size vs. Performance



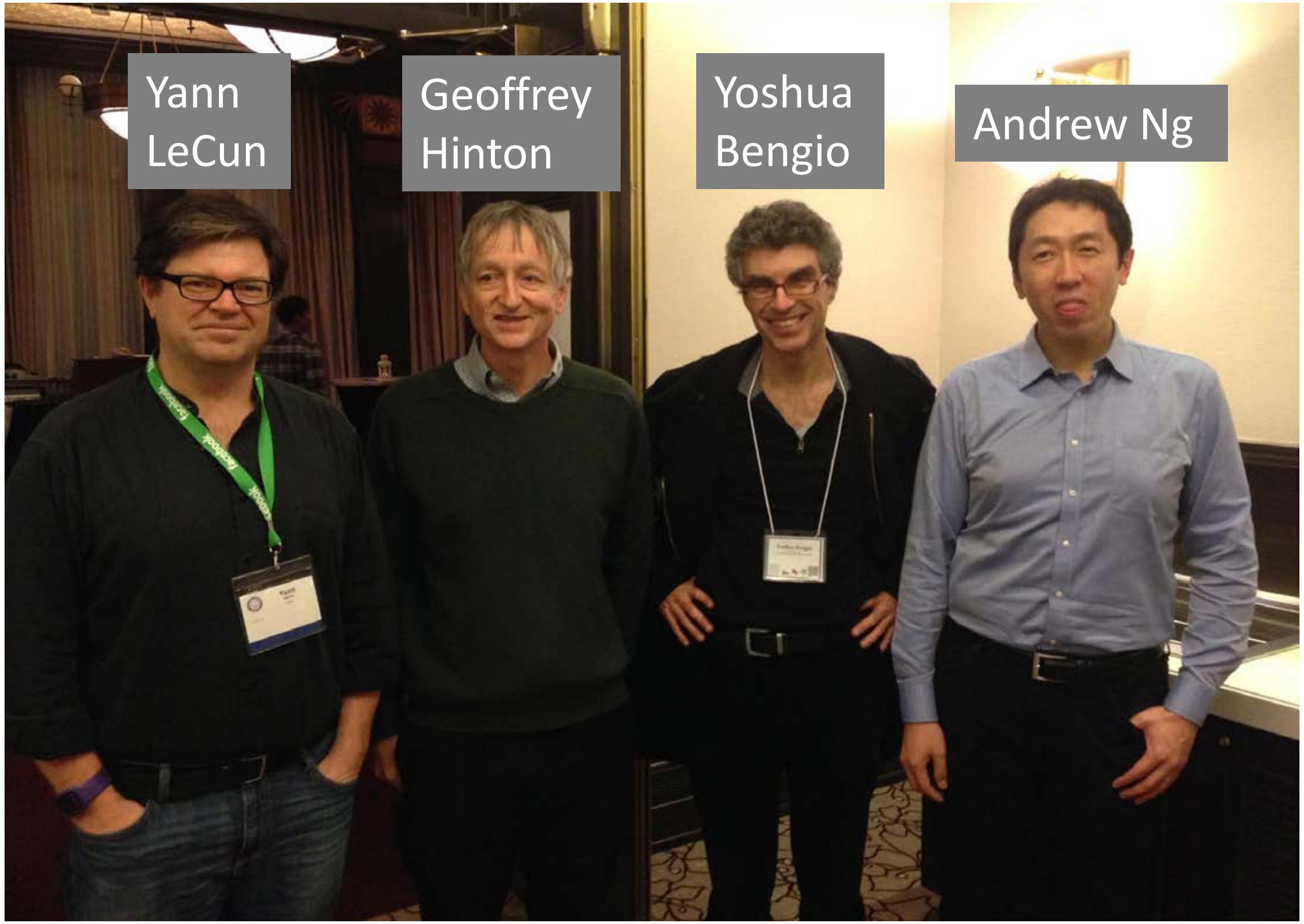

- The performance of traditional ML methods saturates as data sizes increase, while that of Neural Networks keeps growing.

[image credit: Andrew Ng]

# Timeline

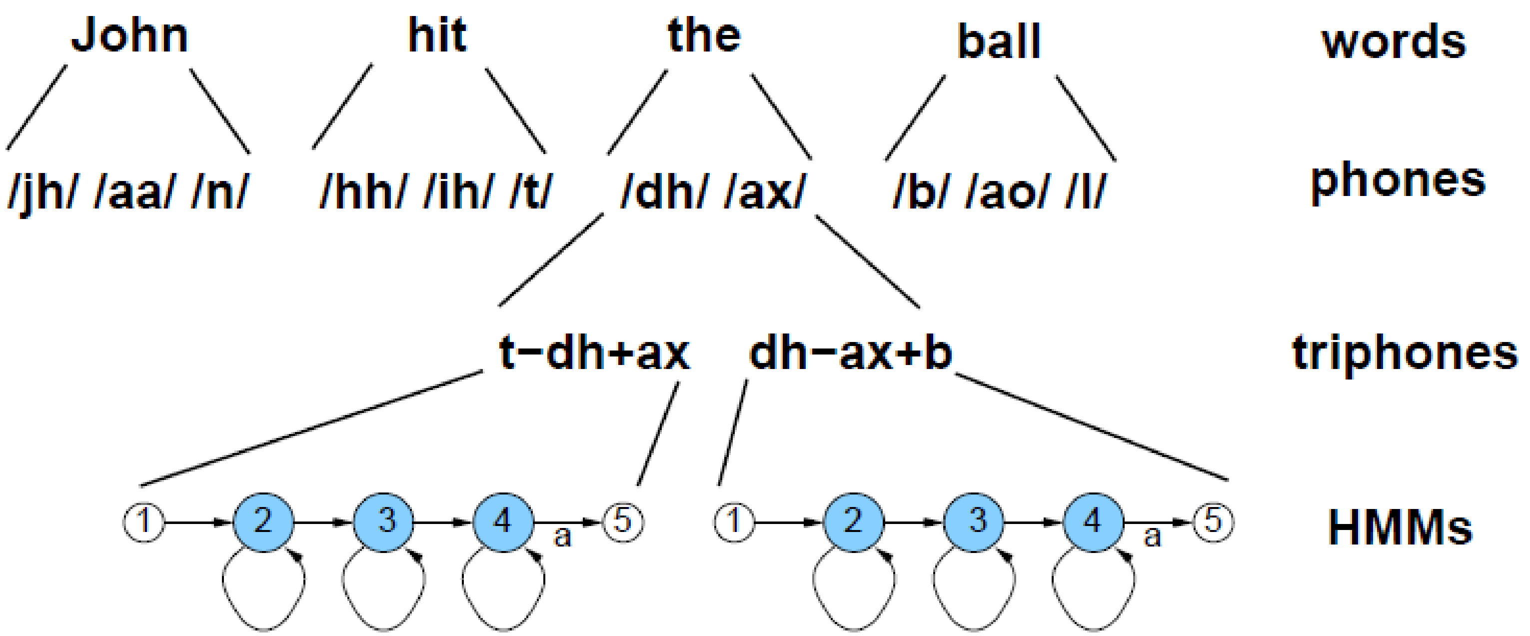

2006
Deep Belief Networks
Hinton
2012
AlexNet (CNN)
Krizhevsky & Hinton
2016
AlphaGo
DeepMind
1990-2006
Neural networks went out of fashion
2009
Speech Recognition
Hinton & MSR
2014
Generative Adversarial Network
Goodfellow

# Heroes of Deep Learning




- LeCun, Hinton, and Bengio are **ACM Turing Award** Winners in 2018.

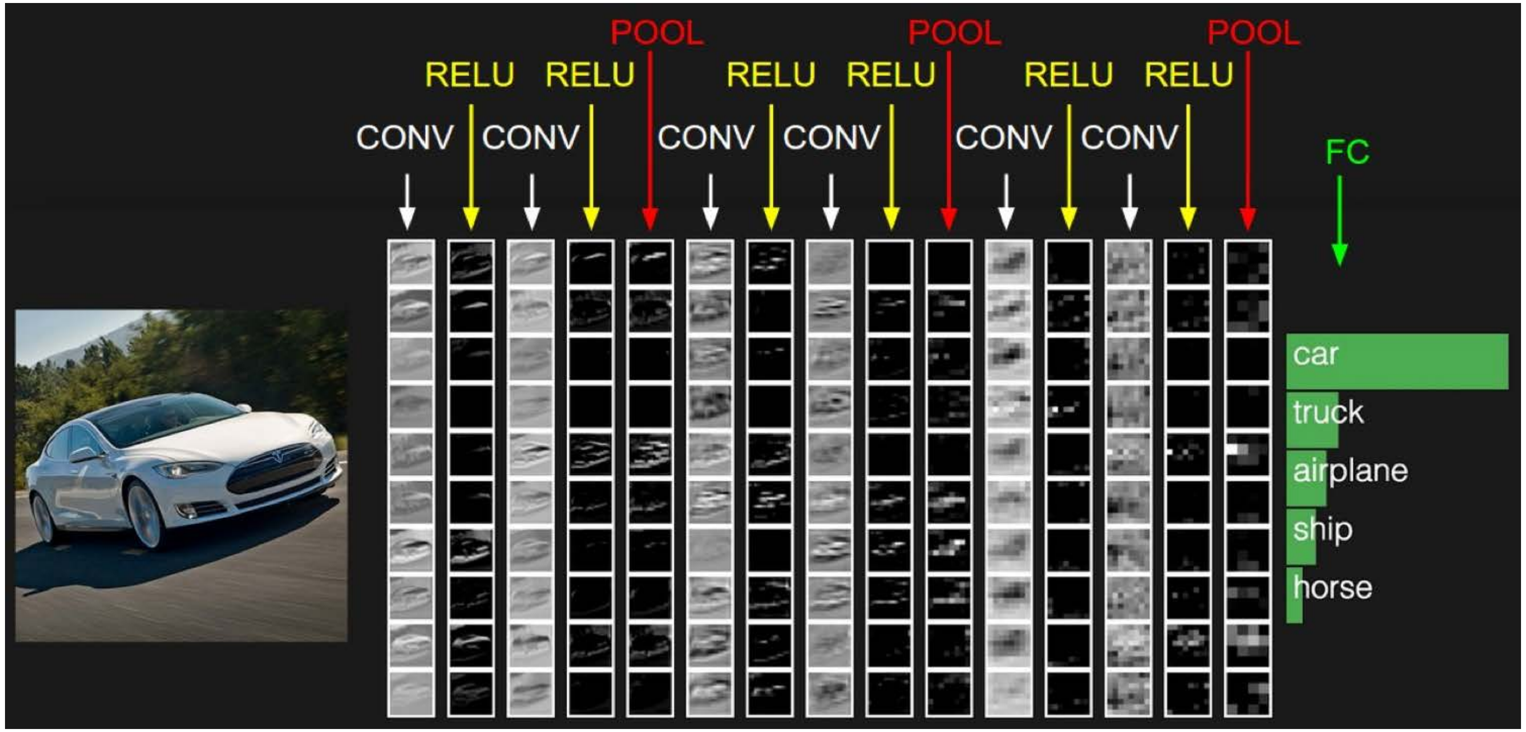
# Speech Recognition (Hinton & MSR 2009)




- Deep learning helped speech recognition take a huge leap forward at Microsoft in 2009, and then Google as well in 2010.

[image credit: Kate Knill]

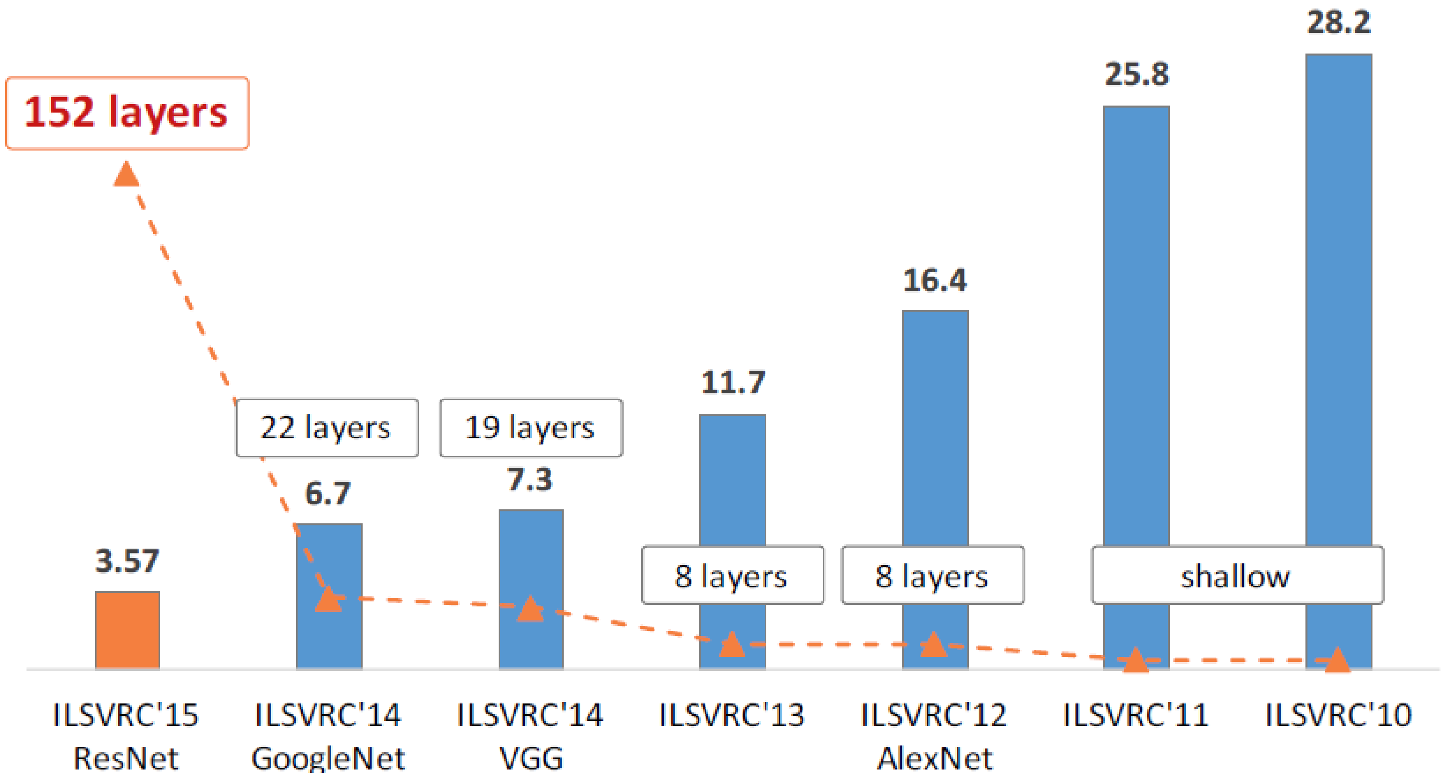
# AlexNet (Krizhevsky 2012)




- The AlexNet achieved a top-5 error of 15.3%, more than 10.8 percentage points ahead of the runner up for the ImageNet competition.

[image credit: Stanford CS231n]

# Revolution of Network Depth



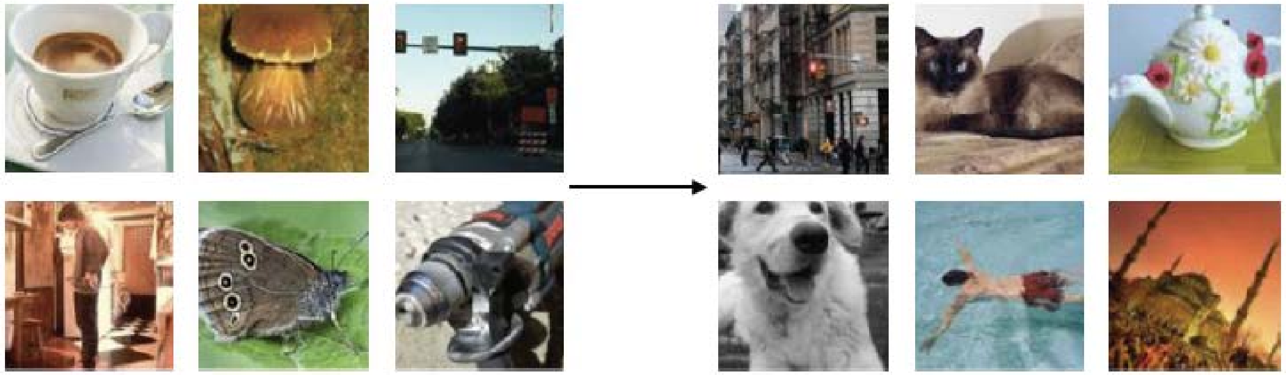

ImageNet Classification top-5 error (%)

[image credit: Kaiming He]

# Generative Adversarial Networks

- Generative models take training samples from some data distribution and learn a model representing that distribution.

Sample generation:



Training samples  Model samples

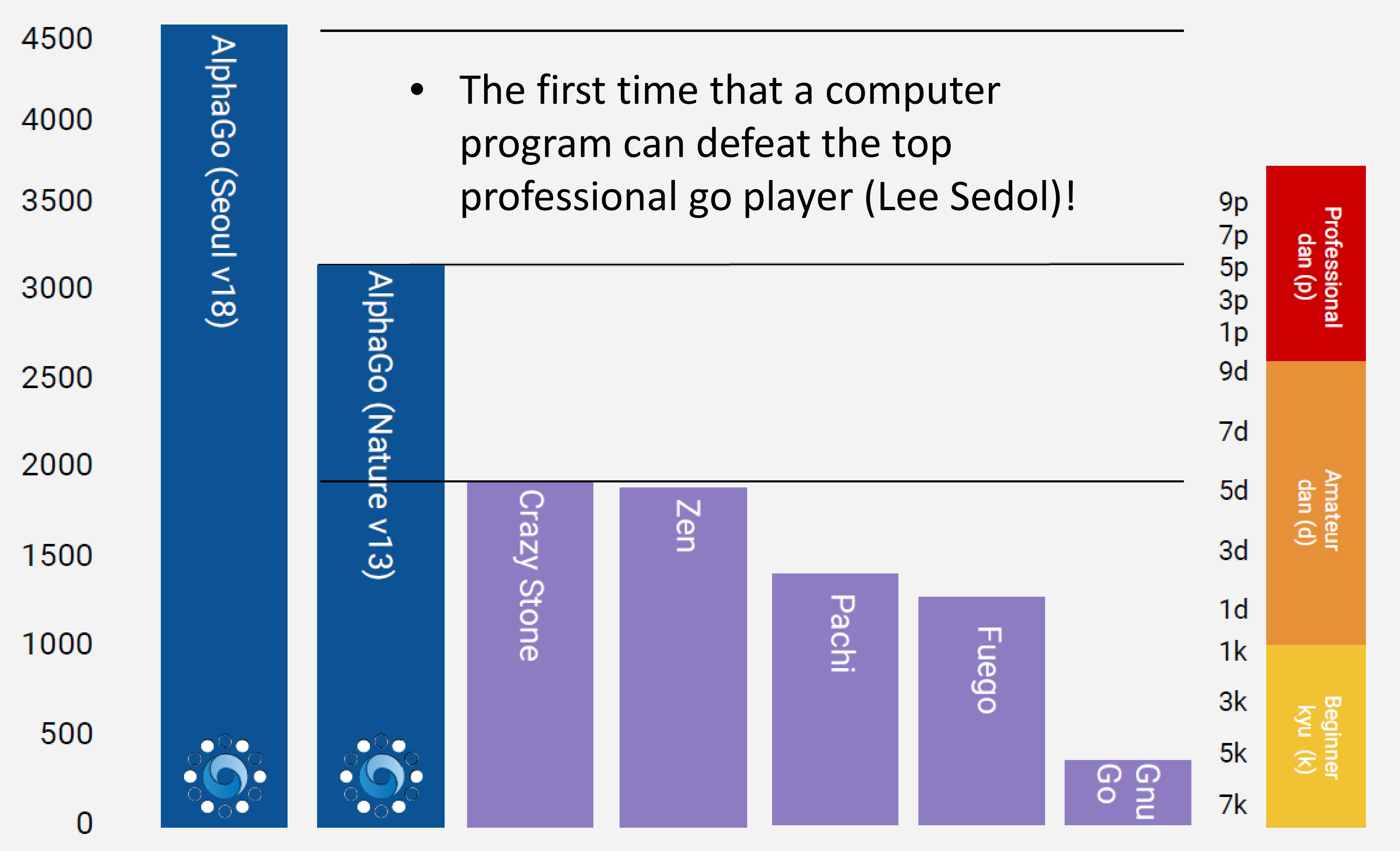
Generative adversarial nets

I Goodfellow, J Pouget-Abadie, M Mirza… - Advances in neural …, 2014 - papers.nips.cc

☆ 99 被引用 8167 次

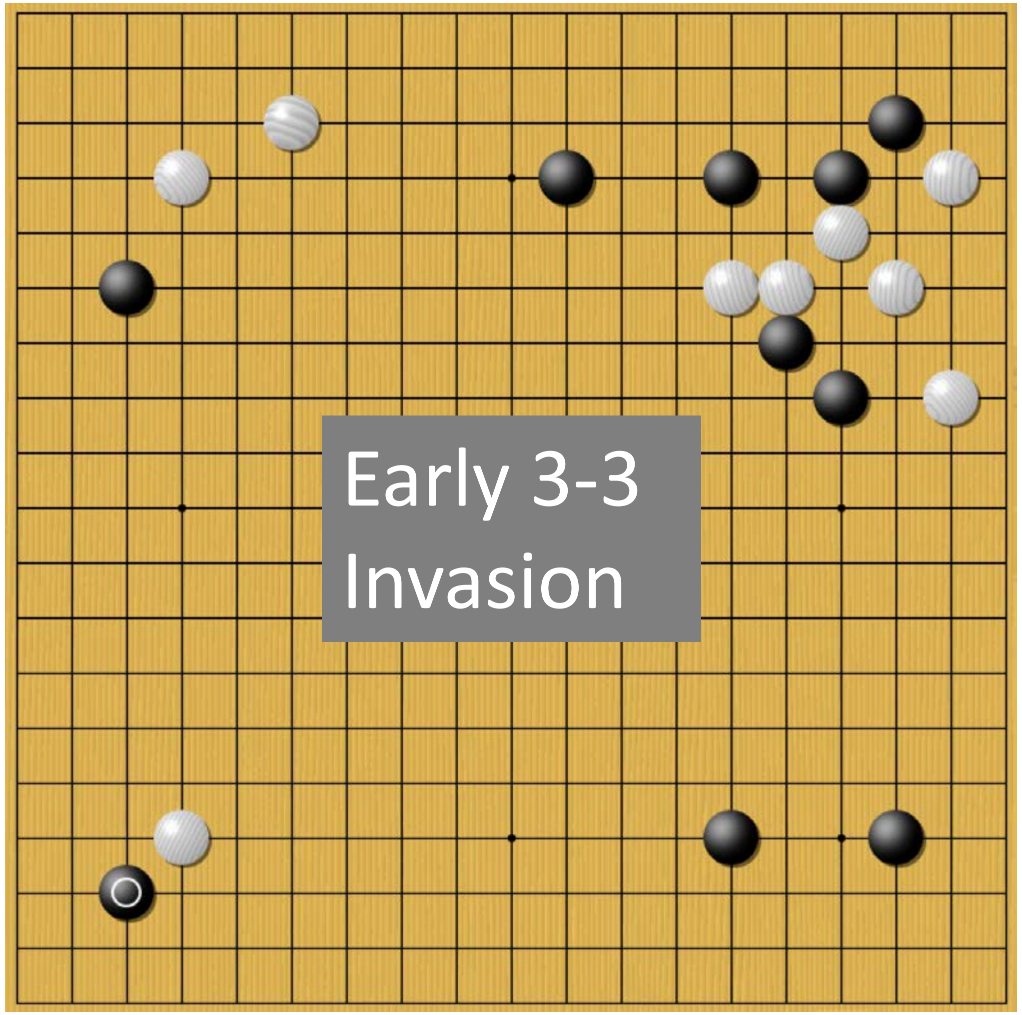
[image credit: Ian Goodfellow]

# AlphaGo (DeepMind 2016)



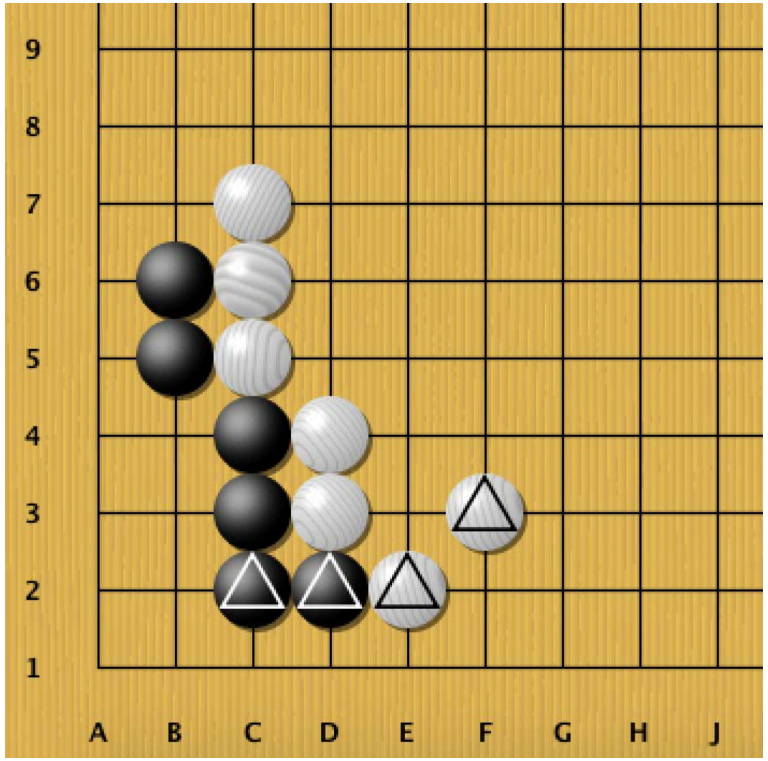

- The first time that a computer program can defeat the top professional go player (Lee Sedol)!

# AlphaGo’s Opening Novelties

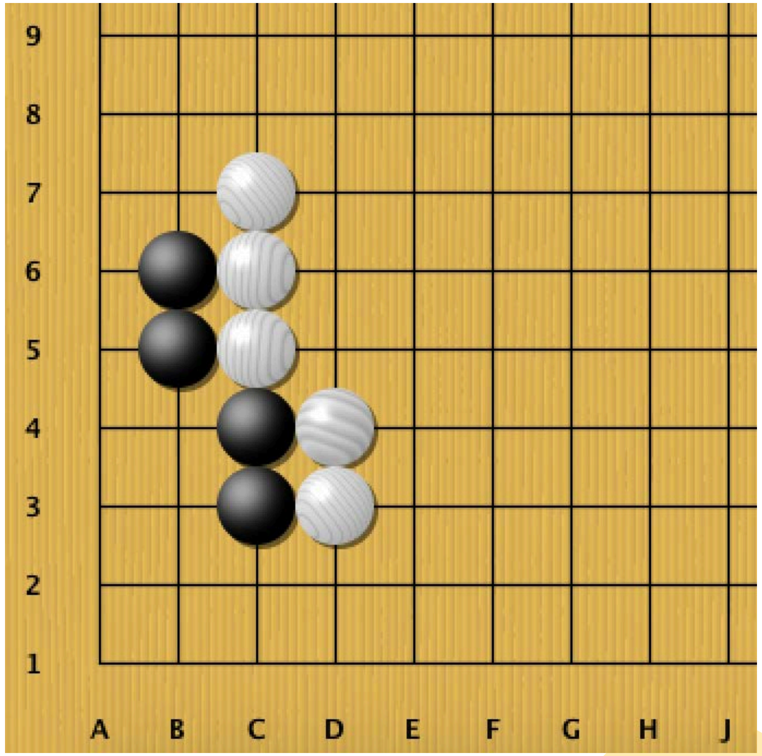


# AlphaGo's Opening Novelties (Cont.)

Textbook sequence




AlphaGo's innovation




- Early 3-3 invasion now becomes a common opening strategy adopted by professional go players.

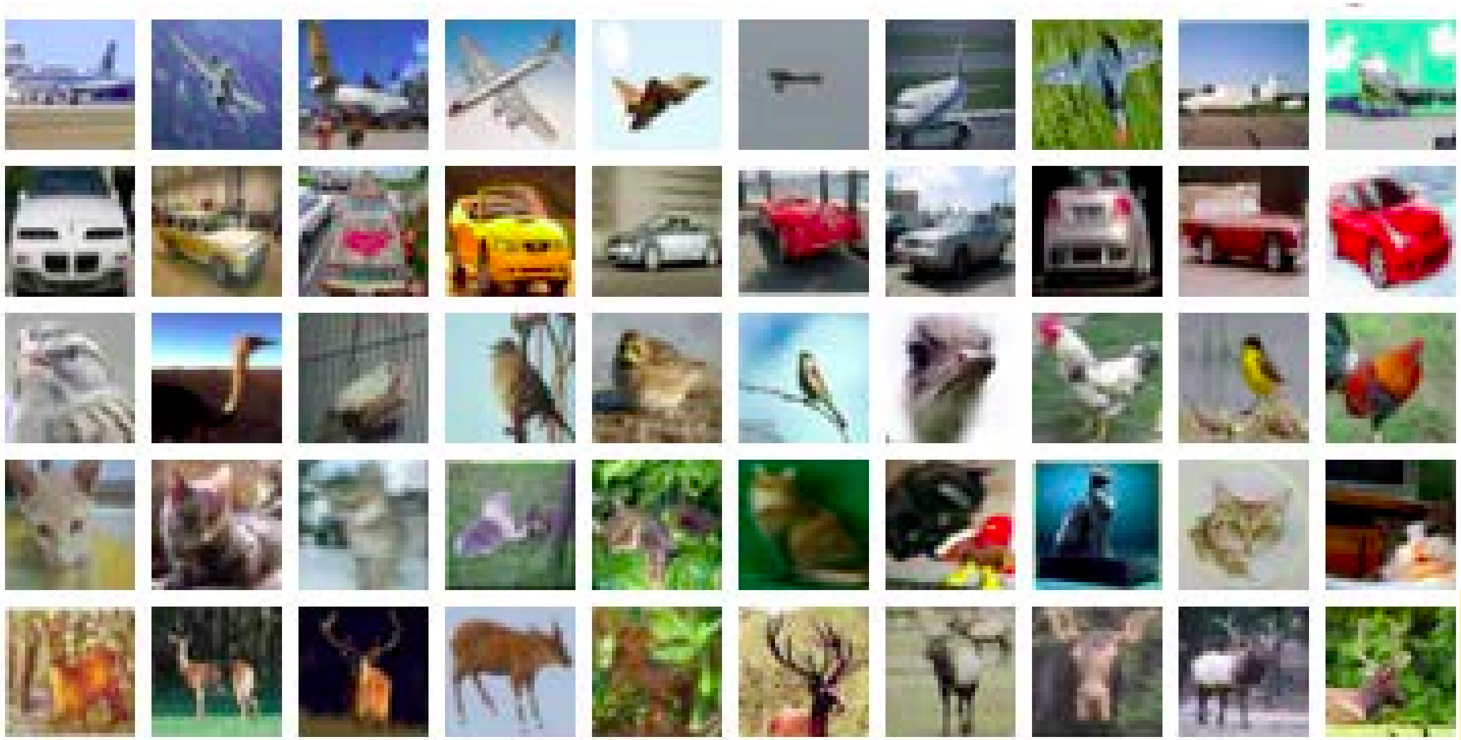
# Three Pillars of Machine Learning

- Supervised Learning:
  - **Data**: (x, y), where x is data, y is label
  - **Goal**: Learn a function that maps x to y
- Unsupervised Learning:
  - **Data**: Just data x and no labels
  - **Goal**: Learn underlying hidden structure of the data
- Reinforcement Learning:
  - Problems involving an agent interacting with an environment, which provides numeric reward signals
  - **Goal**: Learn how to take actions to maximize reward

# Supervised Learning

- Given a training set of input-output pairs, supervised learning learns a function $f$ that can predict the response to the input: $f(x) = y$

airplane
car
bird
cat
deer



# Supervised Learning

# Deep Learning = Learning Representations

- The traditional model of pattern recognition (since the late 50's)

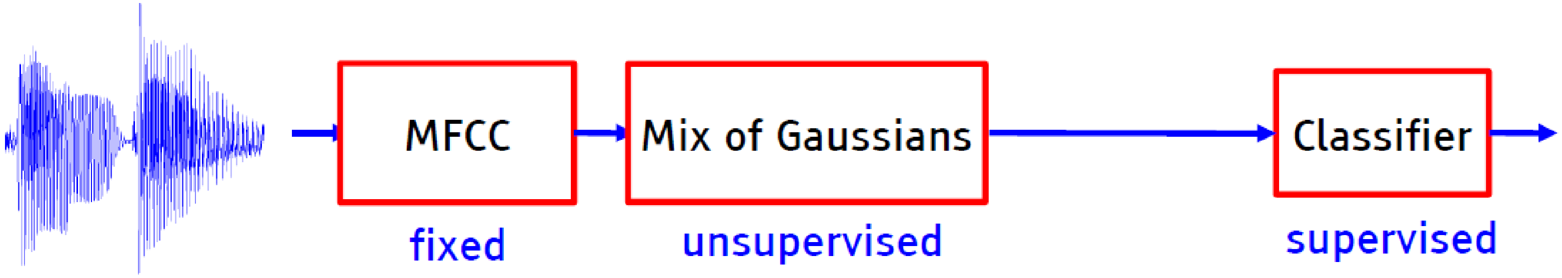
Hand-crafted Feature Extractor → Simple Trainable Classifier →

- End-to-end learning / Feature learning / Deep learning

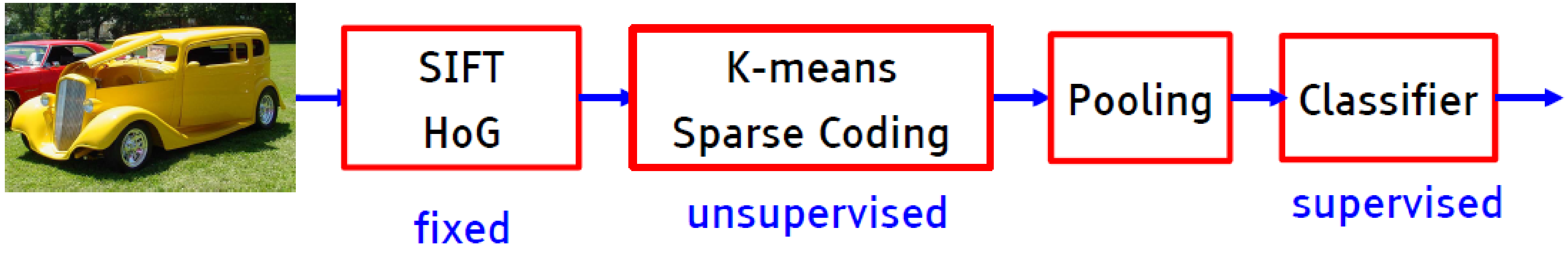
Trainable Feature Extractor → Trainable Classifier →

# Pipelines of Pattern Recognition

- Speech recognition: early 90's – 2011




- Object Recognition: 2006 - 2012

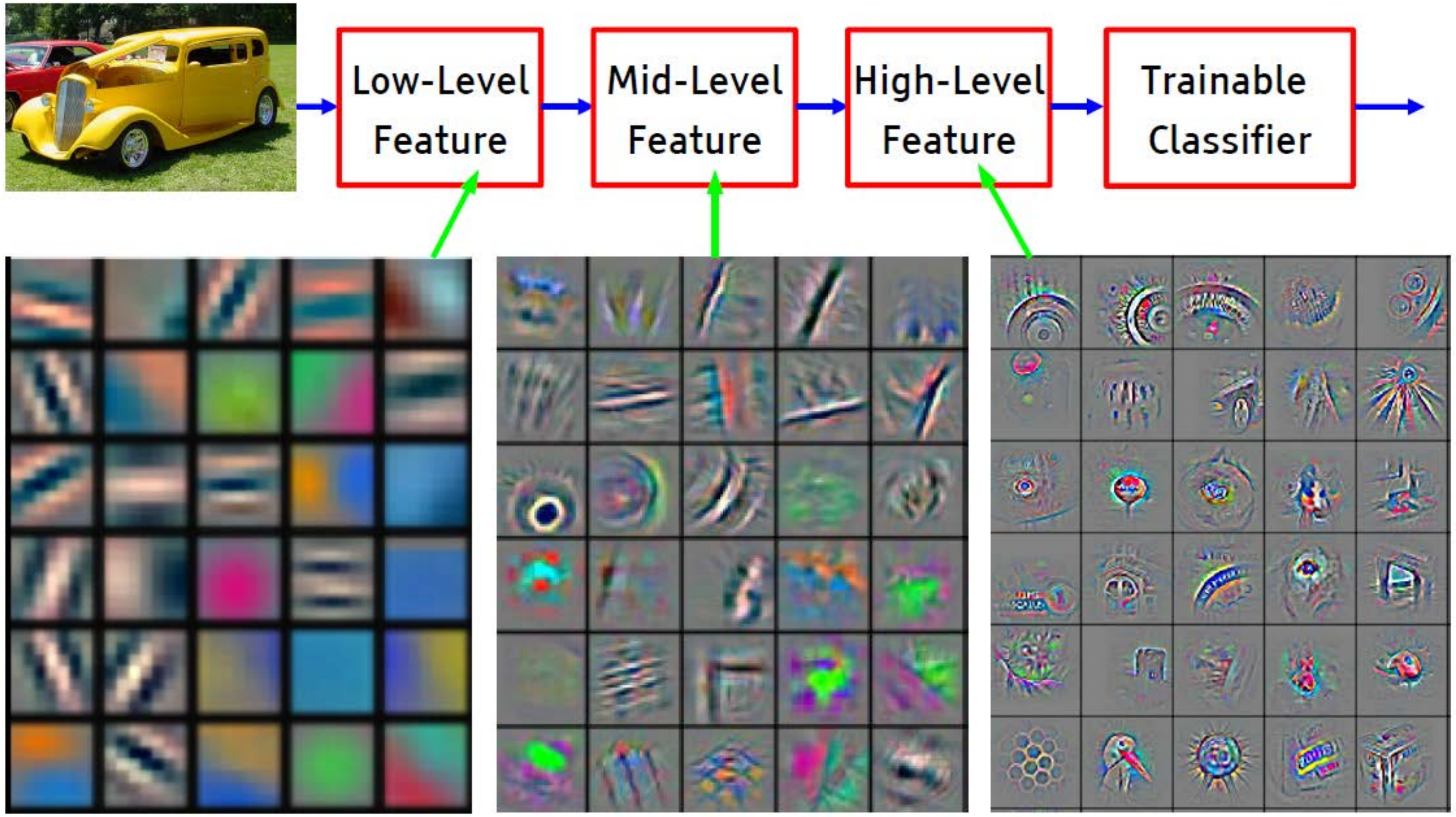



Low-level Features

High-level Features

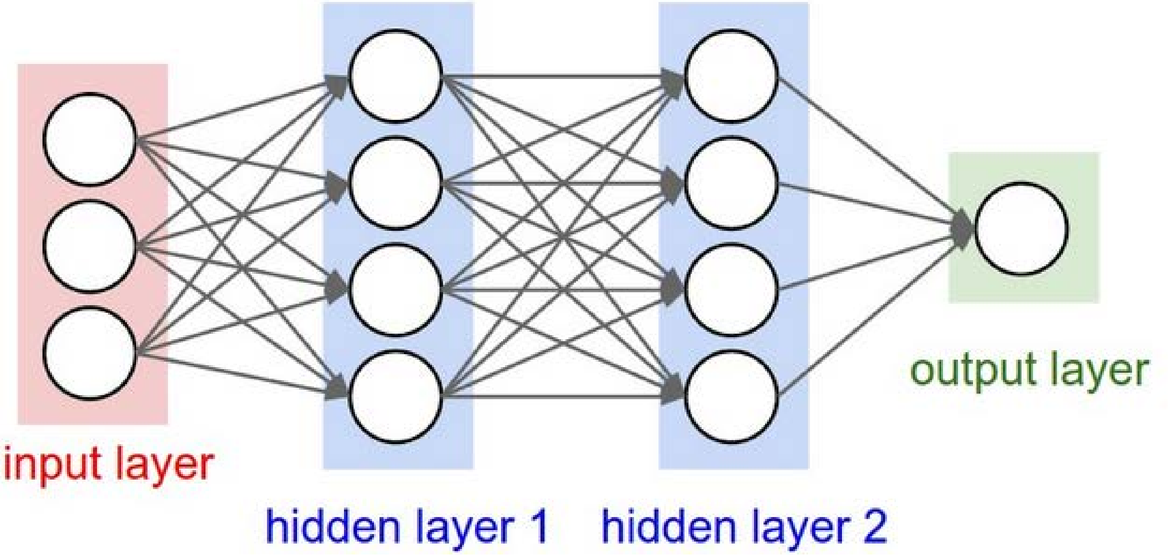
[image credit: Yann LeCun]

# Deep Learning = Learning Hierarchical Representations



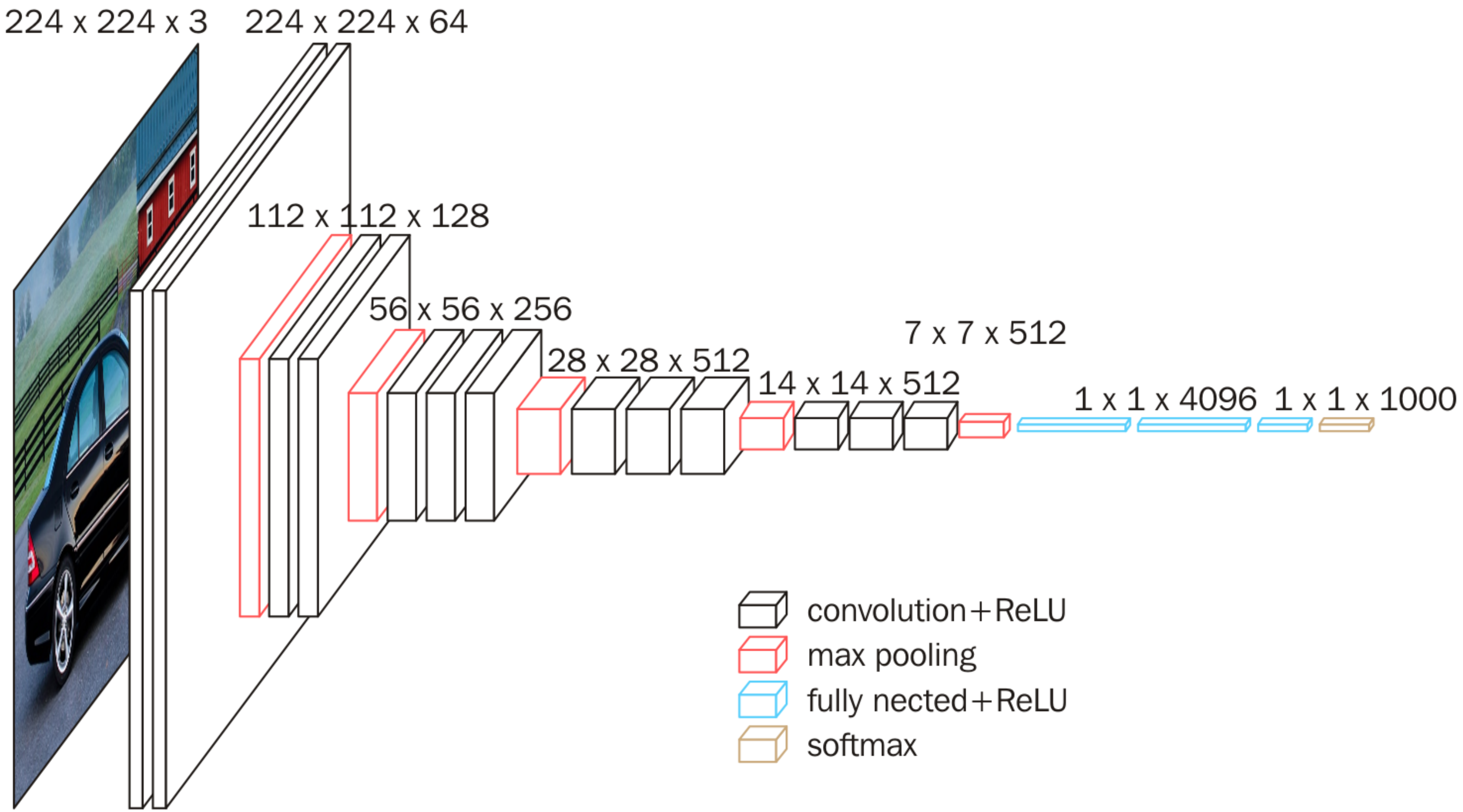

[image credit: Yann LeCun]

# Fully Connected Neural Network




- Each neuron is connected to **all** neurons in the previous layer.
- No spatial information!
- And many, many parameters!

# Convolutional Neural Network (VGG16)

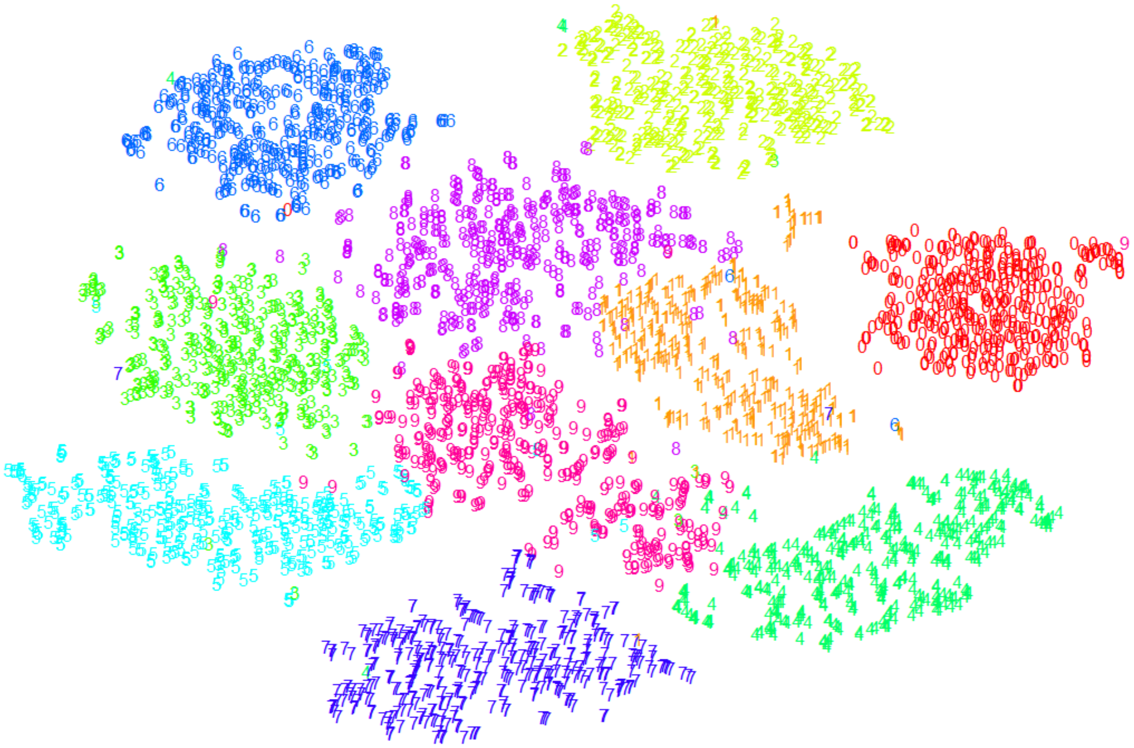


# Three Pillars of Machine Learning

- Supervised Learning:
  - **Data**: (x, y), where x is data, y is label
  - **Goal**: Learn a function that maps x to y
- Unsupervised Learning:
  - **Data**: Just data x and no labels
  - **Goal**: Learn underlying hidden structure of the data
- Reinforcement Learning:
  - Problems involving an agent interacting with an environment, which provides numeric reward signals
  - **Goal**: Learn how to take actions to maximize reward

# Unsupervised Learning: Clustering

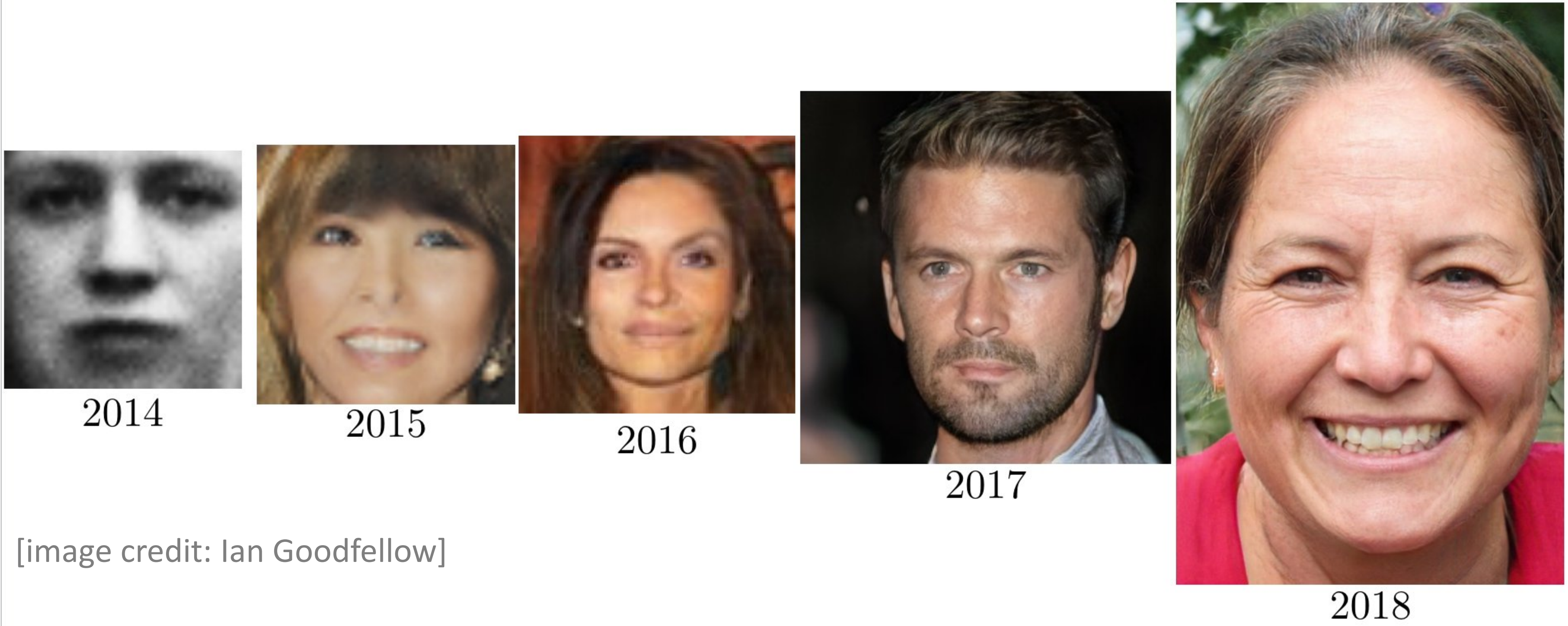
- **Clustering** groups a set of objects such that objects in the same group are more similar to each other than to those in other groups.



# Outline

- Timeline
- Supervised Learning
  - Convolutional Neural Networks (CNN)
- Unsupervised Learning
  - Generative Adversarial Networks (GAN)
- Reinforcement Learning

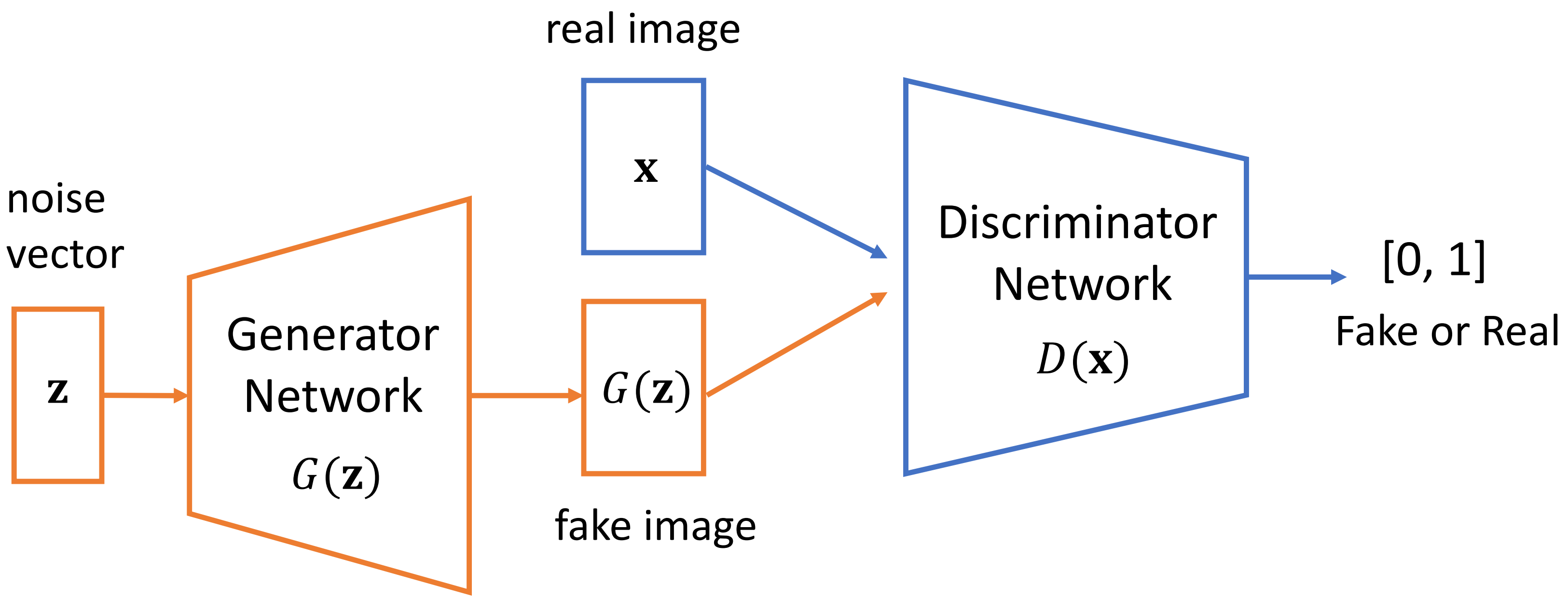
# 4.5 Years of GAN Progress




[image credit: Ian Goodfellow]

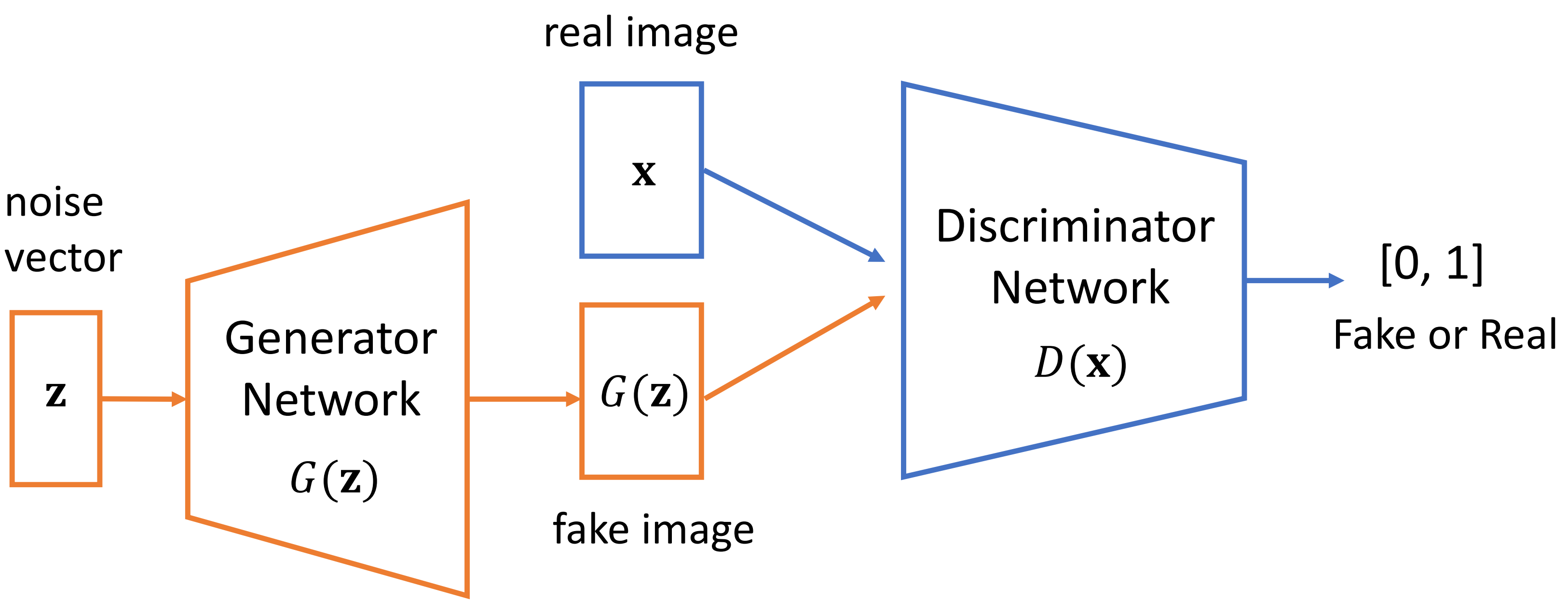
# Which Person is Real?

# Generative Adversarial Network (GAN)




- The image $\mathbf{x}$ is sampled from input data distribution.
- The vector $\mathbf{z}$ is drawn from a random variable of noise.
- $D$ tries to make $D(G(\mathbf{z}))$ near 0
- $G$ tries to make $D\big(G(\mathbf{z})\big)$ near 1

# Two Player Minmax Game[1]



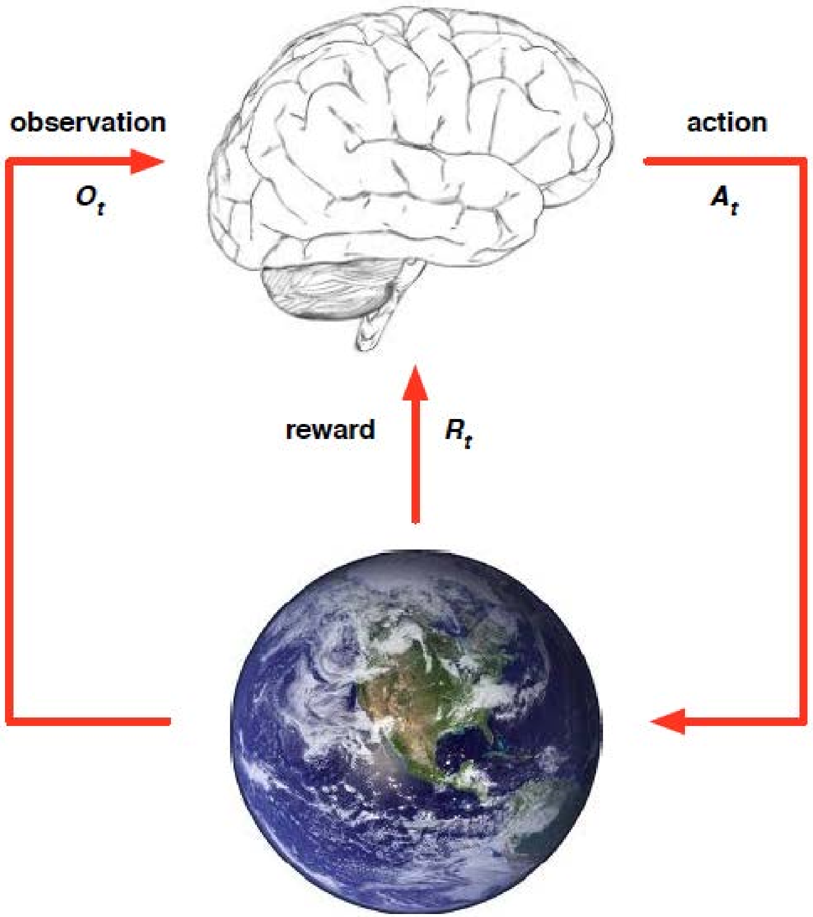

$$\min_{G}\max_{D} V(D,G) = \mathbb{E}_{\mathbf{x}}[\log(D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}}[\log(1 - D(G(\mathbf{z}))]$$

[1] Goodfellow et al., Generative Adversarial Nets, NIPS 2014.

# Three Pillars of Machine Learning

- Supervised Learning:
  - **Data**: (x, y), where x is data, y is label
  - **Goal**: Learn a function that maps x to y
- Unsupervised Learning:
  - **Data**: Just data x and no labels
  - **Goal**: Learn underlying hidden structure of the data
- Reinforcement Learning:
  - Problems involving an agent interacting with an environment, which provides numeric reward signals
  - **Goal**: Learn how to take actions to maximize reward

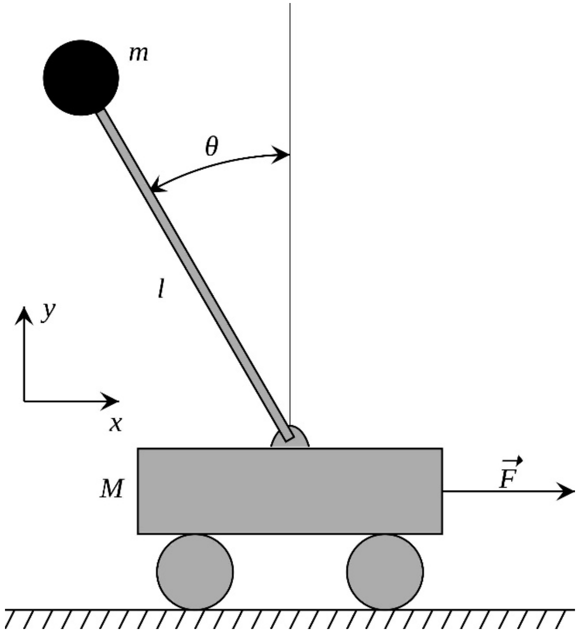
# Reinforcement Learning




- At each step $t$ the agent:
  - Executes action $A_t$
  - Receives observation $O_t$
  - Receives scalar reward $R_t$
- The environment:
  - Receives action $A_t$
  - Emits observation $O_{t+1}$
  - Emits scalar reward $R_{t+1}$
- $t$ increments at env. step

[image credit: David Silver]

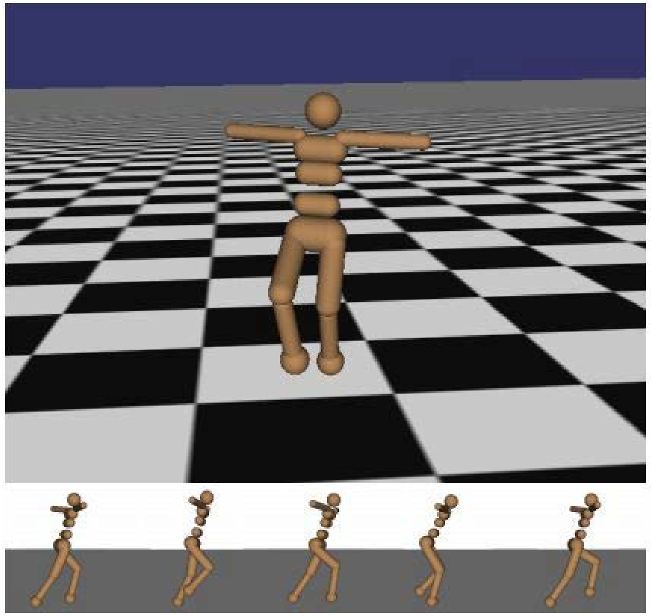
# Cart Pole Problem




[image credit: Stanford CS231n]

- **Objective**: Balance a pole on top of a movable cart
- **State**: angle, angular speed, position, horizontal velocity
- **Action:** horizontal force applied on the cart
- **Reward:** 1 at each time step if the pole is upright

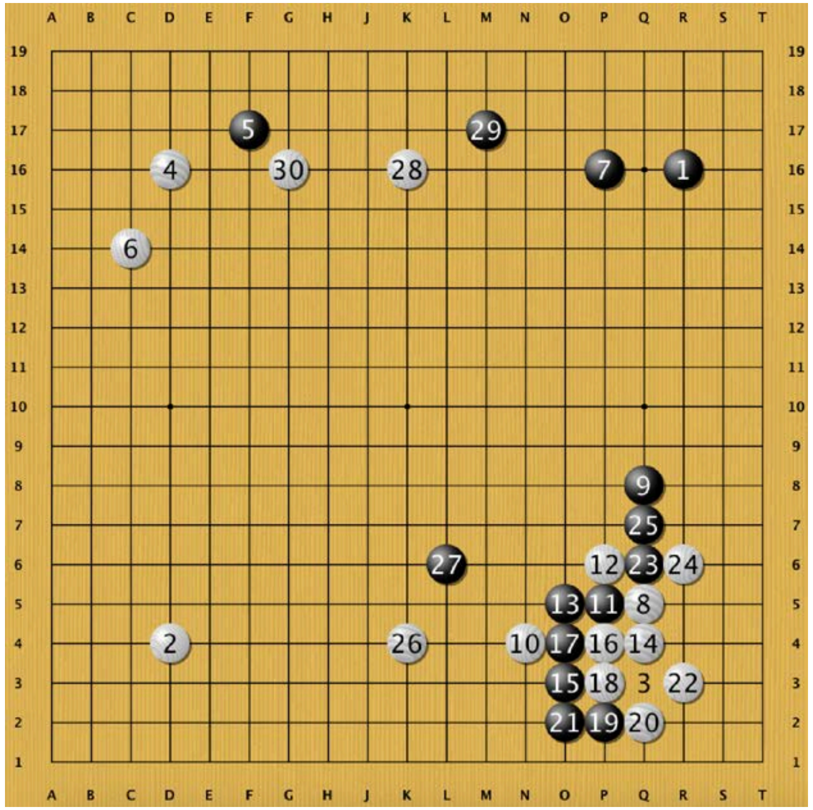
# Robot Locomotion



[image credit: Stanford CS231n]

- **Objective**: Make the robot move forward
- **State**: Angle & position of the joints
- **Action:** Torques applied on joints
- **Reward:** 1 at each time step upright + forward movement

# The Game of Go




- **Objective**: Win the game
- **State**: Position of all pieces
- **Action:** Where to put the next piece down
- **Reward:** 1 if win at the end of the game, 0 other wise

- Deep RL works great for games and virtual environments
- Pure RL is still not useful in the real word
- Because it requires too many trials to learn       - Yann LeCun

# Conclusion

1990-2006

Neural networks went out of fashion

2006
**Deep Belief Networks**
Unsupervised learning

2009
**Speech Recognition**
Supervised learning

2012
**AlexNet (CNN)**
Supervised learning

2014
**Generative Adversarial Network**
Unsupervised learning

2016
**AlphaGo**
Reinforcement Learning