딥러닝 구현을 위한 텐서플로우 개발

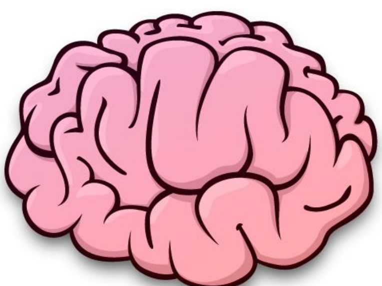
김성균

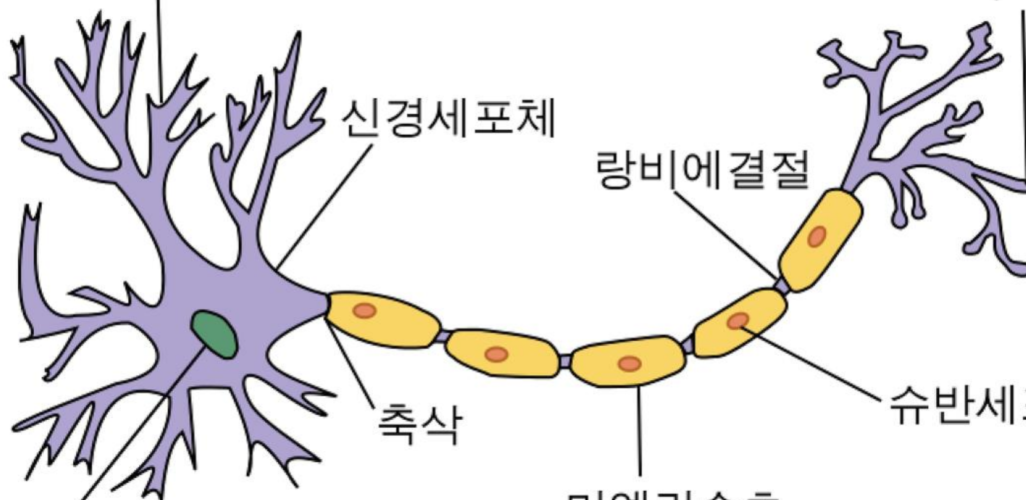
TensorFlow

6강
딥러닝의 기본개념

01.인공신경망(artificial neural network, **ANN**)

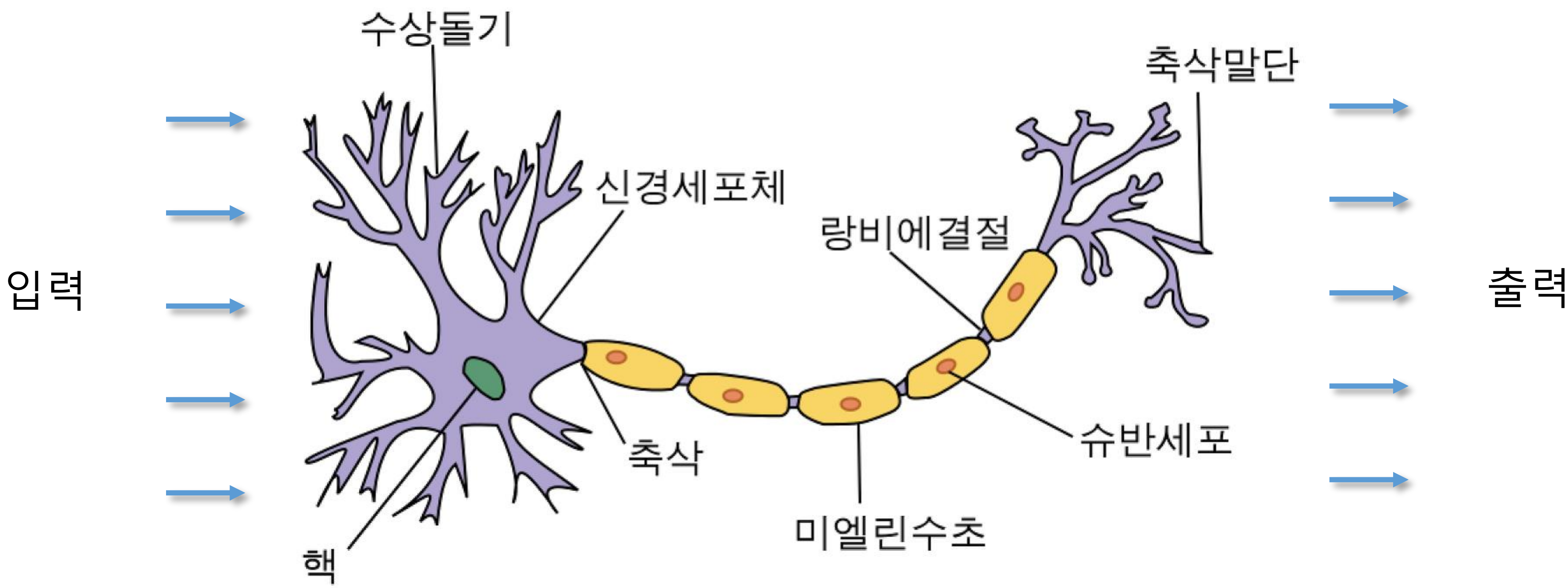
- 생물학의 신경망(동물의 중추신경계중 특히 뇌)에서 영감을 얻은 통계학적 학습 알고리즘



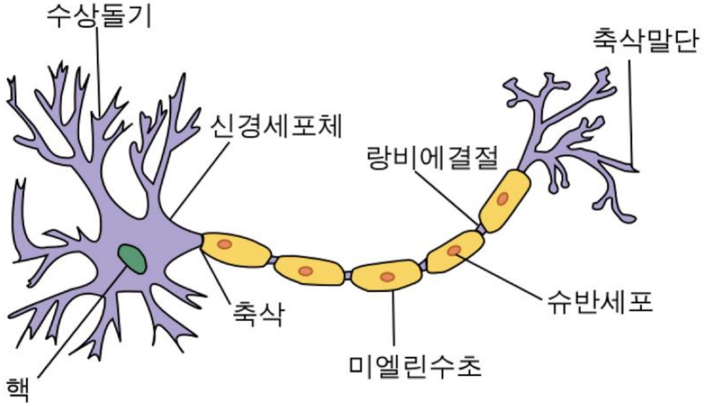


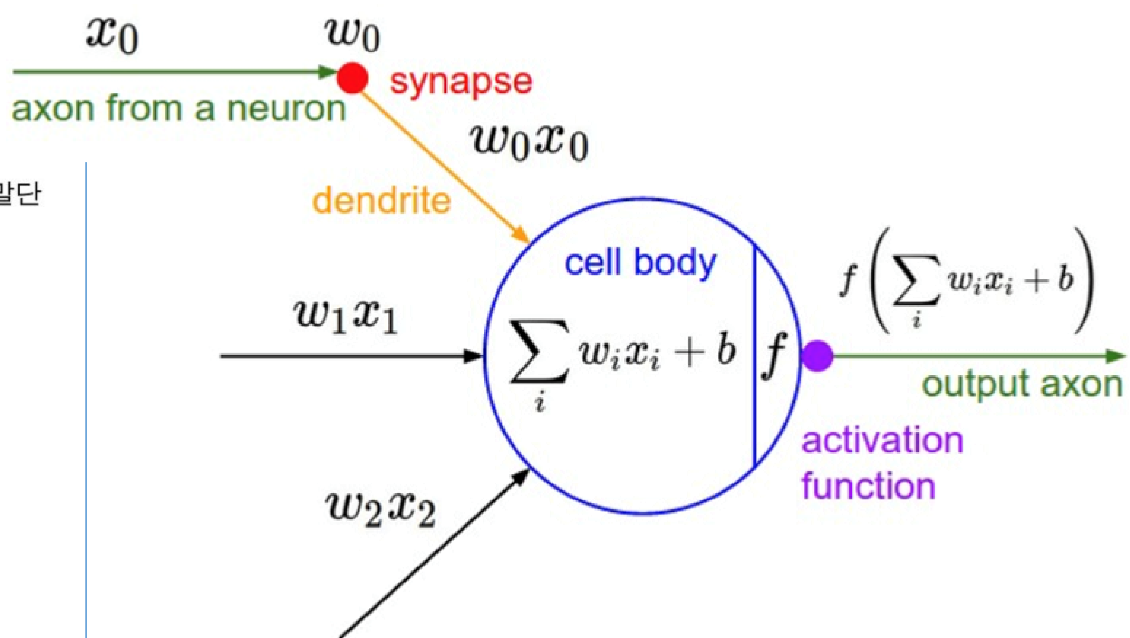
01.인공신경망(artificial neural network, **ANN)**

$xw + b$

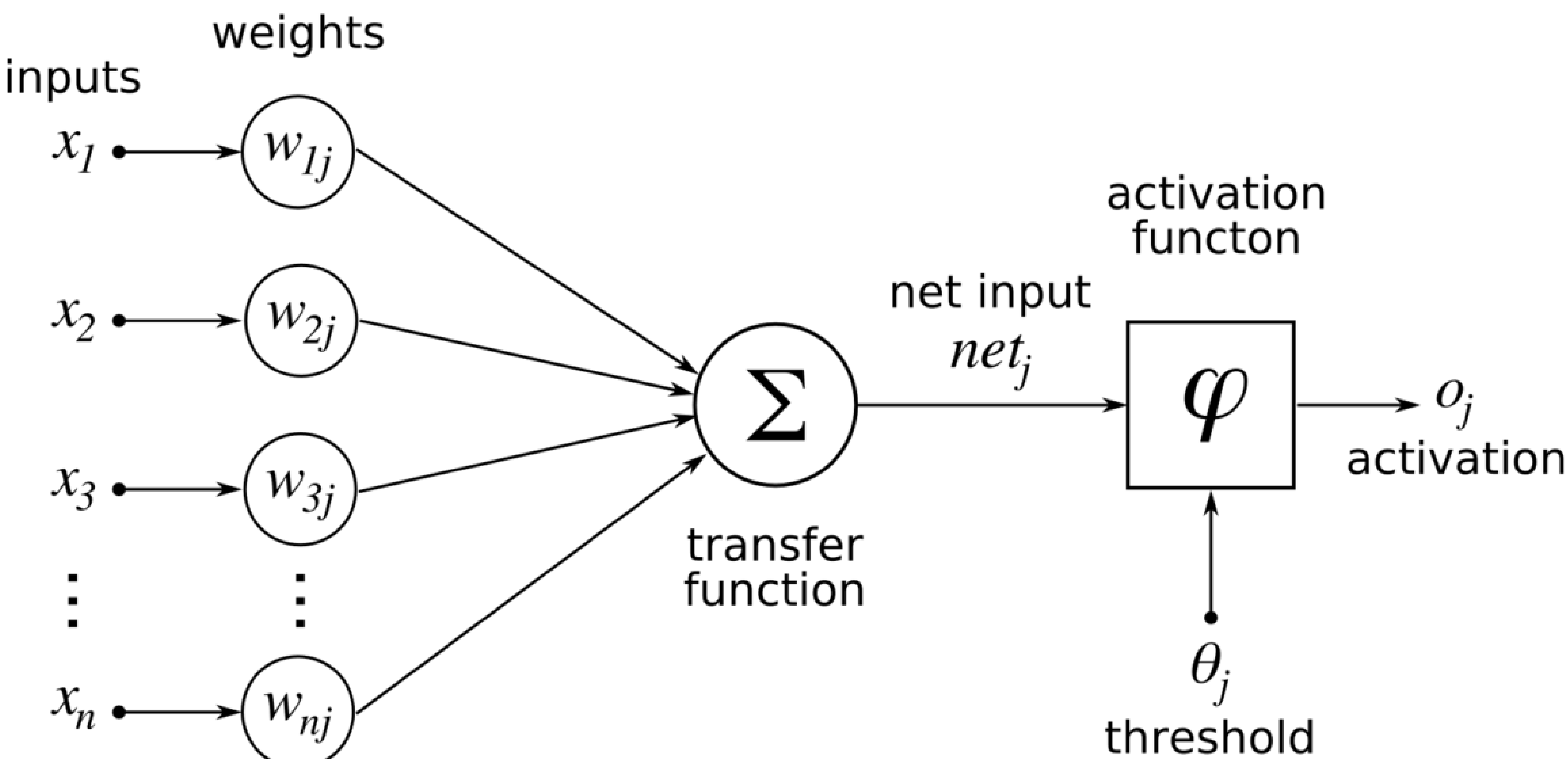


02. Activation Functions in Artificial Neural Networks

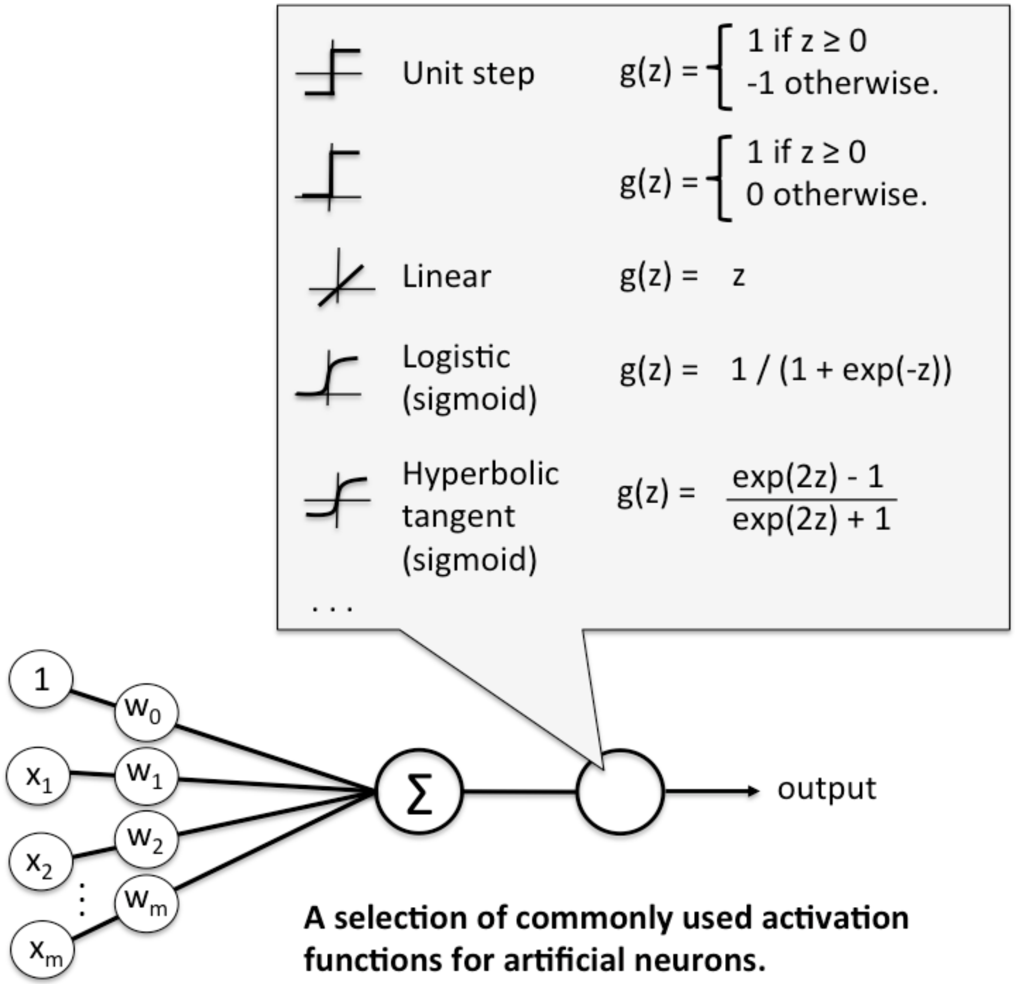




02. Activation Functions in Artificial Neural Networks

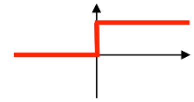
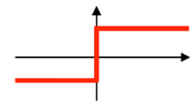
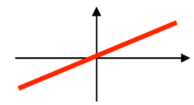
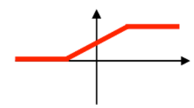
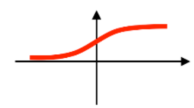
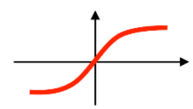
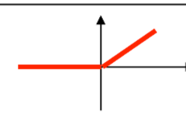
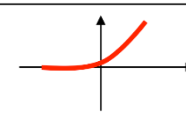


02. Activation Functions in Artificial Neural Networks



A selection of commonly used activation functions for artificial neurons.

02. Activation Functions in Artificial Neural Networks

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$	Multi-layer Neural Networks	
Rectifier, ReLU (Rectified Linear Unit)	$\phi(z) = max(0, z)$	Multi-layer Neural Networks	
Rectifier, softplus 	$\phi(z) = \ln(1 + e^{z})$	Multi-layer Neural Networks	

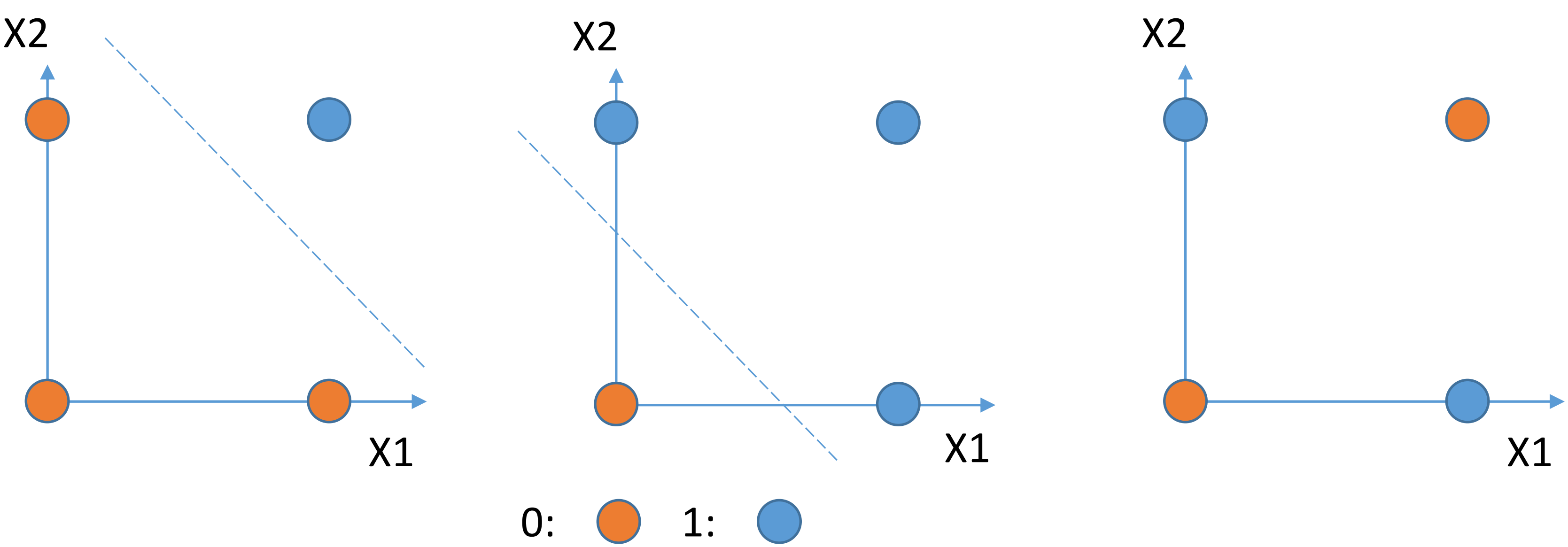
03. XOR 문제 대두

- OR과 AND에 대해서는 잘 동작하는데, XOR 문제는 linear 방식으로 풀 수가 없었음.

X1	X2	AND
0	0	0
0	1	0
1	0	0
1	1	1

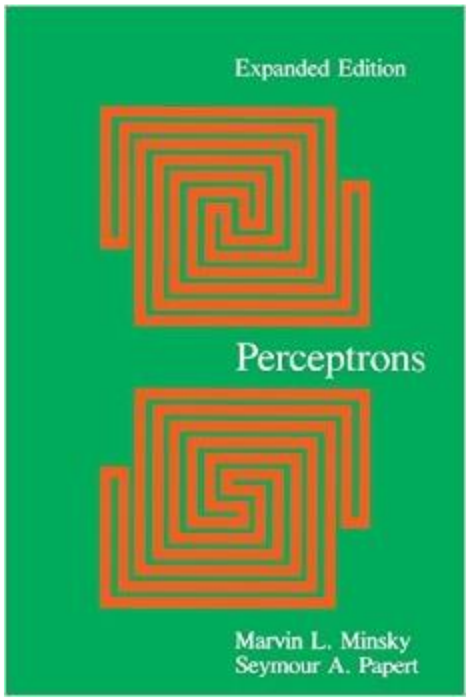
X1	X2	OR
0	0	0
0	1	1
1	0	1
1	1	1

X1	X2	XOR
0	0	0
0	1	1
1	0	1
1	1	0



04. Perceptrons(1969)

- by Marvin Minsky, founder of the MIT AI Lab



- We need to use MLP, multilayer perceptrons (multilayer neural nets)
- No one on earth had found a viable way to train MLPs good enough to learn such simple functions.

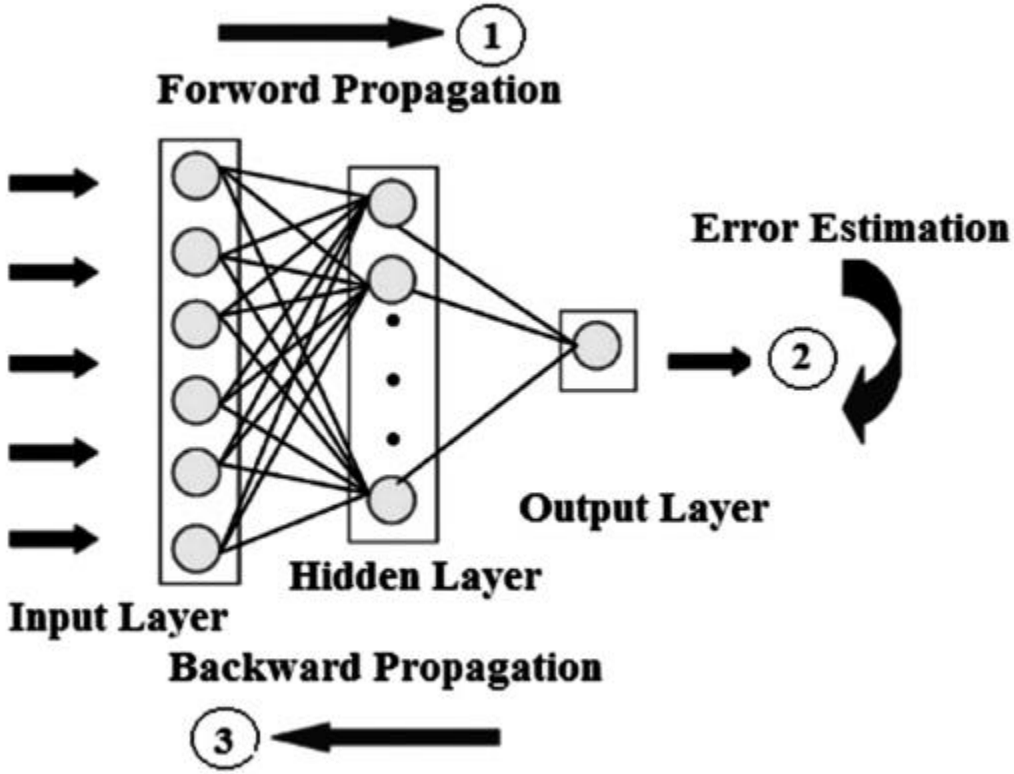
- Marvin Minsky, 1969
  - No one on earth had found a viable way to train –

- layer가 여러 개 있을 때, 각각의 layer에서 사용한 W와 b를 조절할 수 없다고 수식으로 증명.

  => 첫번째 빙하기 도래

05. Backpropagation

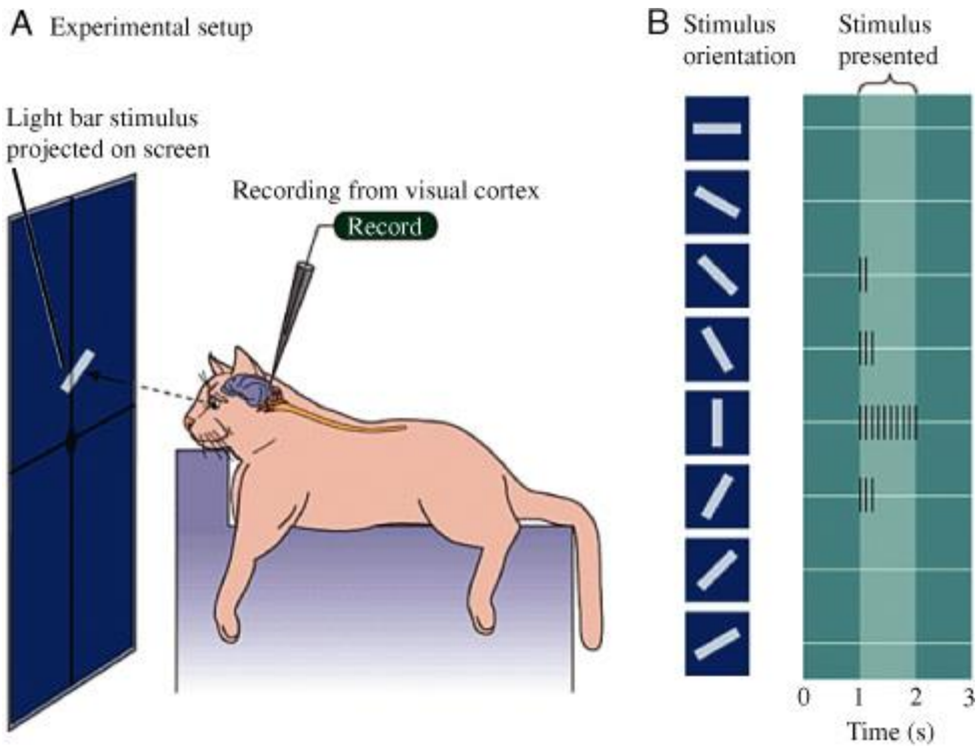
- 1974, 1982 by Paul Werbos, 1986 by Hinton



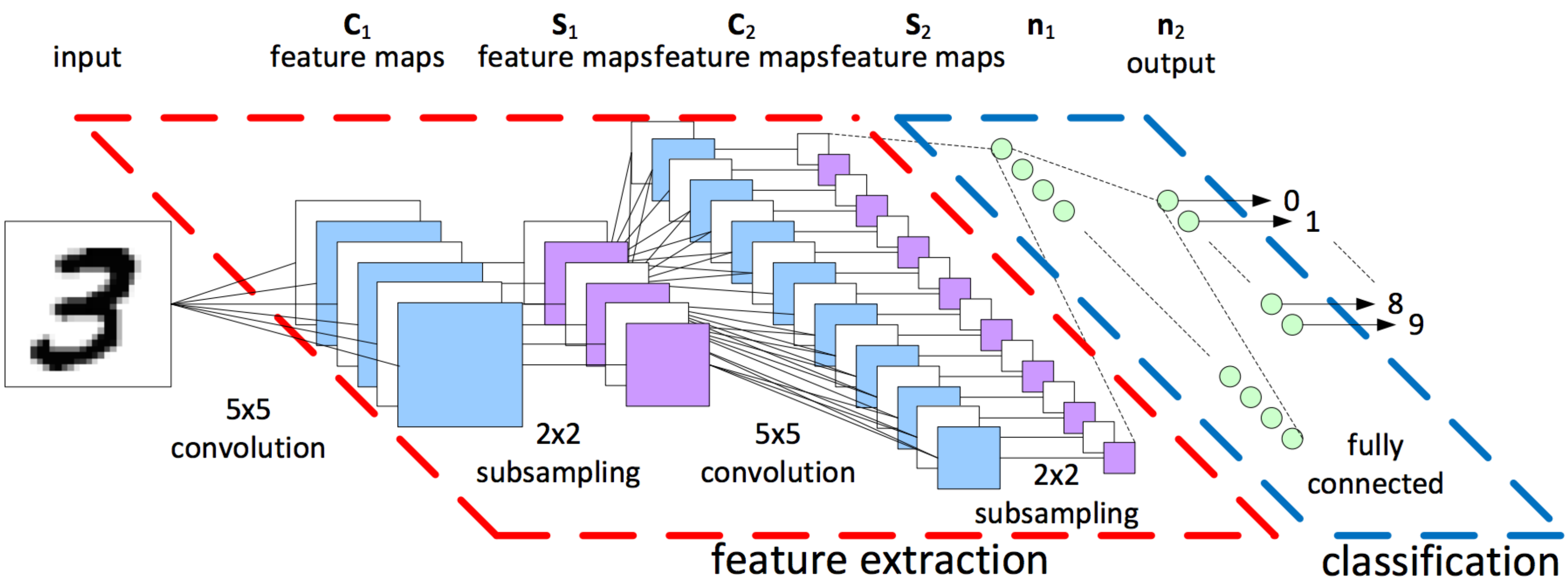
https://www.researchgate.net/figure/Figure-2-Back-propagation-multilayer-ANN-with-one-hidden-layer_241741756_fig2

06. Convolutional Neural Networks

- David Hubel and Torsten Wiesel, 1959

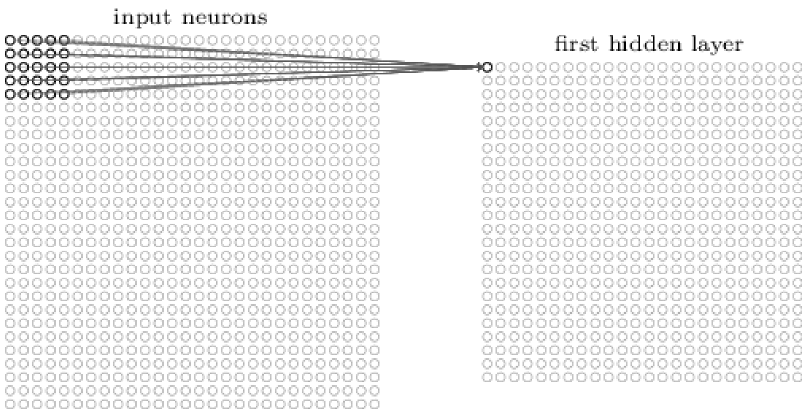


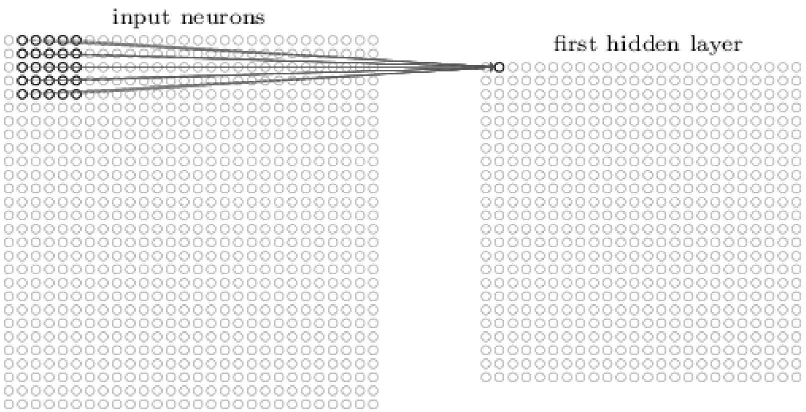
06. Convolutional Neural Networks

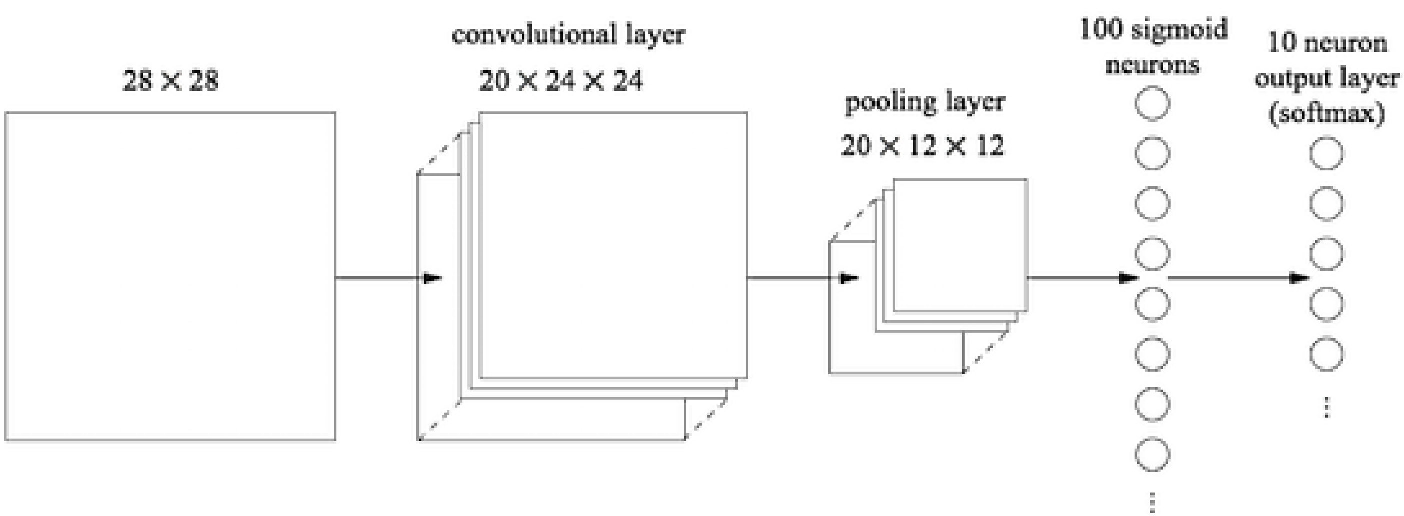


“At some point in the late 1990s,
one of these systems was reading 10 to 20% of all the checks in the US.”

06. Convolutional Neural Networks

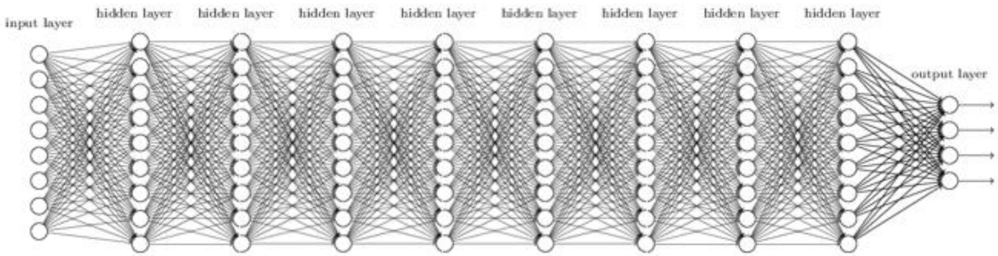




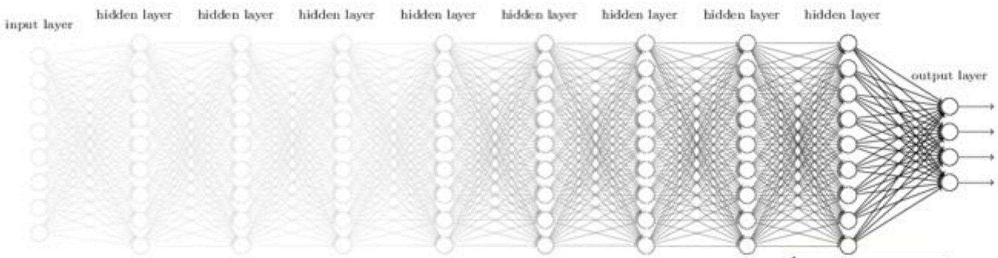


07. 두 번째 빙하기

- 신경망의 깊이가 깊어질수록 원하는 결과를 얻을 수 없다.



Deep Neural Network



Vanishing Gradient

Layer가 일정 개수 이상 늘어나면 기대하는 결과가 나오지 않는 현상 발견

⇒ Backpropagation을 수행할 때 입력층에서 멀리 떨어진 깊은 (deep) layer에 이르게 되면 기울기(gradient)가 급속히 작아지거나 너무 커져 발산해 버리는 **Vanishing Gradient** 문제가 발생

07. **두 번째 빙하기**

- 신경망 학습을 위한 파라미터 값의 최적화에 대한 이론적인 근거가 없었다.

⇒ 좋은 성능을 이끌어내기 위한 파라미터들에 대한 노하우는 있었지만 이론적인 근거가 없었다.

- 다른 머신런닝 알고리즘들의 등장

⇒ SVM(Support Vector Machine)이나 RandomForest 등 새로운 기계학습 알고리즘들이 등장

⇒ Neural network보다 새로운 기계학습 알고리즘의 성능이 더 좋음

08. Breakthrough

- 2006 년 Hinton, Simon Osindero 외 "깊은 믿음의 빠른 학습 알고리즘“
- 2007 년 Yoshua Bengio 외. "Deep Networks의 Greedy Layer-Wise Training“

- 가중치가 무작위보다는 영리한 방법으로 초기화되는 경우 많은 레이어가 있는 신경망을 실제로 잘 훈련 할 수 있다.
- 깊은 기계 학습 방법은 얕은 방법보다 어려운 문제에 더 효율적
- Rebranding to ***Deep Nets***, ***Deep Learning***

IMAGENET Large Scale Visual Recognition Challenge (ILSVRC) 2017

ILSVRC Image Classification (CLS) Task

Steel drum

Output:		Output:	
Scale		Scale	
T-shirt	✔	T-shirt	✗
Steel drum		Giant panda	
Drumstick		Drumstick	
Mud turtle		Mud turtle	

09. IMAGENET Large Scale Visual Recognition Challenge

- ILSVRC Image Localization (LOC) Task

Steel drum

09. IMAGENET Large Scale Visual Recognition Challenge

Steel drum

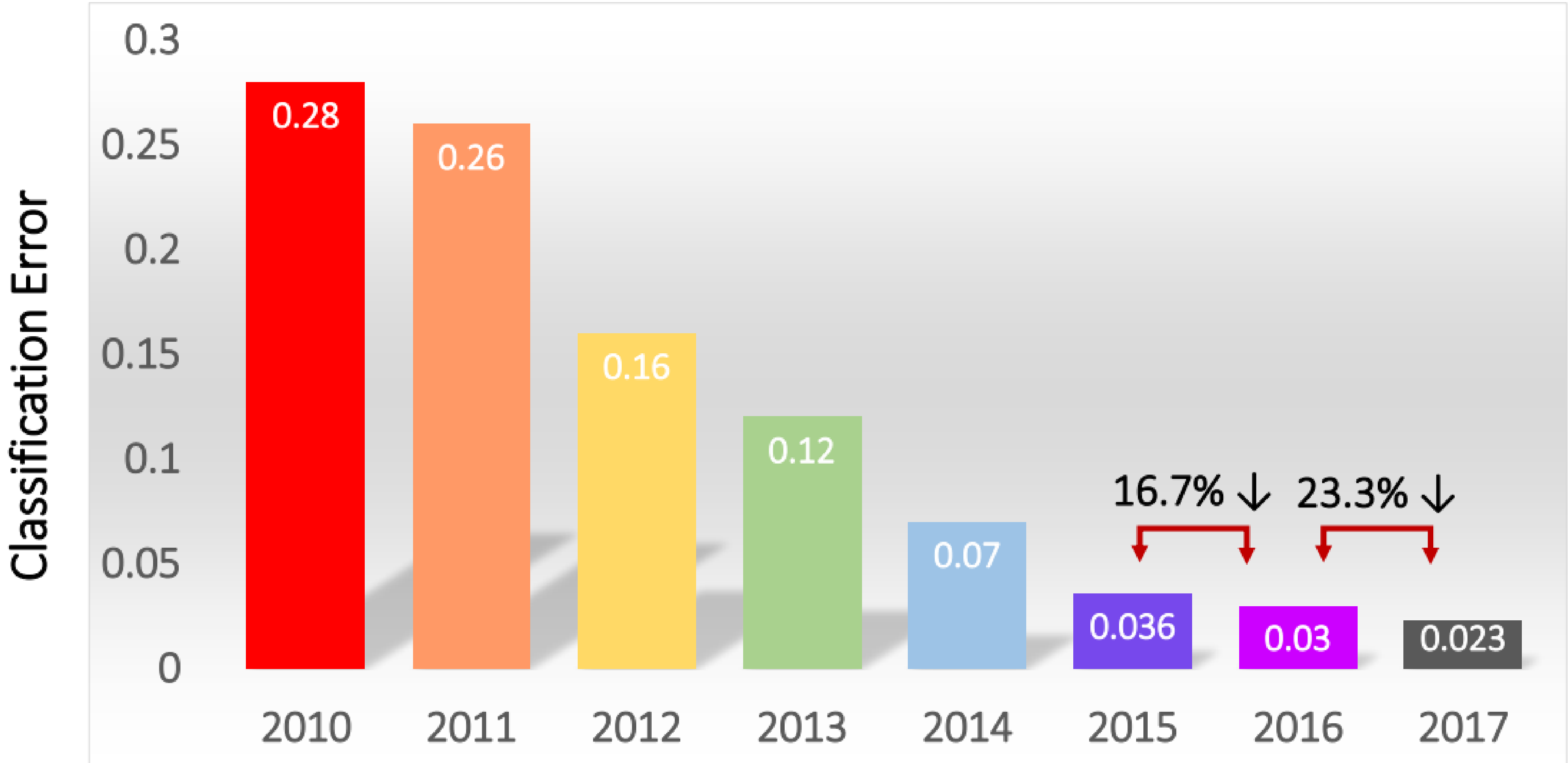
Correct

Bad localization

Bad classification

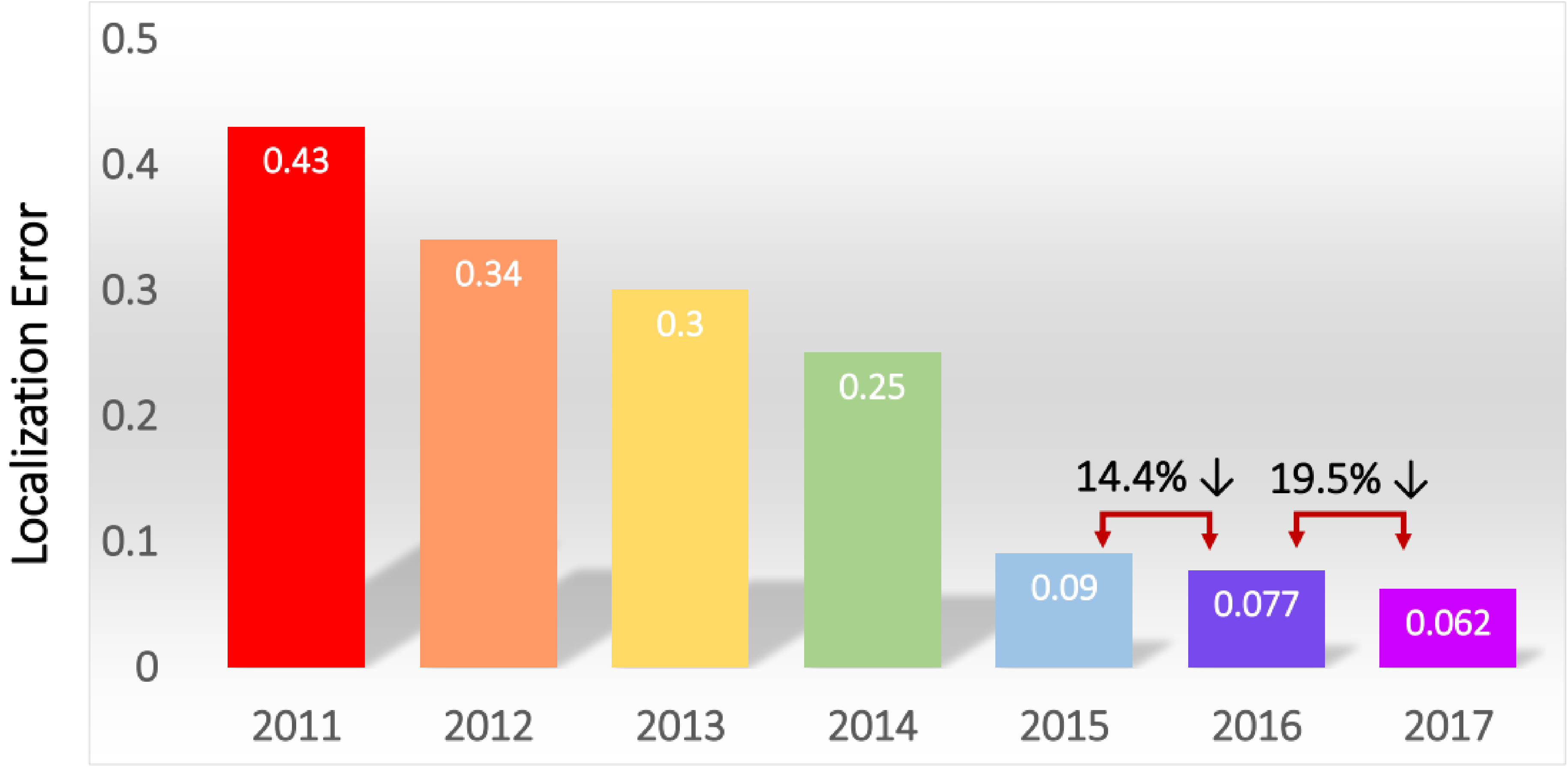
09. IMAGENET Large Scale Visual Recognition Challenge

- Classification Results (CLS)



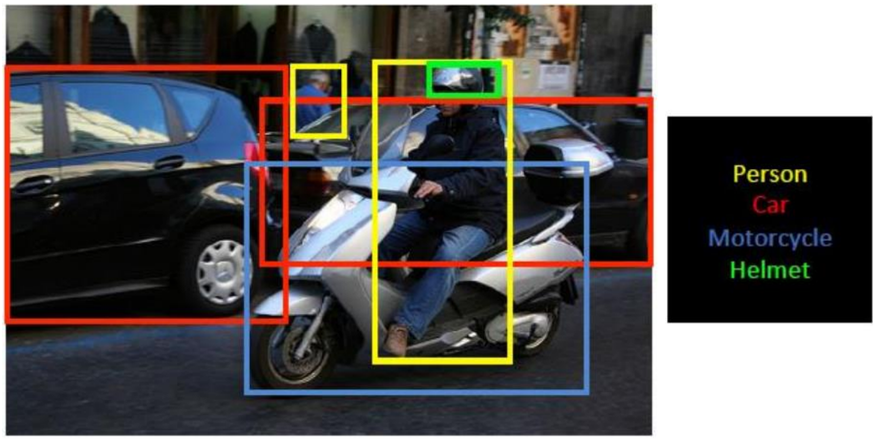
09. IMAGENET Large Scale Visual Recognition Challenge

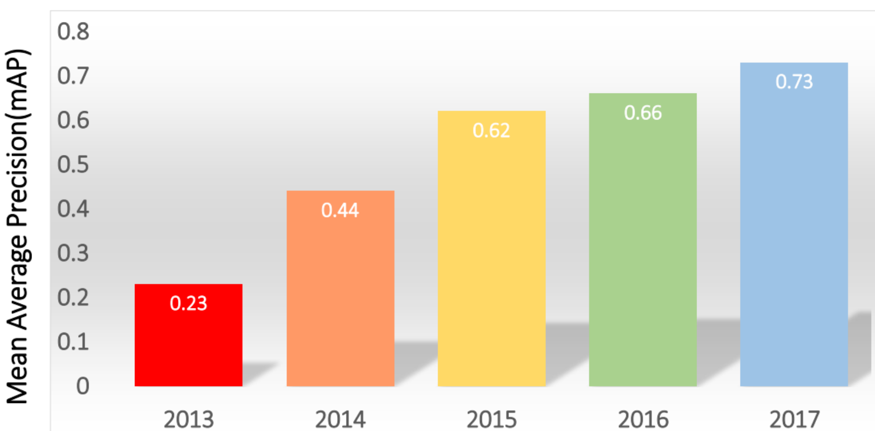
- Localization Results (LOC)



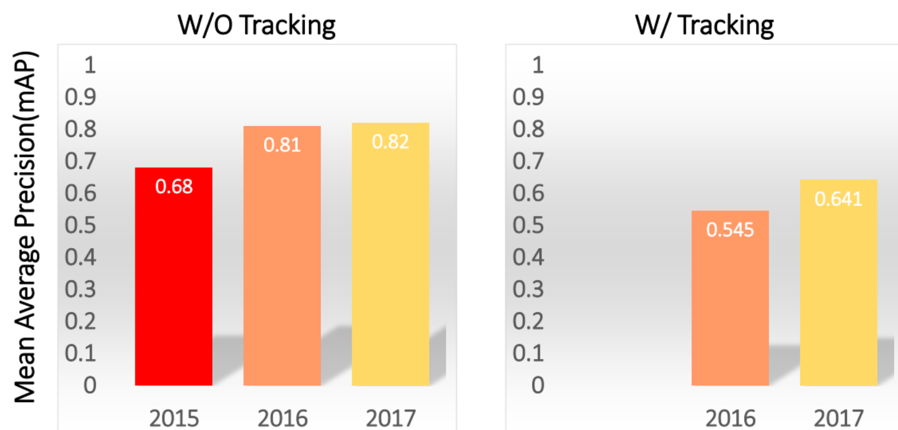
09. IMAGENET Large Scale Visual Recognition Challenge

- ILSVRC Object Detection (DET) Task



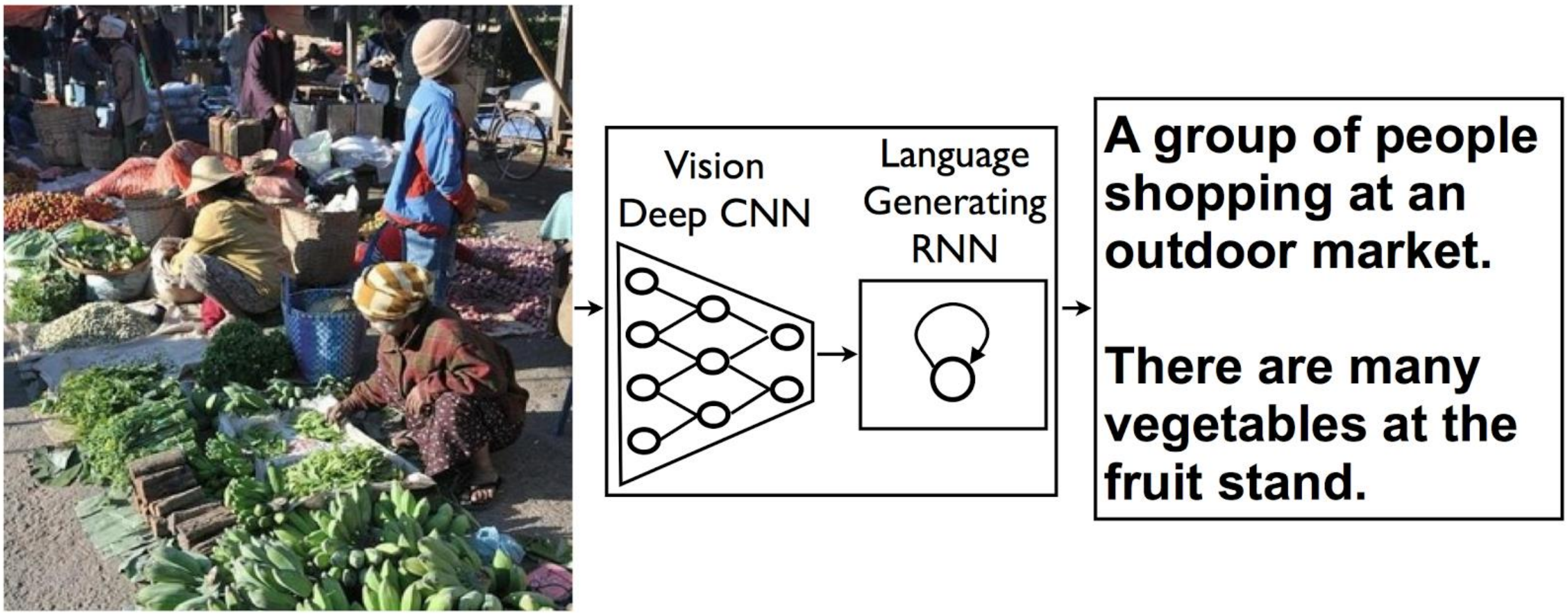


- Object Detection from Video(VID) Task



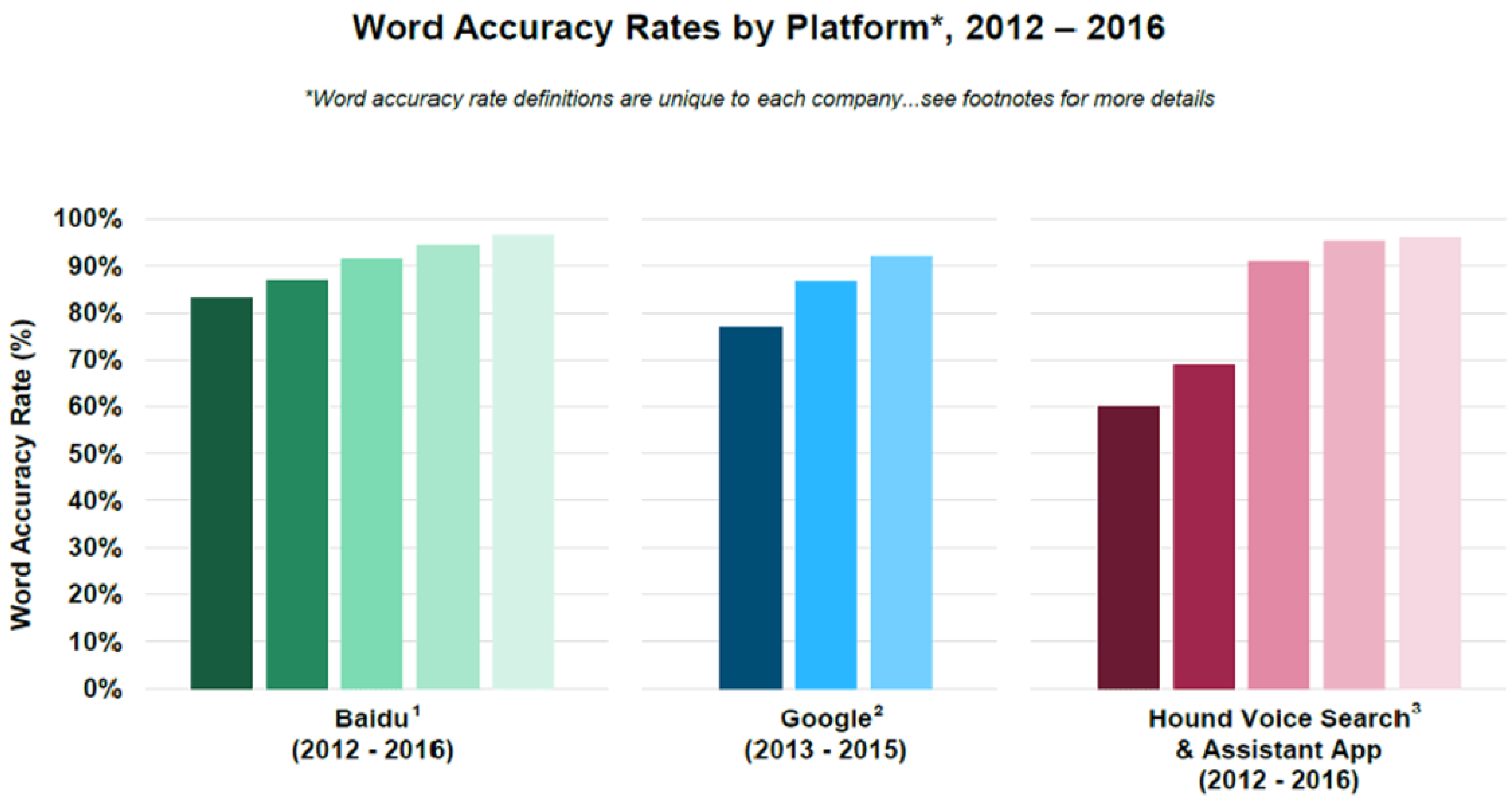
10. neural networks that can explain photos

- Google, Stanford build hybrid neural networks that can explain photos

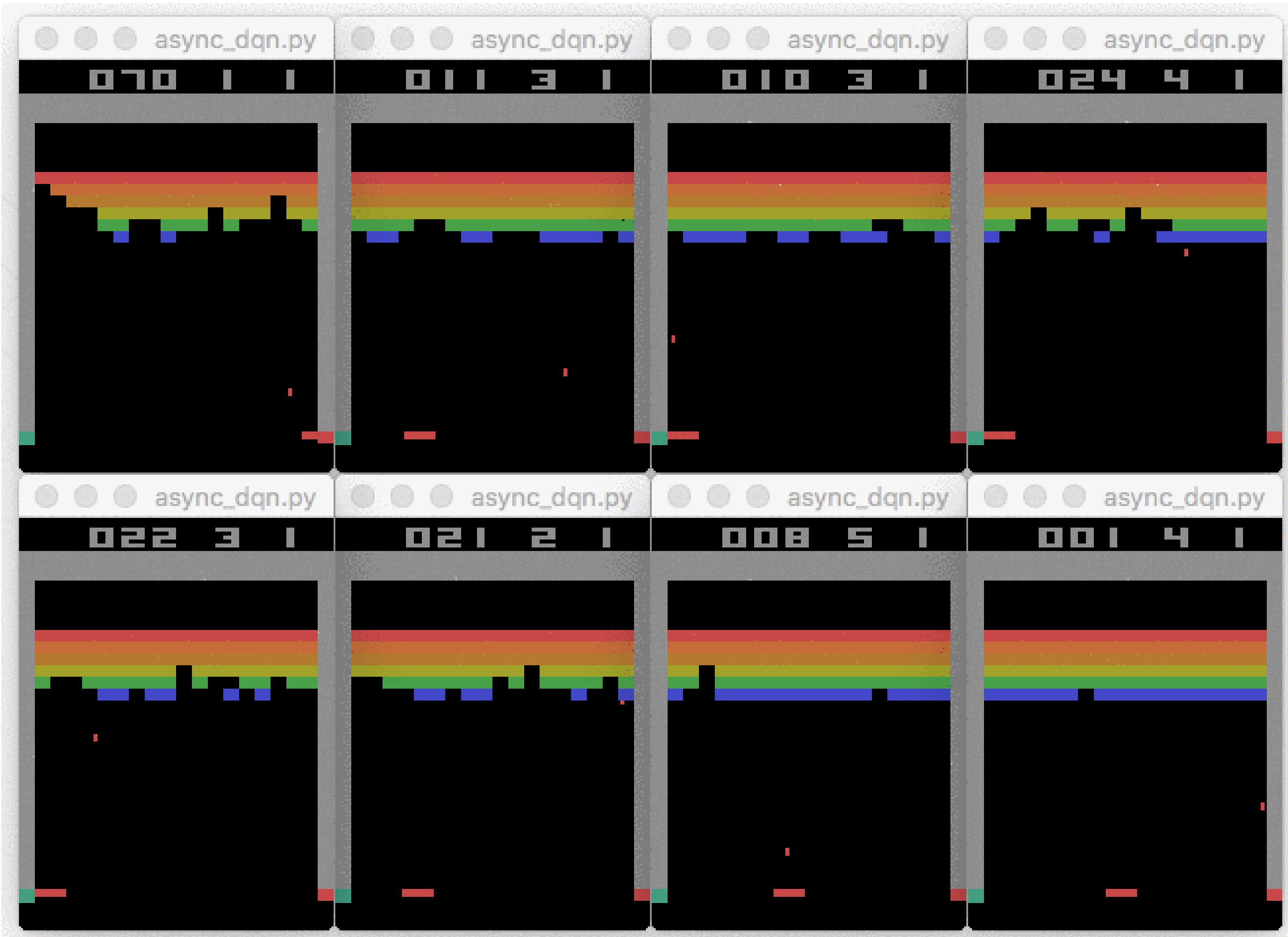


https://gigaom.com/2014/11/18/google-stanford-build-hybrid-neural-networks-that-can-explain-photos/

11. Speech Recognition



12. 강화학습


async_dqn.py
070 1 1
async_dqn.py
011 3 1
async_dqn.py
010 3 1
async_dqn.py
024 4 1
async_dqn.py
022 3 1
async_dqn.py
021 2 1
async_dqn.py
008 5 1
async_dqn.py
001 4 1

12. 강화학습

https://www.youtube.com/watch?v=-hMvDI_tLNk

12. 강화학습

https://youtu.be/exXD6wJLJ6s

13. Geoffrey Hinton's 지금까지의 발견요약

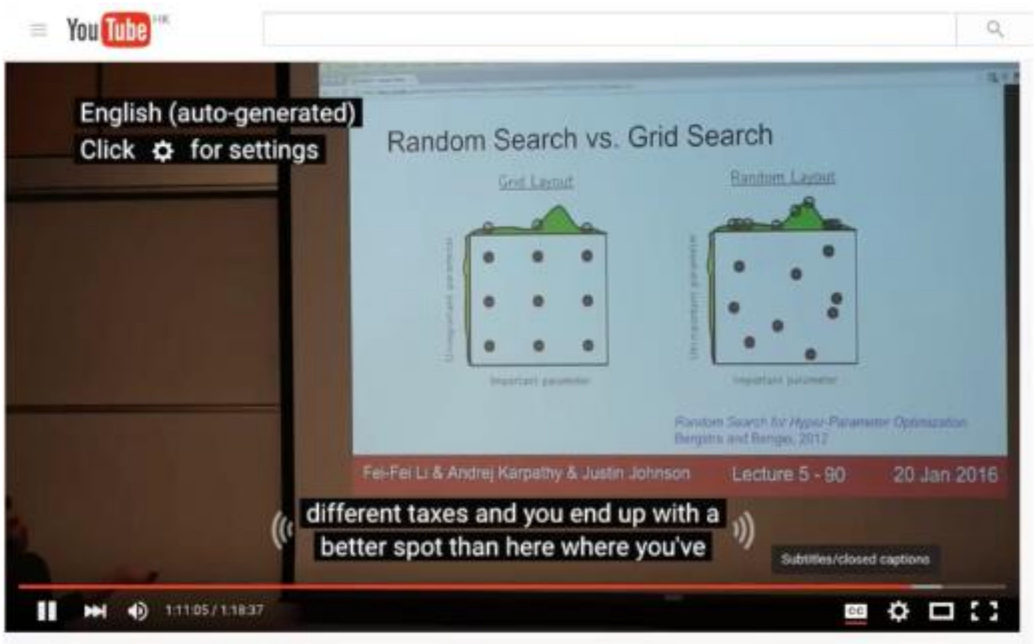
- labeled 데이터셋(학습용 데이터셋)이 너무 작았음
- 옛날엔 컴퓨터가 너무 느렸음
- 초기화가 부적절
- non-linearity(sigmoid)를 잘못 사용했다.

14. 왜 머신러닝을 공부해야하는가?

- 데이터를 갖고 있다면 누구라도 머신러닝을 활용할 수 있다.
- 연구자나 프로그래머가 아니라도 머신런닝을 활용가능

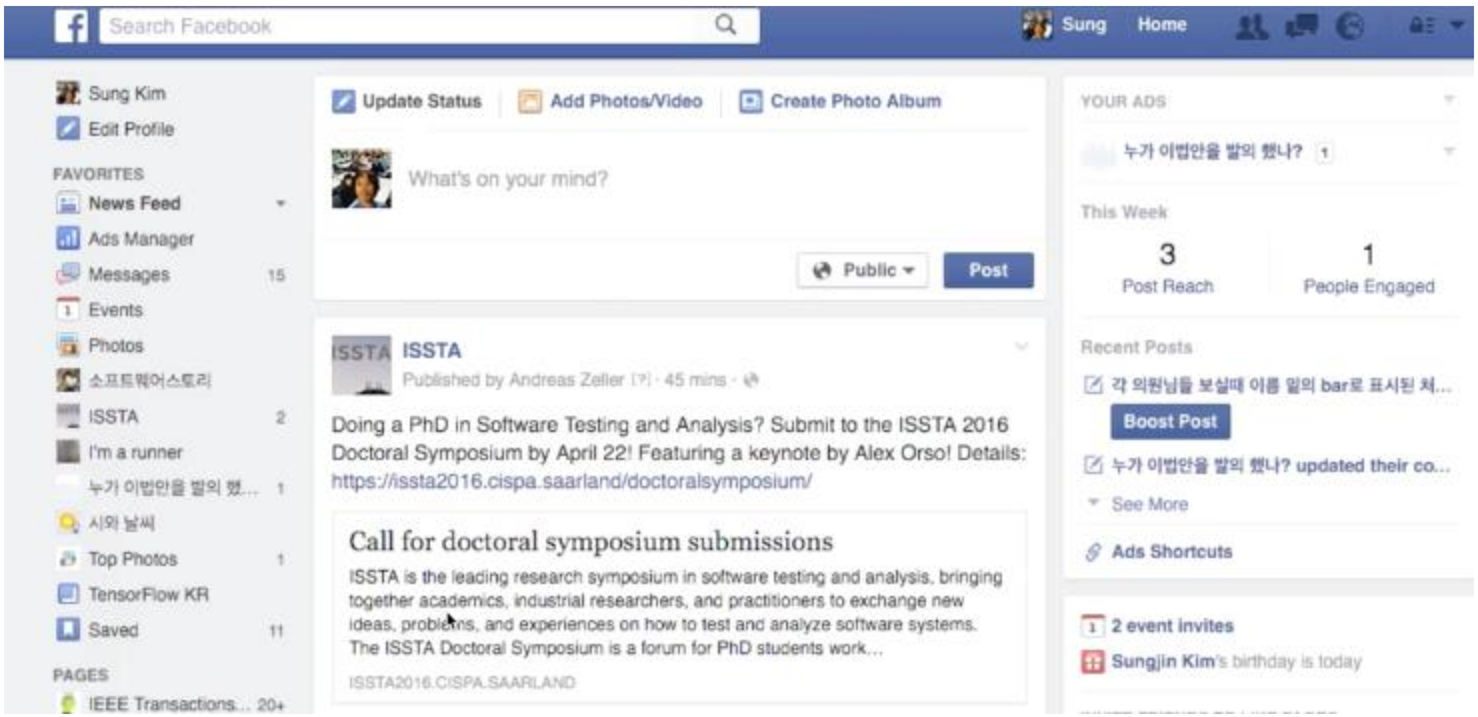
29.머신런닝의 활용

- 음성을 인식해서 자동으로 표시되는 유튜브 자막



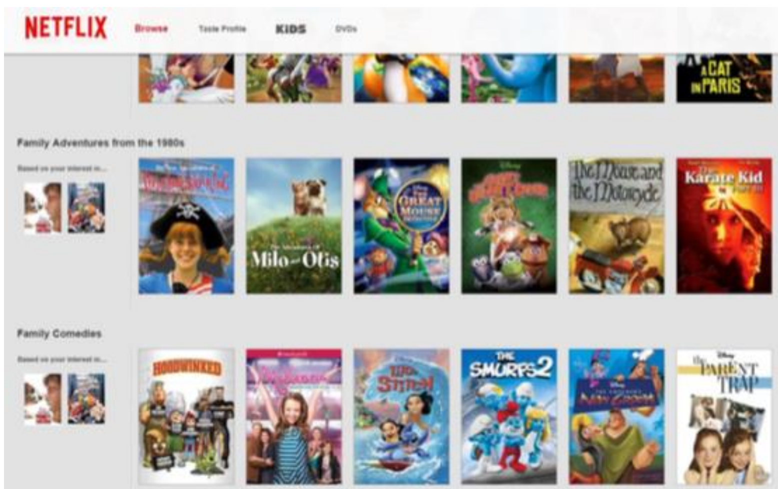
29.머신런닝의 활용

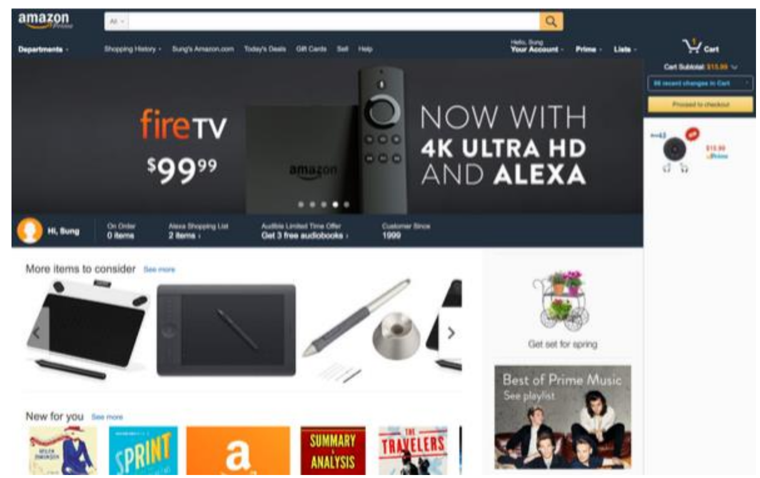
- 친구들이 올린 내용 중에서 내가 좋아할 만한 것만 추천하는 페이스북 뉴스피드 시스템.



29.머신런닝의 활용

- 구글 검색 시스템. 내가 찾고 싶은 것을 상위에 표시해 줌
- 내가 보고싶은 영화를 넷플릭스가 알려 줌
- 아마존의 제품추천 시스템





30. Why Now?

- Students/Researchers
  - Not too late to be a world expert
  - Not too complicated (mathematically)
- Practitioner
  - Accurate enough to be used in practice
  - many ready-to-use tools such as TensorFlow
  - Many easy/simple programming languages such as Python
- After all, it is fun!