Deep Learning (1)

윤명현

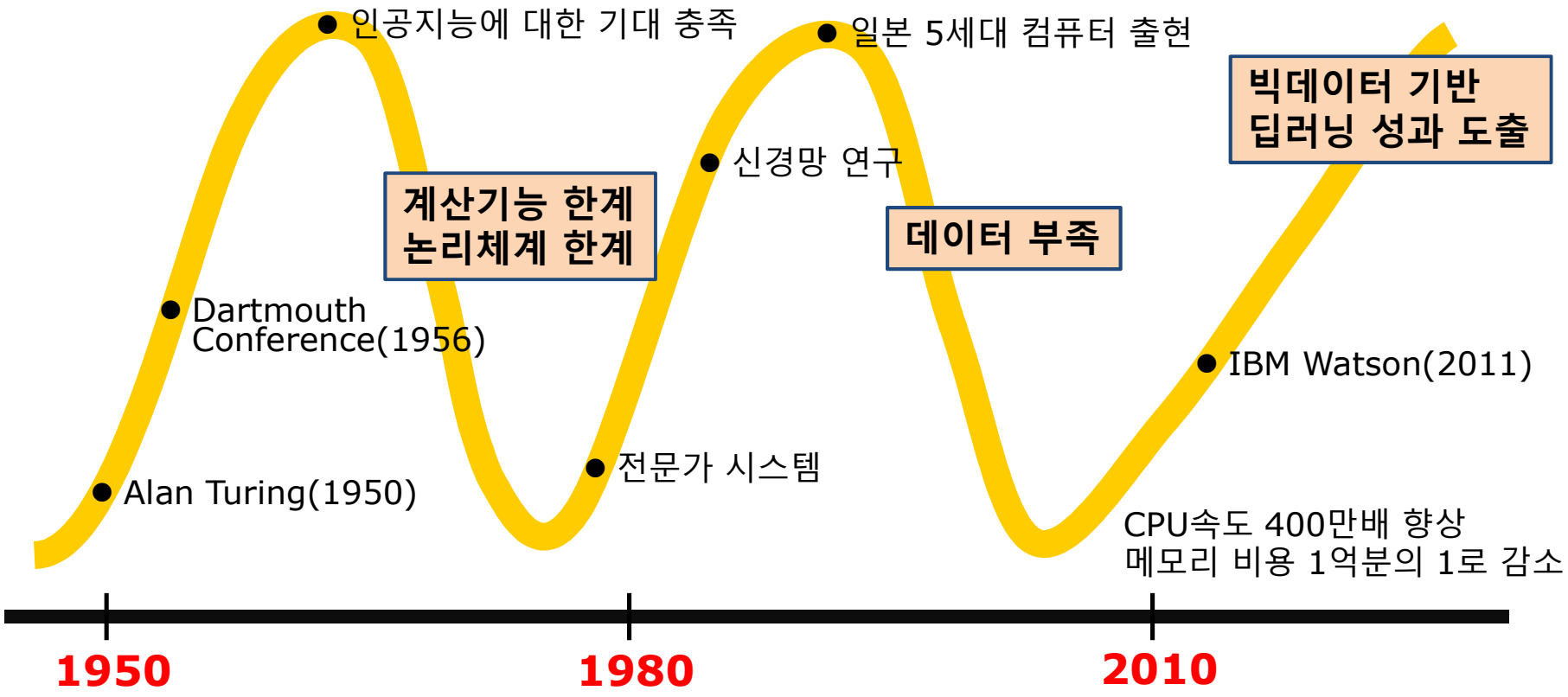
2020. 7.

목 차

History of AI

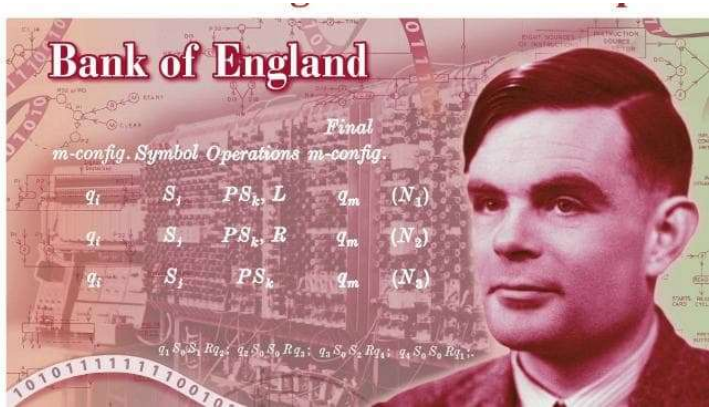
인공지능의 역사

2번의 빙하기와 3번째 부활

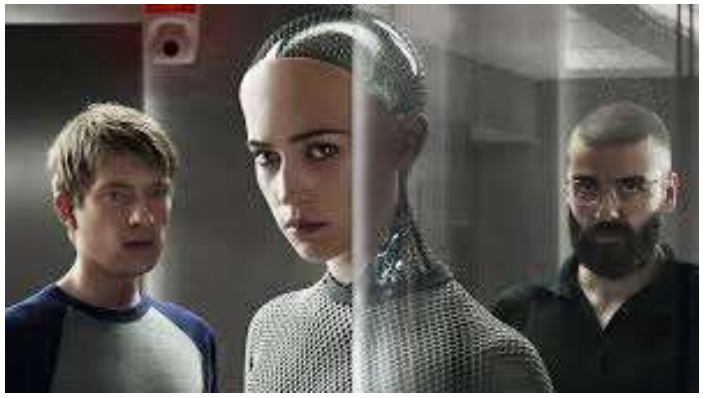


인공지능의 탄생

- Alan Turing(1912~1954)
  - "만약 컴퓨터가 인간을 속여 자신을 마치 인간인 것처럼 믿게 할 수 있다면 컴퓨터를 intelligent하다고 부를만한 가치가 충분히 있다."
  - Turing test



- Dartmouth Conference(1956)
  - Artificial Intelligence 용어 처음 사용
  - 최초의 컴퓨터 ENIAC 탄생 10년
  - 컴퓨터의 엄청난 계산력은 금새 인간의 능력을 능가할 것으로 생각



<Ex Machina(2015)>

1차 AI 붐

- 추론탐색의 시대(1950년대 후반~1960년대)
  - 추론, 탐색을 위한 단순한 룰로 인공지능 실현
  - Toy problem(미로, 퍼즐, 체스)은 풀려도 복잡한 현실의 문제는 풀지 못함
  - 추론탐색 문제
    - Othello 10^{60}
    - Chess 10^{120}
    - 바둑 10^{360}
    - 관측 가능한 우주 전체의 수소 원자 수 : 10^{80}

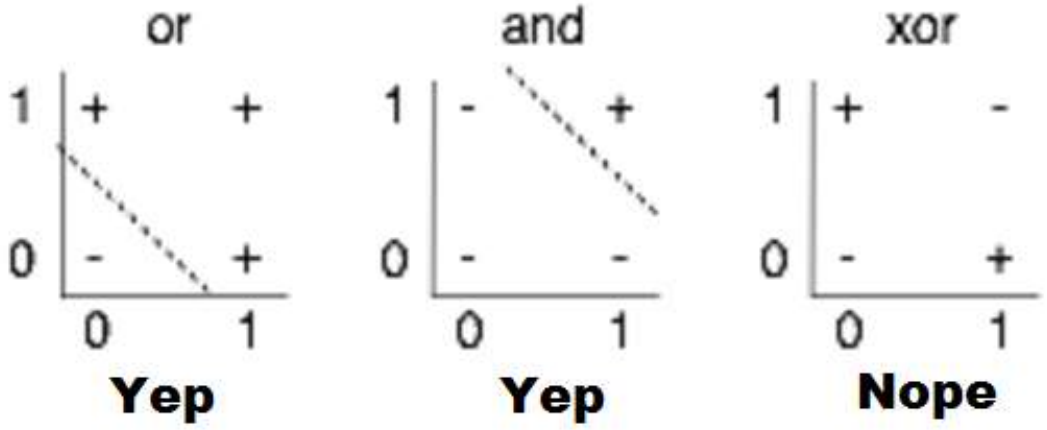
<하노이의 탑>

IBM Deep Blue(1997)

- Deep Blue
  - Chess-playing system developed by IBM
- Chess : 64 locations, 32 pieces
- Deep Blue vs. Garry Kasparov

인공지능의 한계

- 인공지능의 한계 봉착(1970년대)
  - 현실의 문제는 풀지 못함
  - Neural network의 한계 : XOR 구현 못함
  - ALPAC(Automatic Language Processing) 보고서(1966)
    - Machine translation은 당분간 성과가 나올 가망이 없음
  - 인간의 지능을 컴퓨터로 실현하는 것의 어려움



2차 AI 붐

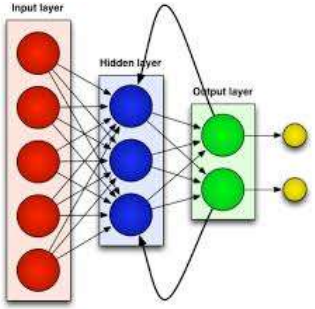
- 지식의 시대(1980년대)
- Expert system : 방대한 지식을 서술하고 관리
  - MYCIN(Stanford대학) : 69% 확률의 적합한 처방 (전문의는 80%)
  - DENDRAL : Edward Feigenbaum이 개발한 유기화합물 분석시스템
- 문제점
  - 지식의 비용
  - 지식과 룰이 늘어날수록 모순발생 (지식의 관리)
  - 넓은 지식을 기술하는데 어려움 (상식 수준의 지식 – 뜻밖의 난제)

지식표현 연구

- Semantic network
  - 인간이 의미를 기억할 때의 구조를 나타내기 위한 모델
  - 개념을 노드로 표시, 노드끼리 링크 – 의미 네트워크
- Ontology
  - 존재론
  - 개념화의 명시적인 사양(사양을 적은 설명서)
- Cyc Project
  - Douglas Lenart라는 사업가에 의해 1984년부터 지금까지 계속
  - 인간의 모든 일반 상식을 입력(인간 지식의 방대함, 기술의 어려움)
  - “Fred shaving in the morning”

3차 AI 붐

- 기계학습과 특징표현 학습의 시대
  - 1990년대 - 검색엔진과 인터넷의 폭발적 보급
    - 1990년 웹의 등장
    - 1993년 모자이크
    - 1998년 구글 검색엔진
  - 2000년대
    - 빅데이터 시대
    - 기계학습 및 딥러닝(특징표현 학습) 발전
    - Watson 같은 성공적인 사건



Deep Learning
기계의 자기학습

Big Data
방대한 학습자료

병렬컴퓨팅
강력한 GPU

AI Events

- Science publication of NIPS reject (2006)
- DARPA Urban Challenge (2007)
- Google Self-Driving Car Project (2009)
- Netflix Prize (2009)
- IBM Watson, Jeopardy에서 승리 (2011)
- AlexNet at ILSVR Challenge (2012)
- DARPA Robotics Challenge (2015)
- Google DeepMind의 AlphaGo (2016)

IBM Watson

- 질문응답 시스템
  - 2011년 Jeopardy에서 역대 우승자에 승리
  - 위키피디아 기반으로 Light Weight 온톨로지 생성, 그것을 정답에 사용

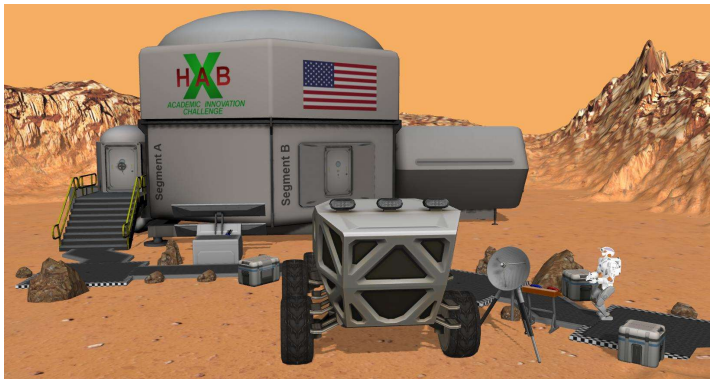
DARPA Grand Challenge

- Grand Challenge(2004, 2005)
  - 2004년 모하비사막 240km 구간에서 1회 대회 개최
  - 100개 참가팀중 완주팀없이 CMU 레드팀이 11.78km 주행
  - 2005년 2회 대회에서는 5개 팀이 240km 완주
  - Stanford 대학팀 우승, CMU 레드팀은 2위
- Urban Challenge(2007)
  - 도심을 가정하여 캘리포니아 조지 공군기지에서 개최
  - CMU와 GM이 합작한 Tartan Racing팀 우승

Robotics Challenge

- DARPA Robotics Challenge(2012~15)
  - 후쿠시마 원전사고를 계기로 만든 재난로봇 경연대회
  - 극한환경의 재난현장을 가상하여 8가지 임무 완수 목표
  - 최종결선에서 24팀중에 미션 8개를 완주한 팀은 3개
  - KAIST 휴보가 최종 우승 (오준호교수)

- NASA Space Robotics Challenge(2017)
  - 화성 우주기지에서 긴급 임무 수행
  - 발키리 로봇을 이용해 3D 로봇 시뮬레이터 Gazebo상에서 수행
  - Coordinated Robotics 우승
  - SRC Phase2 진행중



Netflix Prize

- Netflix Prize
  - 비디오 추천 시스템 대회, 상금 $100만 (2006.10.)
  - Netflix의 Cinematch 성능을 10% 이상 능가하면 우승
- Bellkor's Pragmatic Chaos 팀 우승 (2009)
  - Blended model (ensemble model)

NETFLIX

Netflix Prize

COMPLETED

Home | Rules | Leaderboard | Update

Leaderboard

Showing Test Score. Click here to show quiz score

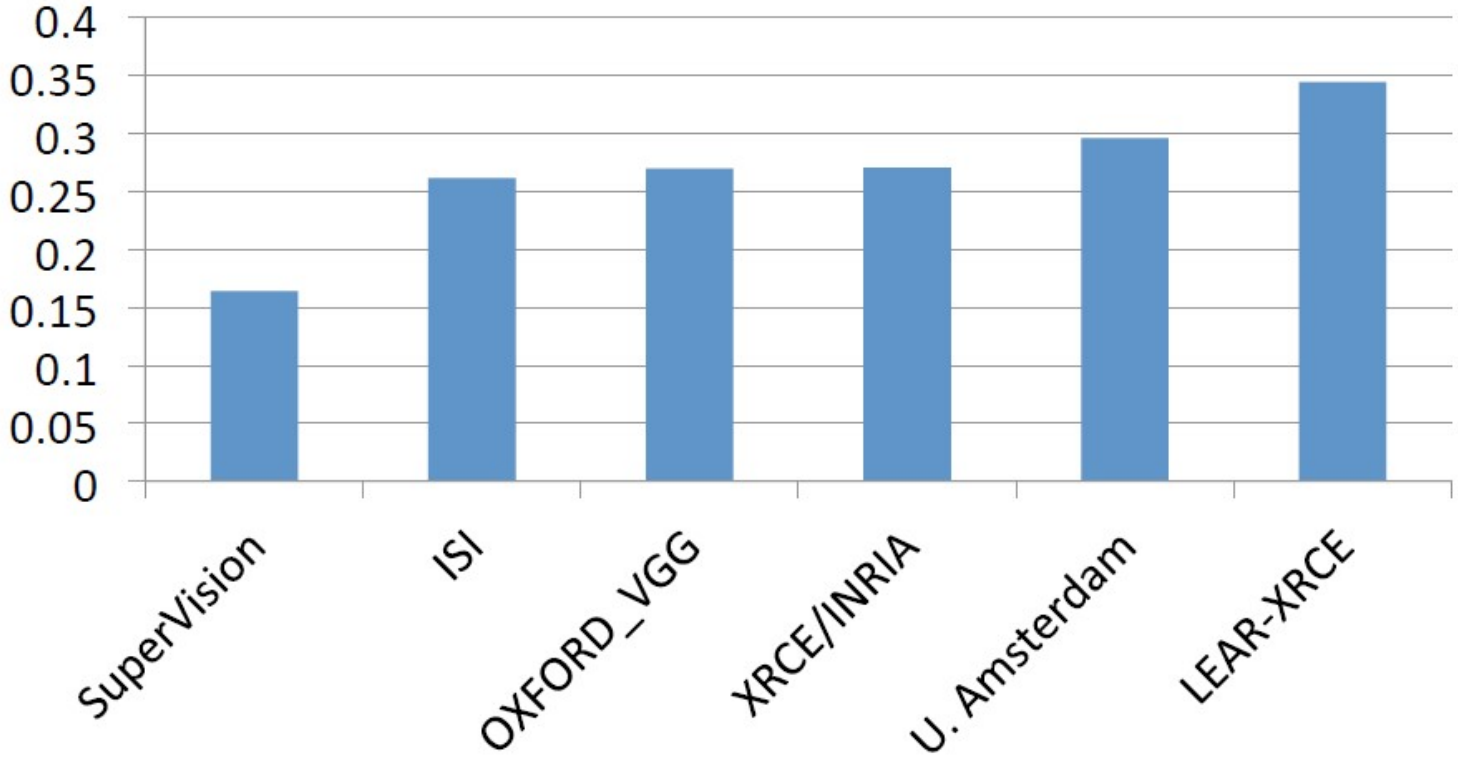
Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Opera Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries !	0.8591	9.81	2009-07-10 00:32:20
6	PragmaticTheory	0.8594	9.77	2009-06-24 12:06:56
7	BellKor in BigChaos	0.8601	9.70	2009-05-13 08:14:09
8	Dace	0.8612	9.59	2009-07-24 17:18:43
9	Feeds2	0.8622	9.48	2009-07-12 13:11:51
10	BigChaos	0.8623	9.47	2009-04-07 12:33:59
11	Opera Solutions	0.8623	9.47	2009-07-24 00:34:07
12	BellKor	0.8624	9.46	2009-07-26 17:19:11

ImageNet Challenge

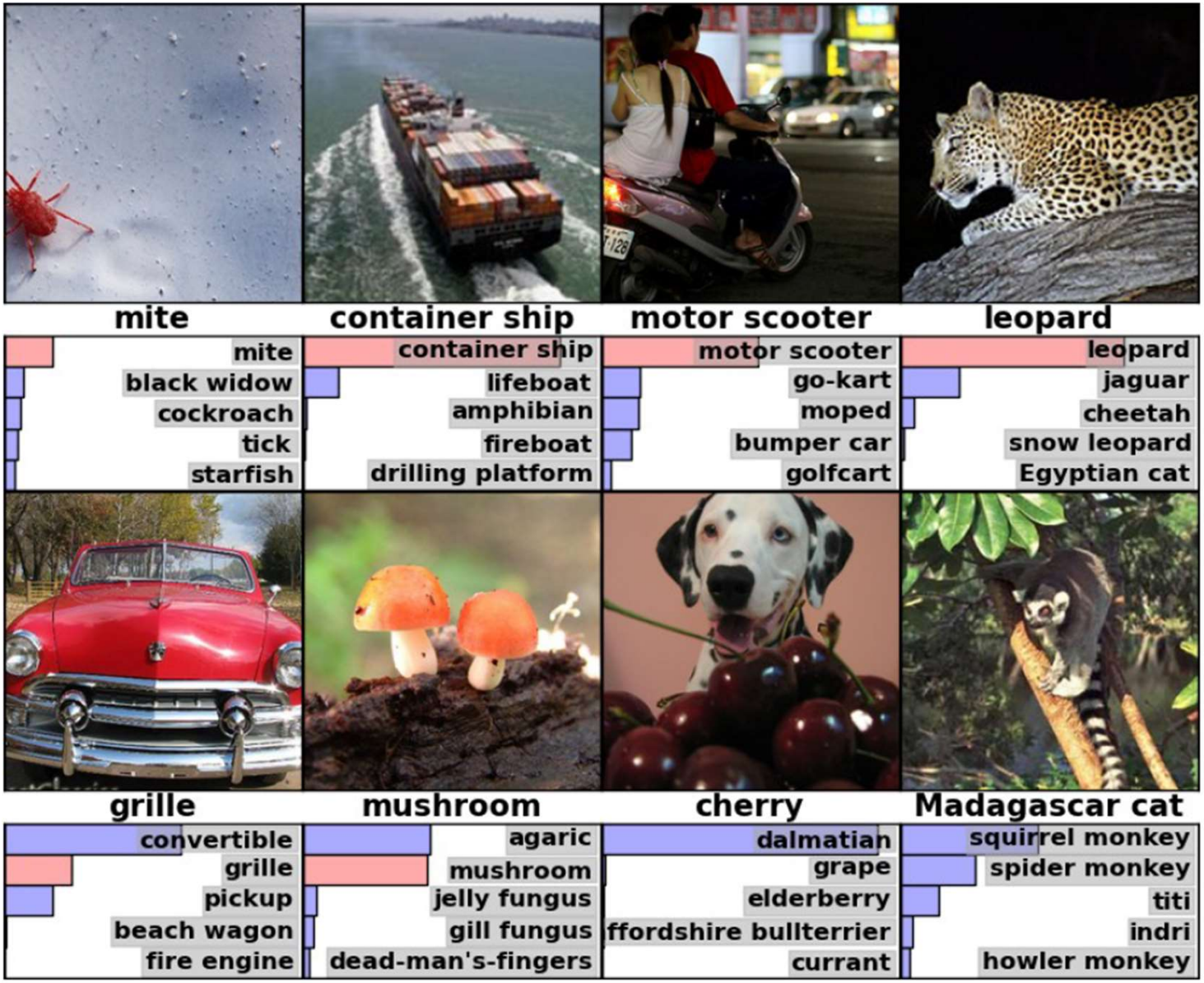
- ImageNet
  - Image database organized according to the WordNet hierarchy
- ILSVR(ImageNet Large Scale Visual Recognition) Challenge
  - Complement PASCAL(Pattern Analysis, Statistical modeling and Computational Learning) VOC(Visual Object Classes) Challenge (2005~2012)
  - Provide a benchmark dataset
  - Let promising techniques emerge via competition

ILSVRC 2012

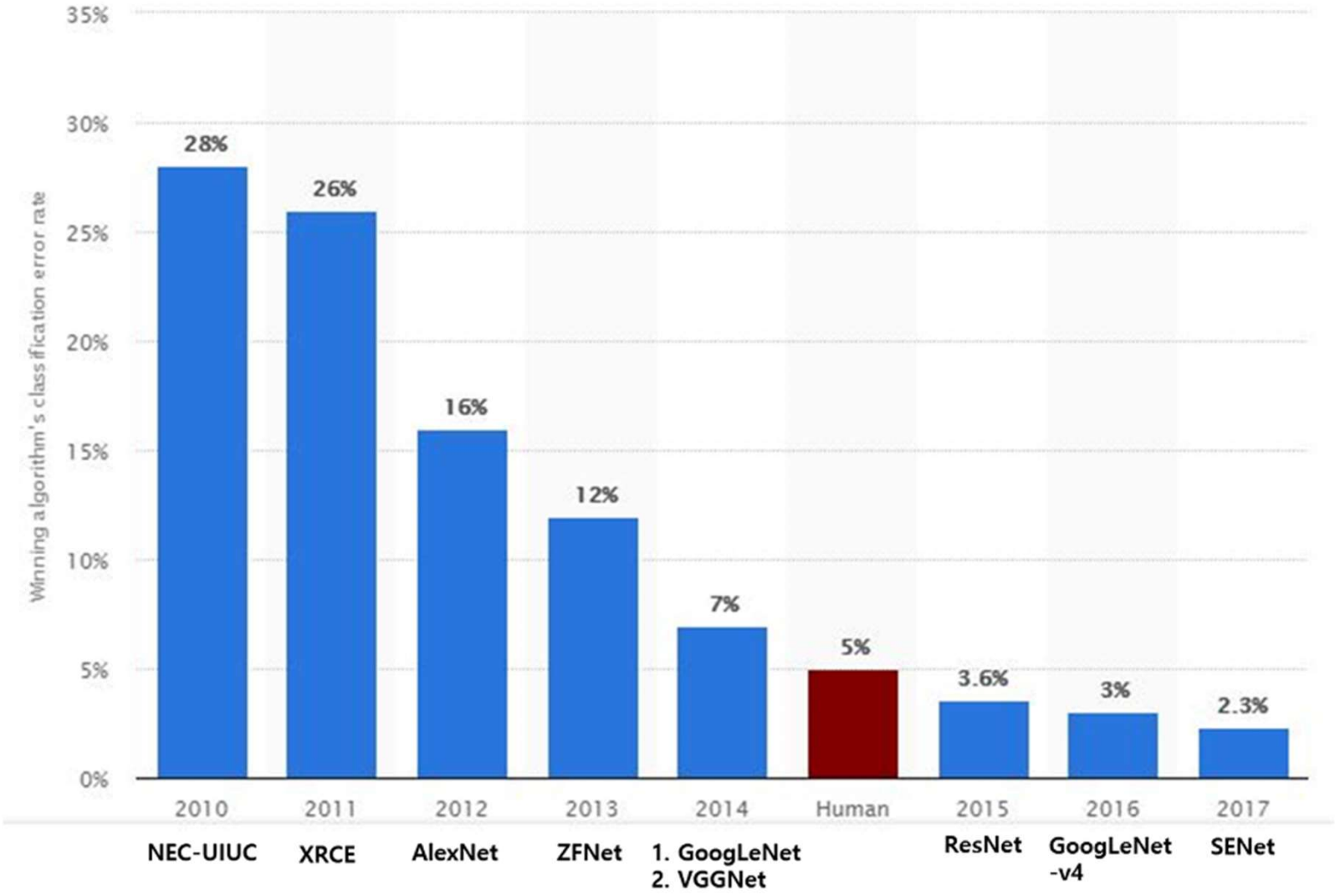
- Winner
  - Supervision (A. Krizhevsky, I. Sutskever, G. Hinton, Univ. of Toronto)



Classification Task



ILSVRC 2010~2017

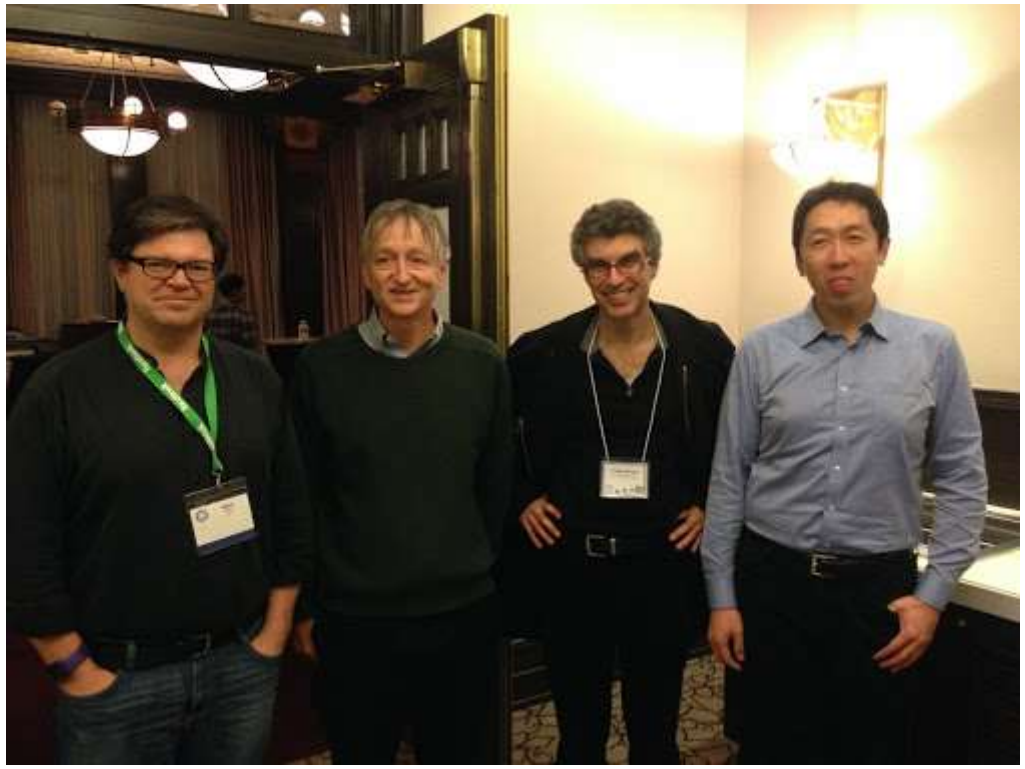


CIFAR

- "Canadian Institute for Advanced Research"
- CIFAR, which encourages basic research without direct application, was what motivated Hinton to move to Canada in 1987
- CIFAR "had a huge impact in forming a community around deep learning" (LeCun)

AI 4대 천왕

Yann LeCun
- NYU 교수
- Facebook AI 연구소



Yoshua Bengio
- Montreal대 교수
- IBM과 공동연구

Goeff Hinton
- Toronto대 교수
- Google 석학연구원

Andrew Ng
- Stanford대 교수
- BAIDU AI연구소

AlphaGo

- Google DeepMind사가 개발한 바둑 인공지능
  - 판후이에 5:0 승 (2015.10.)
  - 이세돌에 4:1 승 (2016.3.)
  - 온라인 바둑 60연승 (커제와 비공식 대결에서 3:0 승)
  - 커제 9단에 3:0 승 (The Future of Go Summit, 2017.5.)

"We hope that in the future by collaborating with human scientists we will make a greater progress in all sorts of things"

Demis Hassabis

AlphaGo Zero

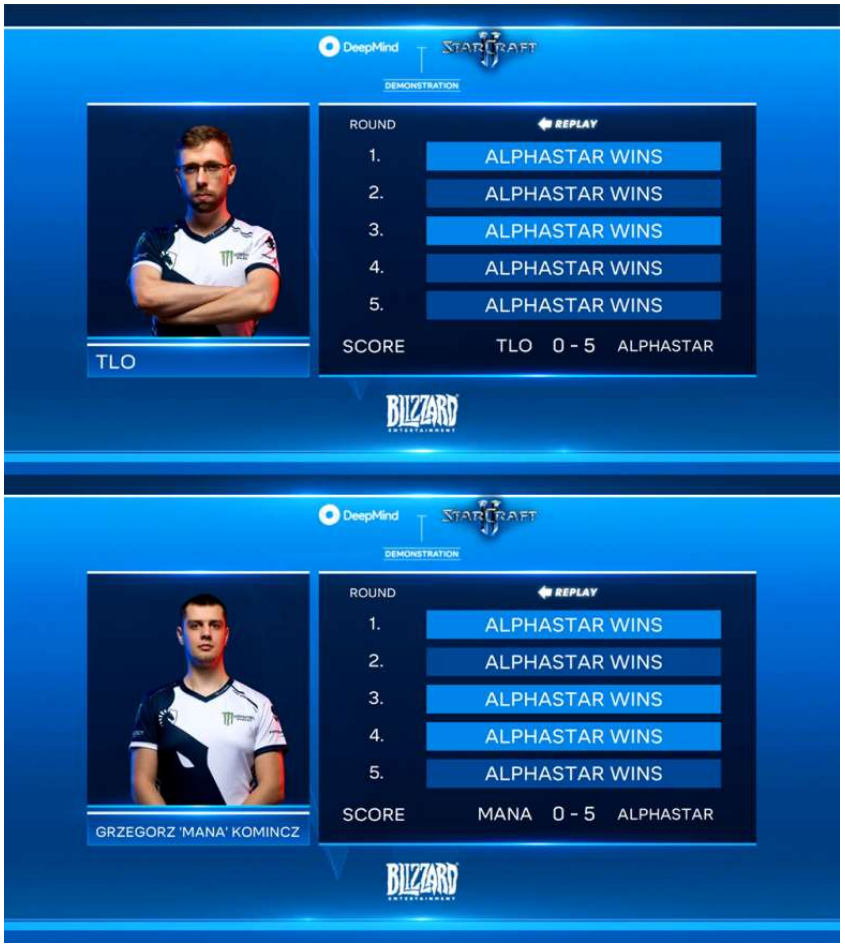
- "Mastering the game of Go without human knowledge" (2017.10., Nature)
  - 사람의 도움없이 스스로 학습하는 인공지능(강화학습)

89:11

100:0

AlphaStar

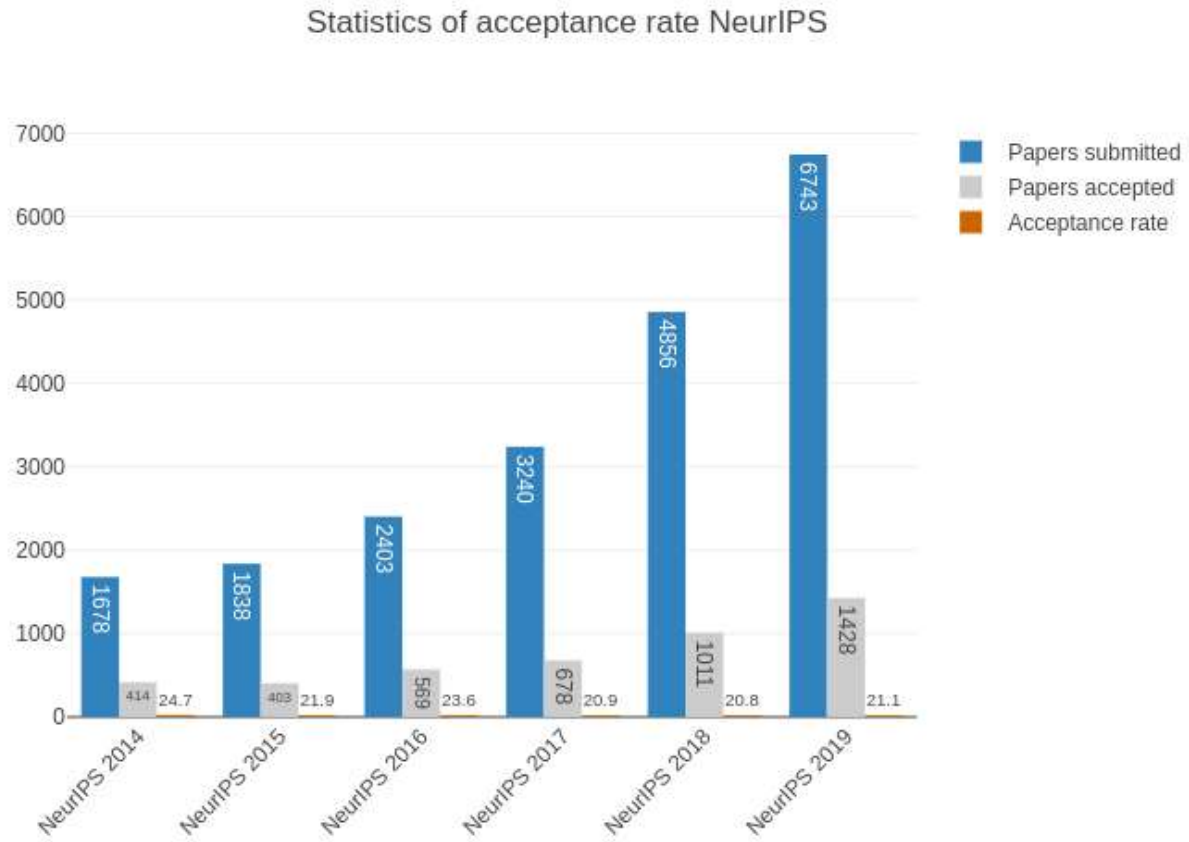
- StarCraft2 AI
  - 블리자드와 협력, 온라인 전략 게임 도전
  - 바둑보다 어려운 종목
    - 경우의 수
    - 진행속도
    - 보이지 않는 상황
- 시범경기 공개(2019.1.25.)
  - TLO 5:0
  - MaNa 5:1



DeepMind의 목표 : 전용 인공지능 → 범용 인공지능

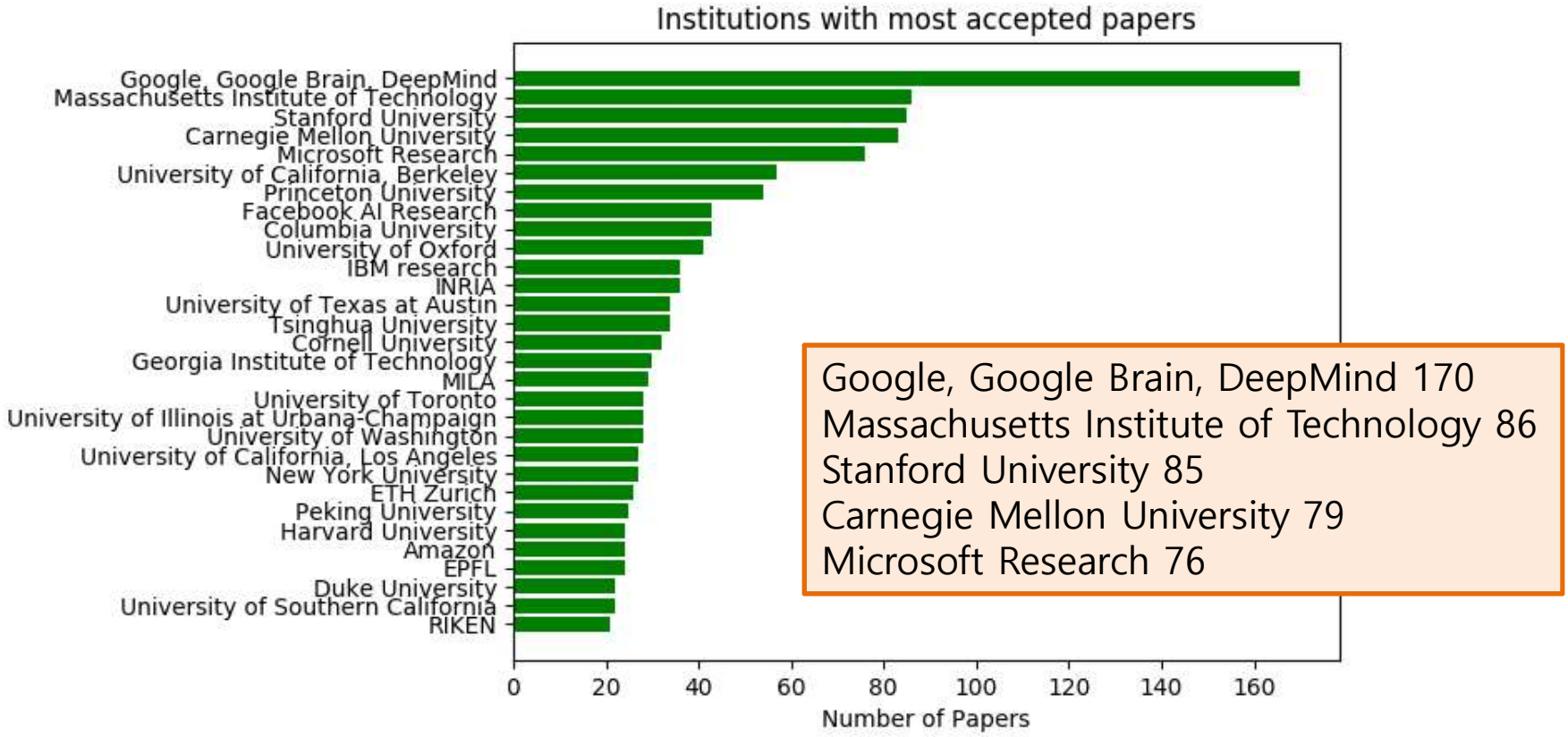
NIPS 2019

- NeurIPS(Neural Information Processing Systems)
  - 1986년 Denver에서 1회 대회 개최
  - 33회 NIPS 2019 캐나다 뱅쿠버에서 열림(2019.12.)



NIPS 2019

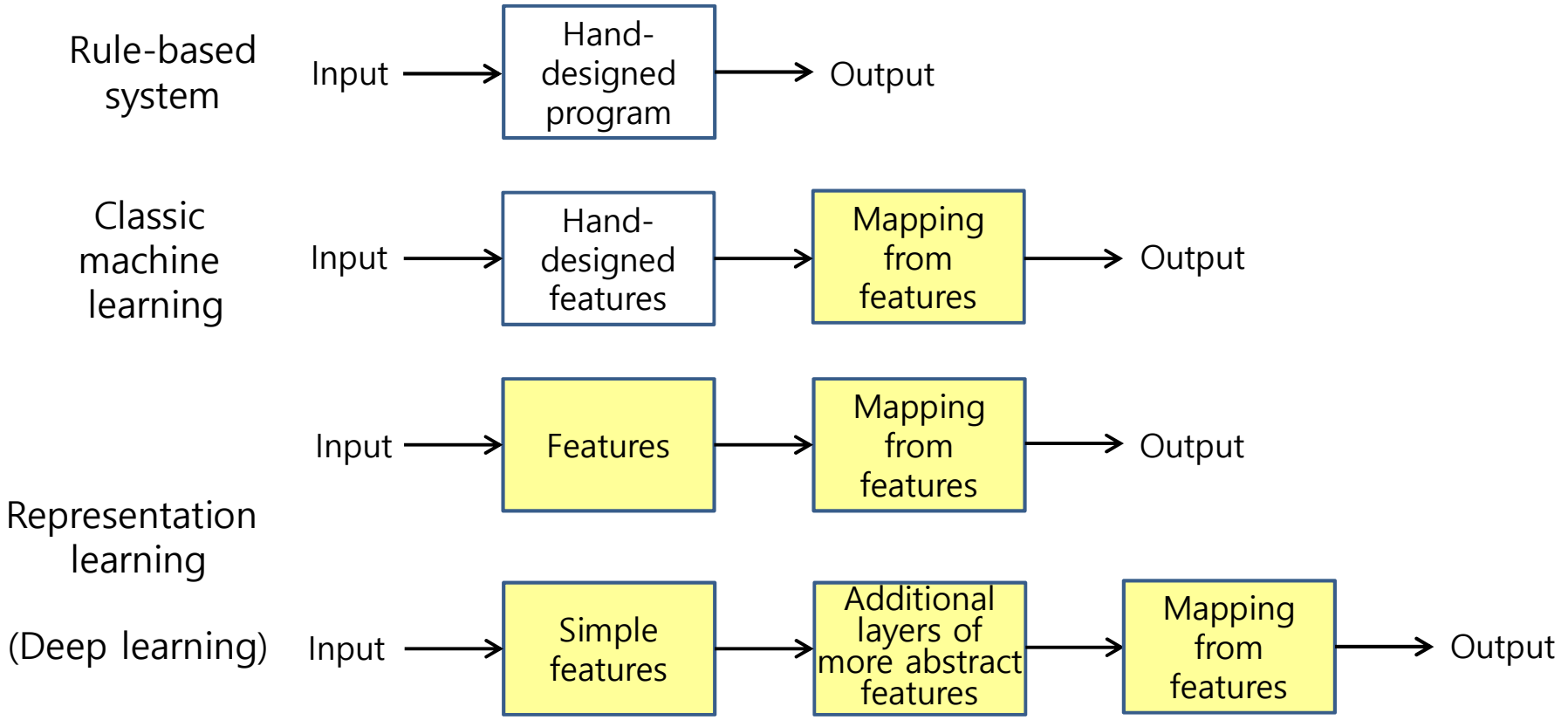
- Accepted papers : 1,428 (6,743 submitted, 21.2%)
  - Oral talks : 36 (0.5%)
  - Spotlights : 164 (2.4%)
  - Posters : 1,228 (18.2%)



Google, Google Brain, DeepMind 170
Massachusetts Institute of Technology 86
Stanford University 85
Carnegie Mellon University 79
Microsoft Research 76

Machine Learning

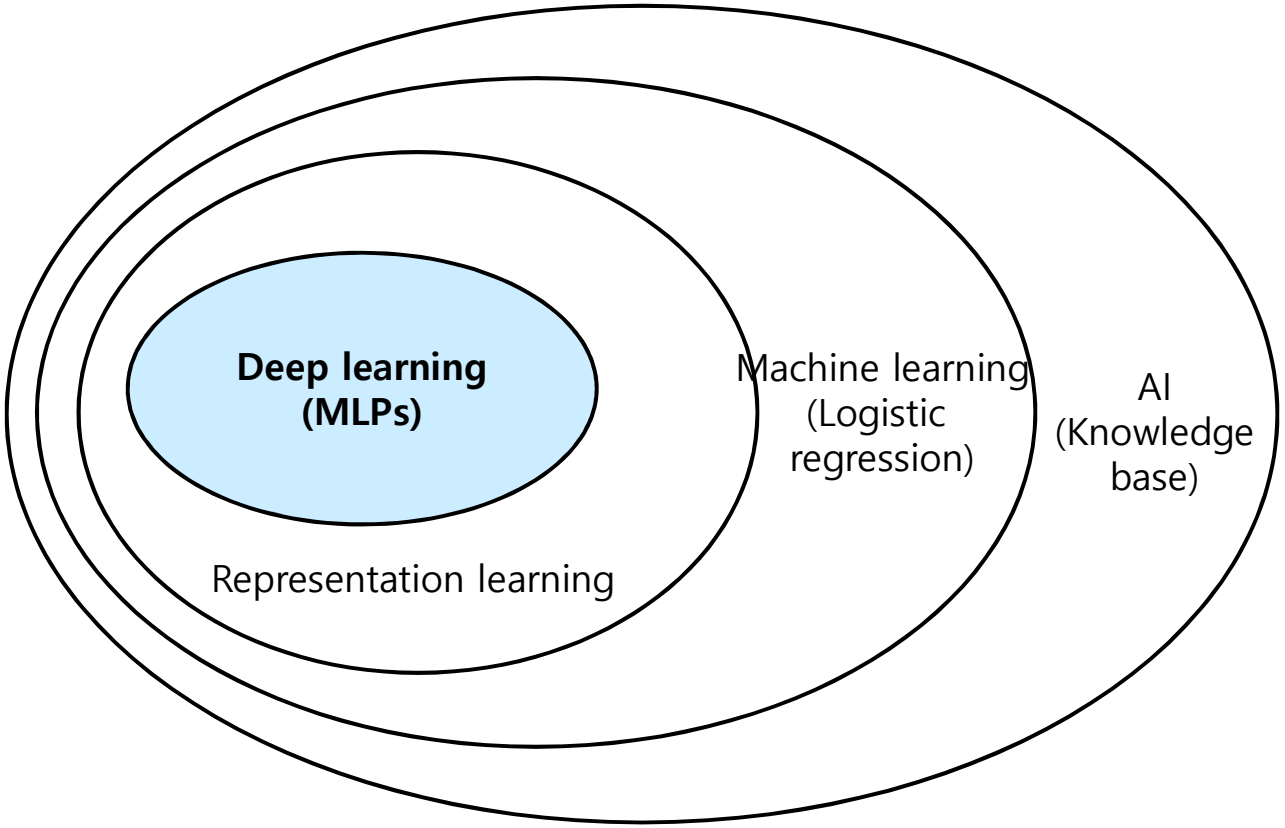
AI Systems



<http://www.deeplearningbook.org>

Deep Learning

- Machine learning에 비해 많은 수의 학습함수와 개념을 가짐

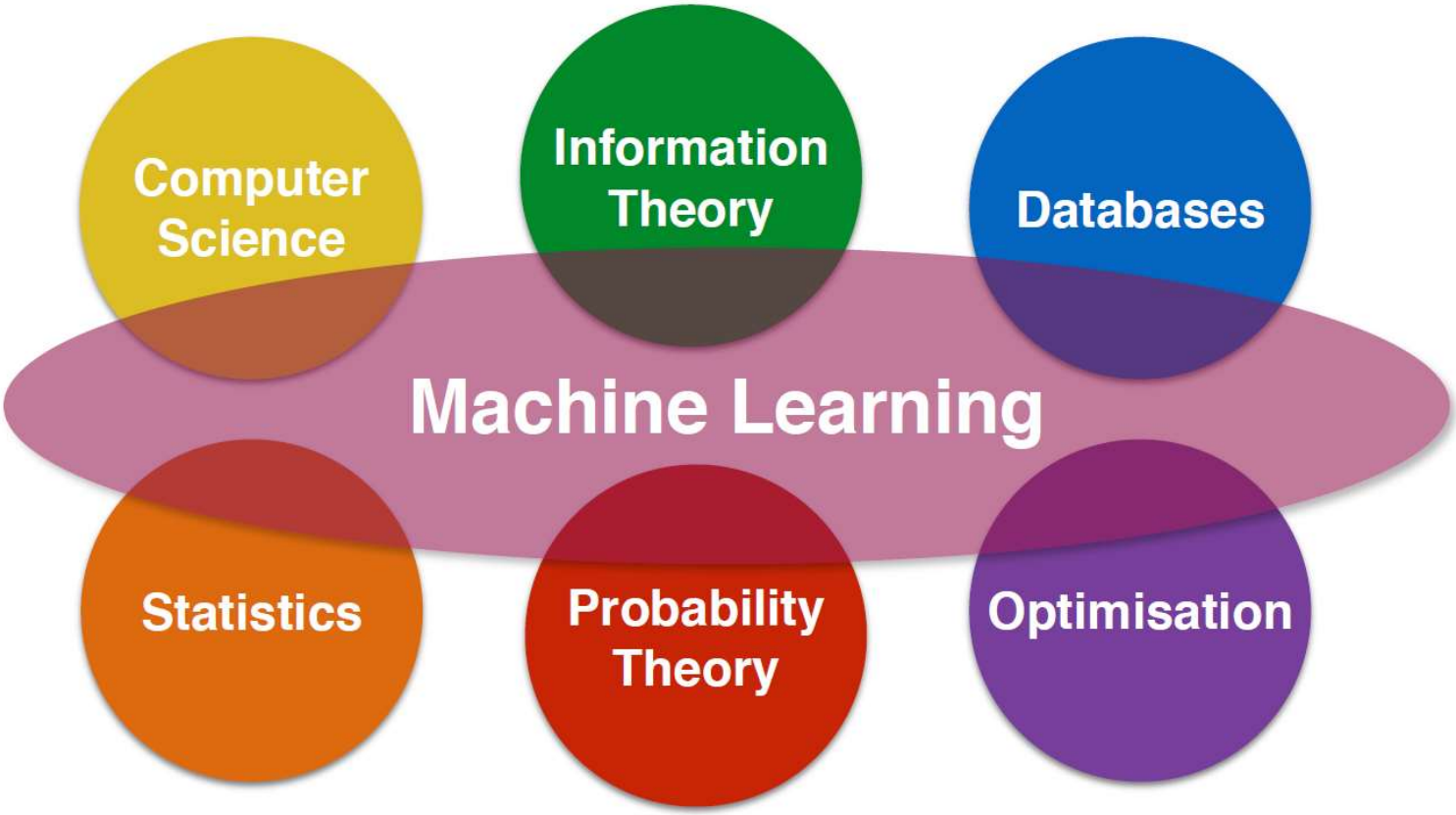


<http://www.deeplearningbook.org>

Machine Learning?

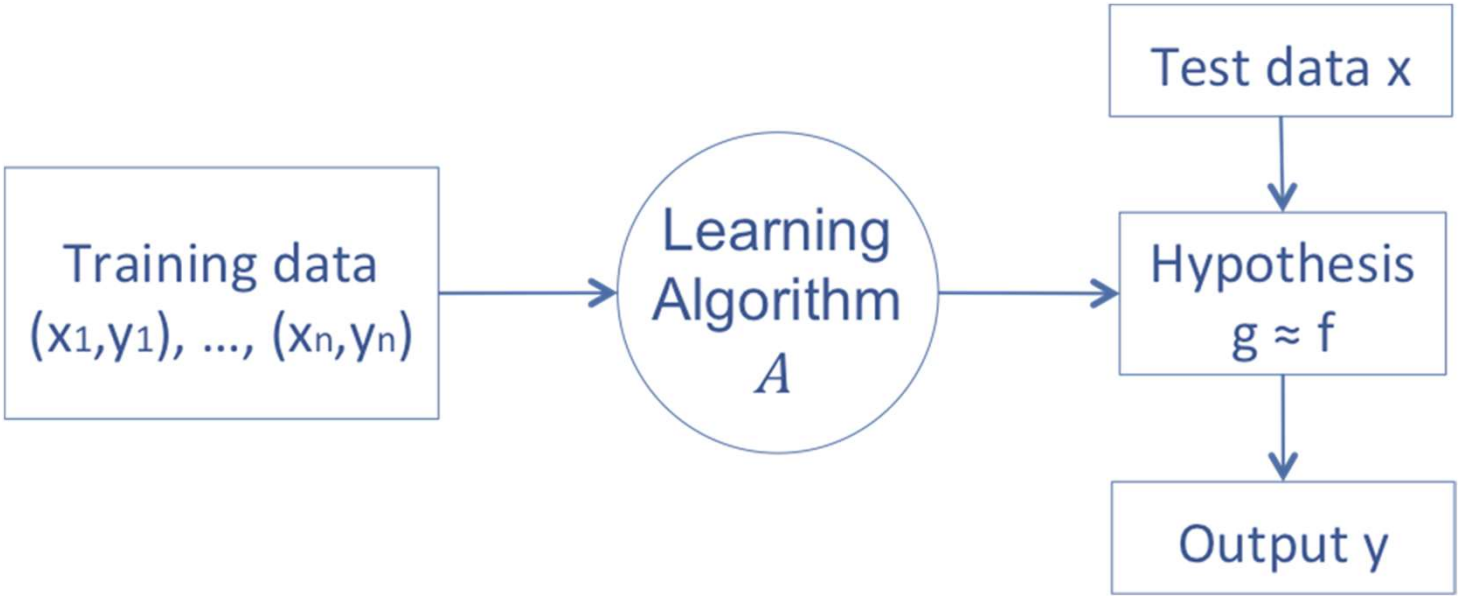
- Limitations of explicit programming
  - Spam filter : many rules
  - Automatic driving : too many rules
- "Field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959)

Machine Learning Disciplines



Learning Algorithms

- Learn from experience E wrt some class of tasks T and performance measure P
  - Task T
  - Performance measure P
  - Experience E



Learning Algorithms & Tasks

- Supervised learning (지도 학습)
  - Regression
  - Classification
- Unsupervised learning (비지도 학습)
  - Clustering
  - Density estimation
  - Visualization
- Reinforcement learning (강화 학습)
  - Credit assignment problem

Supervised Learning의 종류

- Regression (회귀)
  - Predicting final exam score based on time spent
- Binary classification (분류)
  - Pass/non-pass
- Multi-label classification
  - Letter grade (A, B, C, D and F)

x(hour)	y(score)
10	90
9	80
3	50
2	30

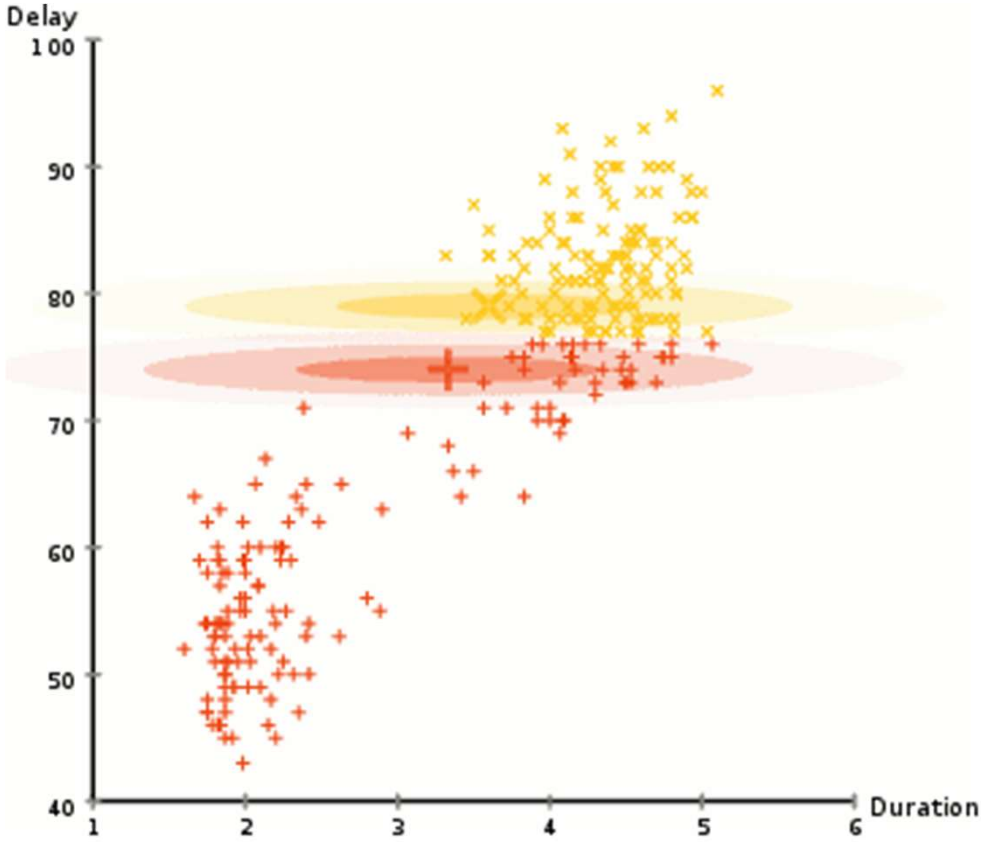
x(hour)	y(P/F)
10	P
9	P
3	F
2	F

x(hour)	y(grade)
10	A
9	B
3	D
2	F

<http://hunkim.github.io/ml/>

Unsupervised Learning

- Clustering algorithm
  - Genomics, Social network analysis, Astronomical data analysis
- EM Clustering



"The 5 clustering Algorithms Data Scientists Need to Know"
(George Seif, https://towardsdatascience.com/>

Classification Algorithms

- Artificial neural networks (Deep Learning)
- Support vector machines
- Logistic regression
- Nonparametric kernel density regression/classification
- Gaussian process regression/classification
- Fisher discriminant analysis (linear discriminant analysis)
- Bayes classification with class conditional models
- Graphical models and inference

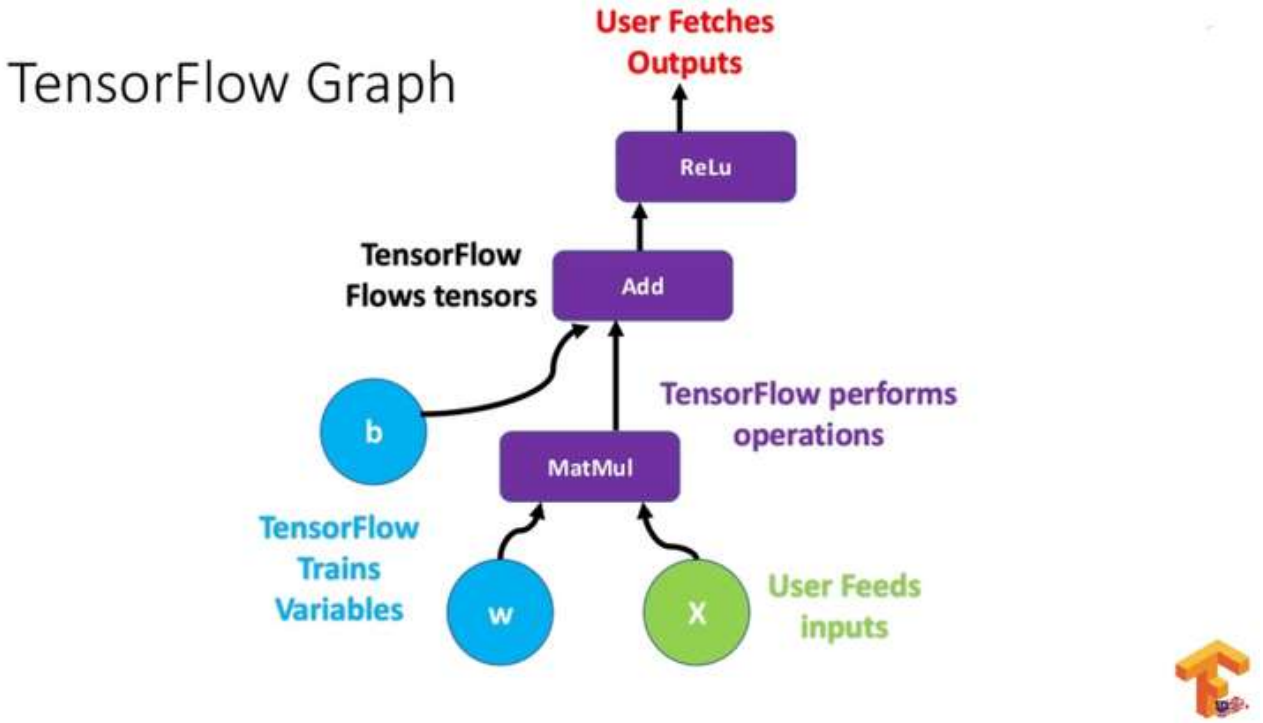
TensorFlow

- TensorFlow
  - An open source SW library for numerical computation using data flow graphs
- Python!

<http://www.tensorflow.org/>

Data Flow Graph?

- Graph: 노드와 엣지로 이루어진 수학적인 구조
  - Nodes : mathematical operations
  - Edges : multidimensional data arrays(tensors) communicated between them



TensorFlow 설치

가상환경 생성

```
>conda create –n ‘가상환경’ python=3.6
>activate ‘가상환경’
>conda install tensorflow==1.15
```

TensorFlow 설치

Version check

```
>>>import tensorflow as tf
>>>tf.__version__

'1.15.0'
```

Hello TensorFlow!

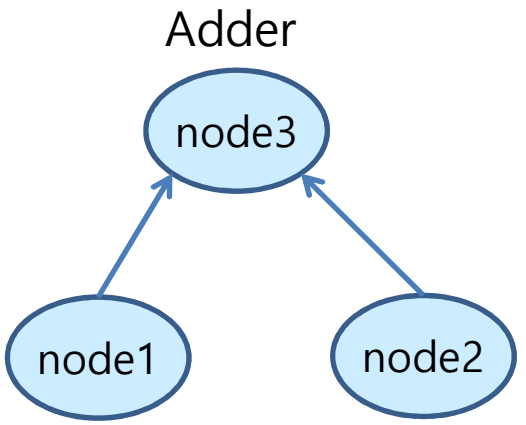
```
import tensorflow as tf
hello = tf.constant("Hello TensorFlow!")
sess  = tf.Session()
print(sess.run(hello))

b'Hello TensorFlow!'
```

Computation Graph

```
node1 = tf.constant(3.0, tf.float32)
node2 = tf.constant(4.)
node3 = tf.add(node1, node2)
```

```
sess = tf.Session()
print(sess.run([node1, node2]))
print(sess.run(node3))
```



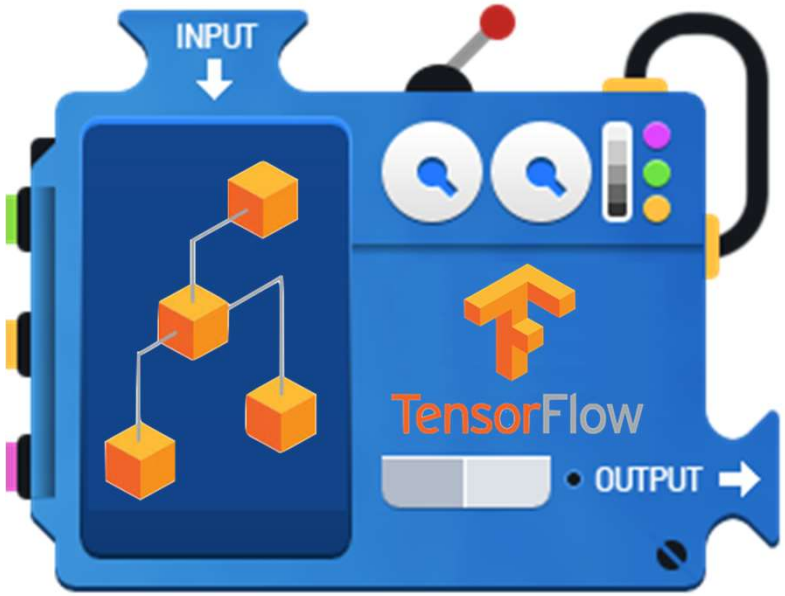
* Placeholder

```
a = tf.placeholder(tf.float32)
b = tf.placeholder(tf.float32)
adder_node = a + b

sess = tf.Session()
print(sess.run(adder_node, feed_dict={a: 3, b: 4.5}))
print(sess.run(adder_node, feed_dict={a: [1, 3], b: [2, 4]}))
```

TensorFlow Mechanics

2 feed data and run graph (operation)
sess.run (op)



1 Build graph using
TensorFlow operations

3 update variables
in the graph
(and return values)

WWW.MATHWAREHOUSE.COM

Tensor Ranks, Shapes

Rank	Math entity	Python example
0	Scalar (magnitude only)	`s = 483`
1	Vector (magnitude and direction)	`v = [1.1, 2.2, 3.3]`
2	Matrix (table of numbers)	`m = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]`
3	3-Tensor (cube of numbers)	`t = [[[2], [4], [6]], [[8], [10], [12]], [[14], [16], [18]]]`
n	n-Tensor (you get the idea)	`....`

Rank	Shape	Dimension number	Example
0	[]	0-D	A 0-D tensor. A scalar.
1	[D0]	1-D	A 1-D tensor with shape [5].
2	[D0, D1]	2-D	A 2-D tensor with shape [3, 4].
3	[D0, D1, D2]	3-D	A 3-D tensor with shape [1, 4, 3].
n	[D0, D1, ... Dn-1]	n-D	A tensor with shape [D0, D1, ... Dn-1].