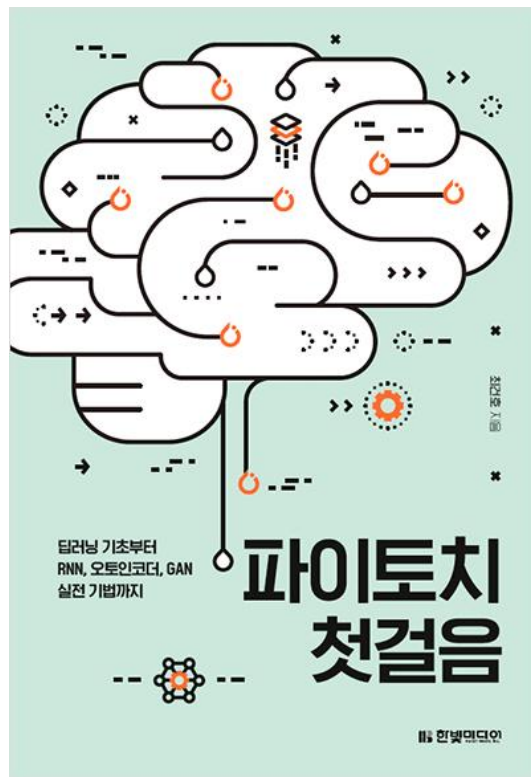


<http://bit.ly/39FINWZ>

# Deep Learning & PyTorch

박종민

## 5주 동안



출처: [한빛출판네트웍](#)

# Deep Learning

- Artificial Intelligence

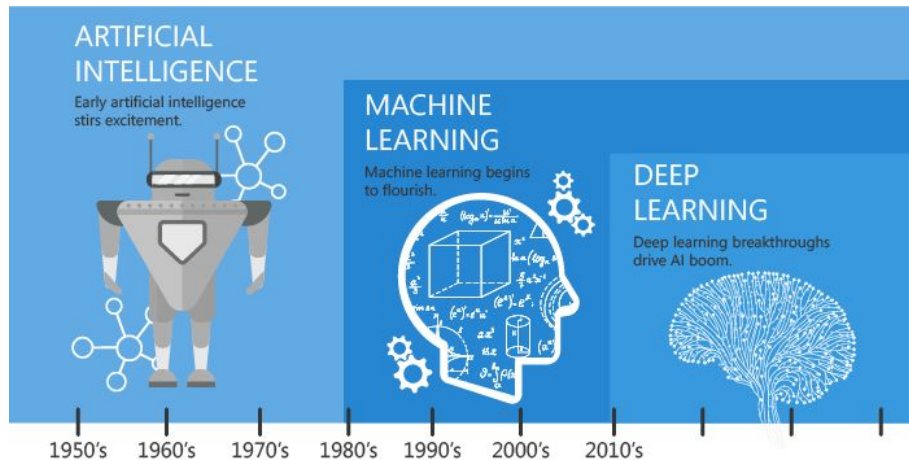
- 기계가 인간의 인지 과정을 모방하는 것

- Machine Learning

- 명시적으로 프로그래밍되지 않은 상태에서 기계에게 학습할 능력을 부여하는 것

- Deep Learning

- ML 중에서 Hidden Layer를 1개 이상 사용하는 Neural Network



출처: [Difference between Artificial Intelligence, Machine learning, and deep learning](#)

# 학습 방법

- Supervised Learning

- Data와 Label의 pair가 있다면 서로의 관계를 학습하는 방법

- Unsupervised Learning

- Data만 있는 상태에서 그것들의 pattern을 찾는 방법

- Reinforcement Learning

- 설정된 Environment에서 Agent의 Reward를 극대화 하는 방법
- 반대로 잘못된 방법으로 Agent가 움직인다면 Penalty 부여
- 최근 Reward를 사용하지 않는 방법도 제시됨

# 발전 원인

- Hardware

- 게임 그래픽 성능을 올리기 위해 병렬 처리가 용이한 GPU 발전
- NVIDIA의 [CUDA](#) 출시 및 ImageNet(2011)에서 CUDA를 사용하기 시작

- Open Dataset과 Benchmark

- [ImageNet](#)
- [COCO](#)
- [SQuAD](#)
- [kaggle](#)

# 무엇을 할 수 있는가

- 사람과 비슷한 수준의 이미지 분류, 음성 인식, 필기 인식
- 자연스러워진 번역기 ([Google Translate](#), [Papago](#))
- TTS(Text To Speech)와 STT(Speech To Text)
- Google, Baidu, Bing, Facebook 광고 타게팅
- [AlphaGo](#)
- 자율 주행 자동차

# 왜 배워야 하는가

- 최근 5년 사이 놀라울 정도의 기술 개발 및 성능 향상
- 의료 진단, 디지털 비서 등의 중요한 문제에 적용하기 시작
- 앞으로 개발 중이거나 적용할 산업계, 생활 문제가 많음
- 단기간 과대 선전으로 주춤할 수 있으나 장기적으로 반드시 **AI** 시대가 올 것

# PyTorch

- 정보
  - [공식 사이트](#), [Forum](#), [GitHub](#) ([Facebook AI Research](#) 팀에서 관리)
  - GPU를 활용하여 인공 신경망 모델을 만들고 학습시키는 **framework**
- 특징
  - 이름에서 유추 가능한 **Python** 베이스
  - **Deep Learning** 프레임워크 중 사용 비중이 급격히 상승
  - 주로 연구, 개발 목적으로 사용 (서비스 목적은 **TensorFlow**)



# PyTorch vs. NumPy vs. TensorFlow

- vs. NumPy

- 딥러닝에 특화되어 계산 코드가 단순 (pre-defined classes and functions)
- GPU 연산 가능

- vs. TensorFlow

- **Graph**를 만드는 동시에 값을 할당하여 더 빠름 (연산, 모델에 따라 다름)
- 개발 속도가 빠름, 서비스가 목적이라면 불편 (개인 선호도 차이)
- 경쟁 구도로 인해 서로 긍정적인 영향을 줌
- 국내 커뮤니티: [TensorFlow KR](#), [PyTorch KR](#)

# Linux 환경

- GPU가 필요한가?
  - 속도 향상을 원한다면 당연히! (CPU로 1주일 소요되는 일이 GPU라면 반나절 안에 가능)
- 왜 굳이 Linux?
  - Windows: 개발되어 있는 프로젝트들 중 Shell script를 사용하는 등 호환성 문제
  - MacOS: 공식적으로 NVIDIA GPU 사용 불가, [더 이상 드라이버 업데이트 제공 안 함](#)
  - 터미널을 통해 CUDA 및 cuDNN을 설치할 때 반드시 운영체제와 버전 확인
  - `lsb_release -a`

# Python 환경

- Anaconda?
  - 프로젝트에 따라 요구되는 Python 버전이 필요
  - 관리 및 설치가 편리
  - 불필요한 라이브러리 설치로 인한 혼동을 줄임
  - 독립 환경에서 필요한 라이브러리만 설치 가능
  - 참고: [Why Anaconda? How to control Anaconda?](#)

# PyTorch 설치

PyTorch Build	Stable (1.3)			Preview (Nightly)			
Your OS	Linux		Mac		Windows		
Package	Conda		Pip		LibTorch	Source	
Language	Python 2.7		Python 3.5		Python 3.6	Python 3.7	C++
CUDA	9.2		10.1			None	
Run this Command:	<code>conda install pytorch torchvision cudatoolkit=10.1 -c pytorch</code>						

출처: <https://pytorch.org/get-started/locally/>

# Docker 활용

- 공식 [Docker Image](#)
- [Dockerfile](#) 기준 스펙
  - Ubuntu 16.04
  - CUDA 10.1
  - cuDNN 7.x
  - Python 3.7

# Colab

- [Python 기초](#)
- [PyTorch와 TensorFlow 비교](#)
- [PyTorch 활용](#)

Q & A