

20기 정규세션

ToBig's 19기 강의자
김은호

Week 1-(2): Frameworks (and Libraries)

Contents

Unit 01	Frameworks and Libraries
Unit 02	Types of frameworks and libraries
Unit 03	Sklearn, TensorFlow, Pytorch
Unit 04	Coding Examples

Unit 01 | Frameworks and Libraries

Framework Definition :

“ A framework is a structure that you can **build software on**. “

- 구조와 규칙을 제공하는 기반
- 개발자는 프레임워크를 사용해 애플리케이션의 구조와 동작 정의
- 모듈, 라이브러리, 도구, 템플릿 등으로 구성

프레임워크 위에서 코드를 짠다

Library Definition :

“ **Collection of codes and modules** for optimizing the current task. “

- 재사용 가능한 코드의 집합
- 필요한 기능의 구현과 작업의 단순화를 돕는 역할
- 개발자가 특정 기능을 수행하기 위해 호출할 수 있는 함수, 클래스 , 모듈로 구성

코드에서 라이브러리를 활용한다

Unit 01 | Frameworks and Libraries

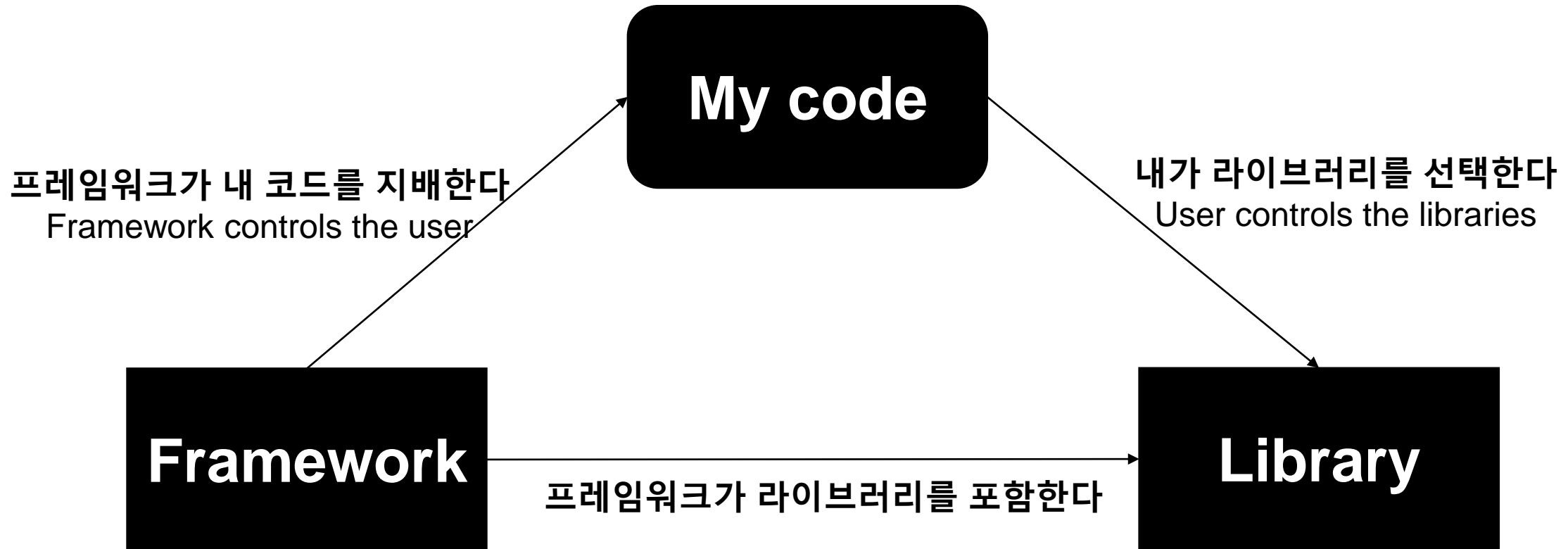
Framework

- **Backbone** of code
- 집으로 치면 집 짓기 전 건설사,
선택이 제한되며 건설사의 규정을 따를 의무
- 개발자는 해당 프레임 워크의
규칙과 구조를 따라야 한다

Library

- **Additional tools** for code
- 집으로 치면 이미 완성된 집에
가구, 장치 등을 추가하는 것
- 개발자는 필요한 기능을 수행하기 위해
라이브러리를 선택하고 함수를 호출해 사용

Unit 01 | Frameworks and Libraries



Contents

Unit 01	Frameworks and Libraries
Unit 02	Types of frameworks and libraries
Unit 03	Sklearn, TensorFlow, Pytorch
Unit 04	Coding Examples

Unit 02 | Types of frameworks and libraries

Most popular **frameworks** in Python

- Django
- Web2Py
- Flask
- Bottle
- CherryPy



django

WEB2PY

Bottle



Unit 02 | Types of frameworks and libraries

Popular **libraries** in Python

- os
- Numpy
- Pandas
- Matplotlib
- Scipy
- sklearn
- Tensorflow
- Pytorch



pandas



NumPy

matplotlib



SciPy



TensorFlow



PyTorch



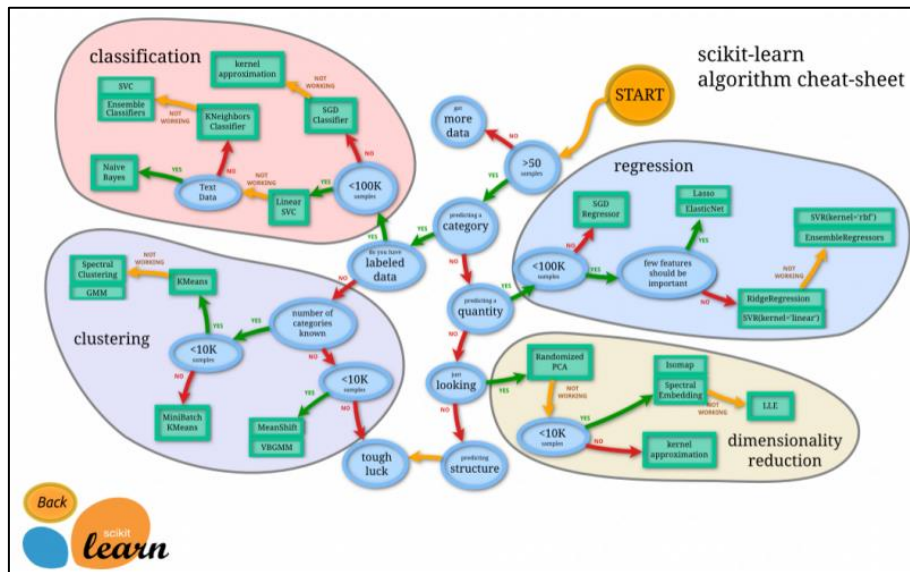
Contents

Unit 01	Frameworks and Libraries
Unit 02	Types of frameworks and libraries
Unit 03	Sklearn, TensorFlow, Pytorch
Unit 04	Coding Examples

Unit 03 | Sklearn, Tensorflow, PyTorch

sklearn

- Scikit-learn(사이킷런) 의 약자 sklearn
- 머신러닝 및 데이터 전처리할 때 자주 사용하는 라이브러리
- 회귀, 분류, 군집화, 차원축소 등 다양하고 검증이 된 기능들을 가지고 있다.
- 머신러닝, 인공지능의 대표적이고 필수적인 라이브러리.



```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y,test_size=0.2, random_state=42)
RF_model = RandomForestRegressor(n_estimators=2000, random_state=42)
RF_model.fit(X_train,Y_train)
```

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y,test_size=0.2, random_state=41)
XGB_model = xgboost.XGBRegressor(n_estimators=2000, random_state=41)
XGB_model.fit(X_train,Y_train)
```

Unit 03 | Sklearn, Tensorflow, PyTorch

Tensorflow

- Tensorflow(텐서플로우)는 파이썬 딥러닝에 특화된 라이브러리.
- 구글이 2011년에 개발을 시작해 2015년에 오픈 소스로 공개한 딥러닝 학습 라이브러리



TensorFlow를 사용 해야 하는 이유

전문가이든 초보자이든, TensorFlow는 ML 모델을 쉽게 빌드 및 배포할 수 있게 해주는 엔드 투 엔드 플랫폼입니다.

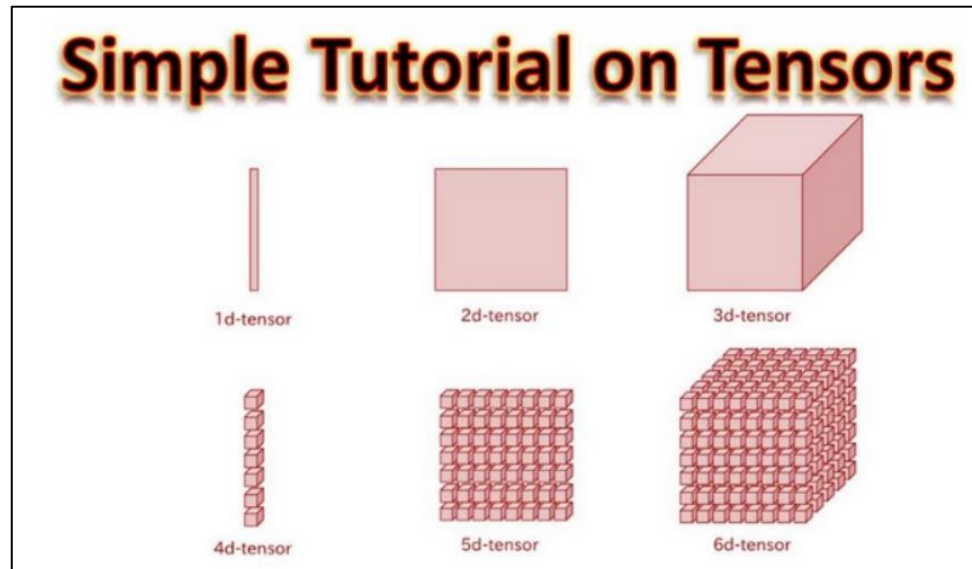
[동영상 보기](#) [우수사례](#) →



Unit 03 | Sklearn, Tensorflow, PyTorch

Tensor?

- Tensor는 수학, 물리 등에서 다양하게 쓰이는 용어.
- 컴퓨터 과학에서는 numpy array (행렬, matrix) 와 많이 유사하지만, tensor는 GPU 에서 병렬학습이 가능하다는 차이가 존재한다.
- 효율적인 딥러닝 학습을 위해서는 tensor 을 이용한 병렬학습이 강제되기 때문에, 딥러닝을 한다면 tensor로의 변환은 필수적이다.



Unit 03 | Sklearn, Tensorflow, PyTorch

PyTorch

- PyTorch도 Tensorflow와 같이 딥러닝 학습에 특화된 라이브러리
- 페이스북 인공지능 연구팀이 개발
- Autoencoder, CNN, RNN, LSTM 등 다양하고 폭넓은 딥러닝 패키지 제공



FROM
RESEARCH TO
PRODUCTION

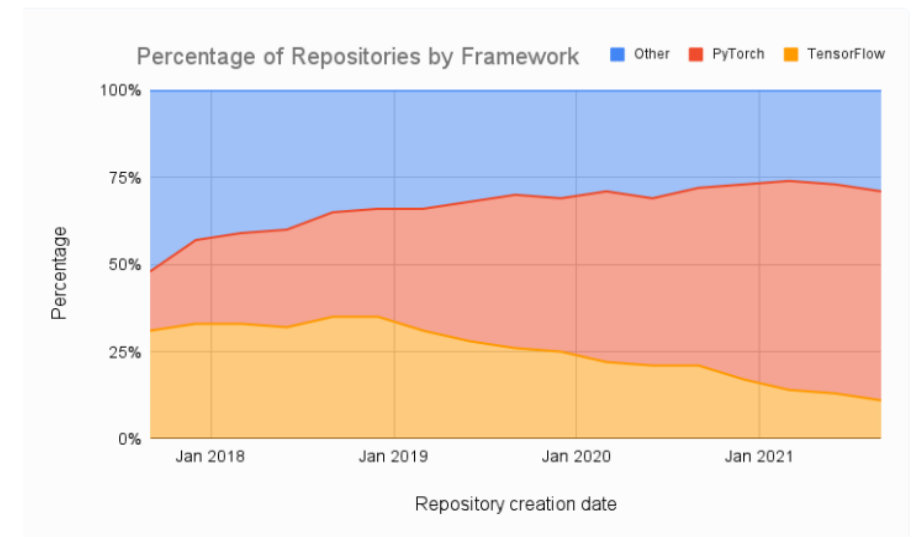
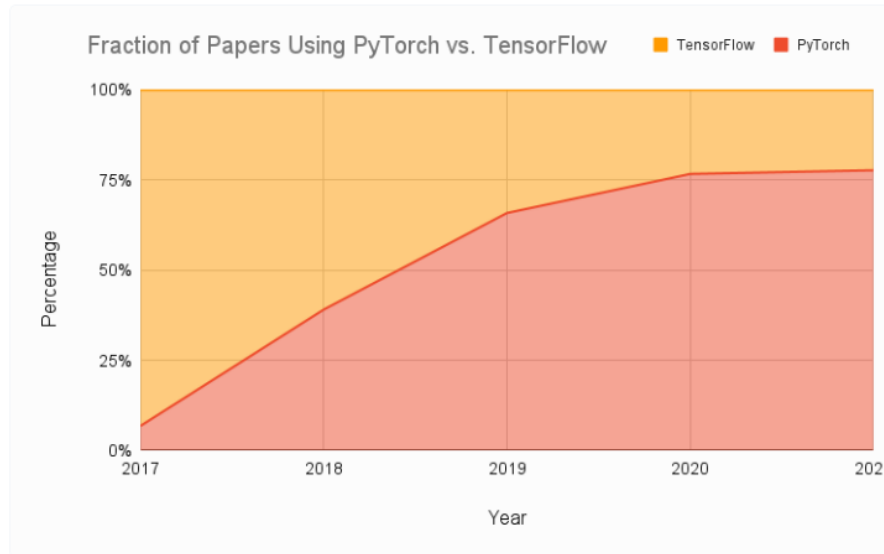
An open source machine learning framework that accelerates the path from research prototyping to production deployment.

Install >

Unit 03 | Sklearn, Tensorflow, PyTorch

Tensorflow
VS
PyTorch

- PyTorch 와 Tensorflow 모두 대표적인 딥러닝 라이브러리
- 대부분의 연구는 Tensorflow 에서 PyTorch의 이용으로 옮겨가는 경향
- 아직 많은 산업들은 Tensorflow 사용 (연구실보다 적응이 느리기 때문)

출처 : <https://paperswithcode.com/trends>출처 : <https://thegradient.pub/state-of-ml-frameworks-2019-pytorch-dominates-research-tensorflow-dominates-industry/>

Contents

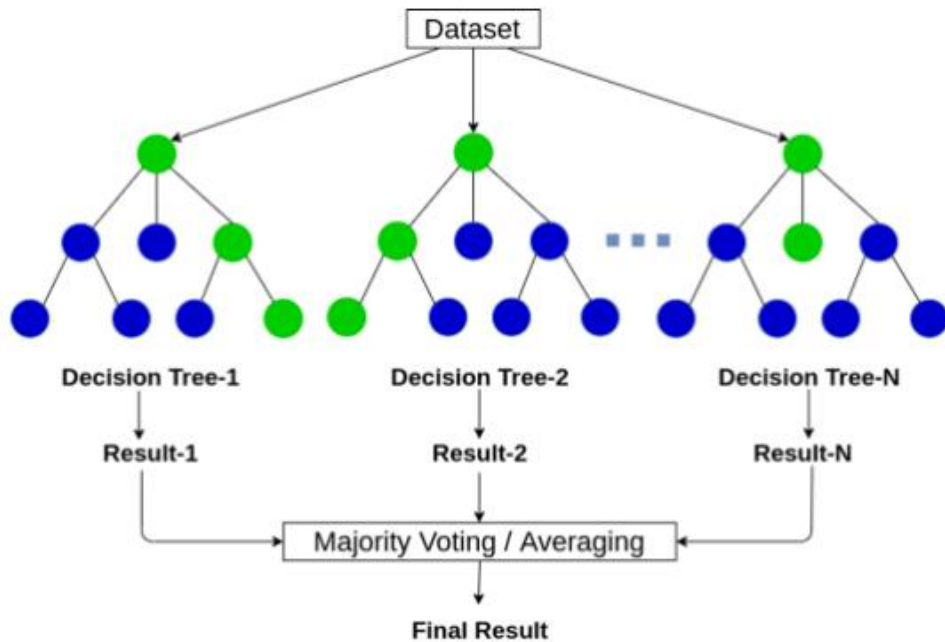
Unit 01	Frameworks and Libraries
Unit 02	Types of frameworks and libraries
Unit 03	Sklearn, TensorFlow, Pytorch
Unit 04	Coding Examples

Unit 04 | Coding Examples

sklearn

Random Forest Algorithm

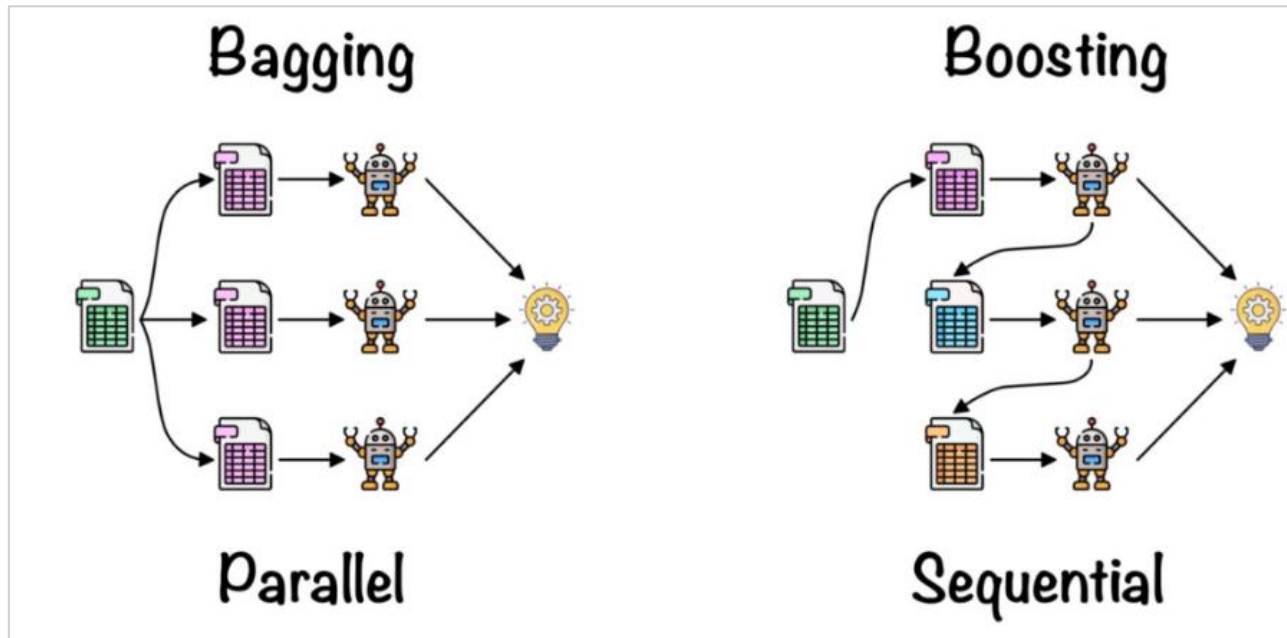
- Random Forest 은 이전의 ML 알고리즘의 주를 이루었던 Least Square Regression 방식에서 벗어난 Tree 를 이용한 앙상블(Ensemble) 기법.
- 머신러닝 알고리즘 중에서 인기 많은 모델이며, 과적합(Bagging 의 특징) 및 이상치(Tree 의 특징)에 강하다는 특징이 있다.
- 트리를 대량으로 생산하기 때문에 시간이 비교적 오래 걸린다는 단점이 존재한다.



Unit 04 | Coding Examples

sklearn

Ensemble Method. Bagging and Boosting



Bagging(bootstrap aggregating)

→ 다수결

→ Random Forest

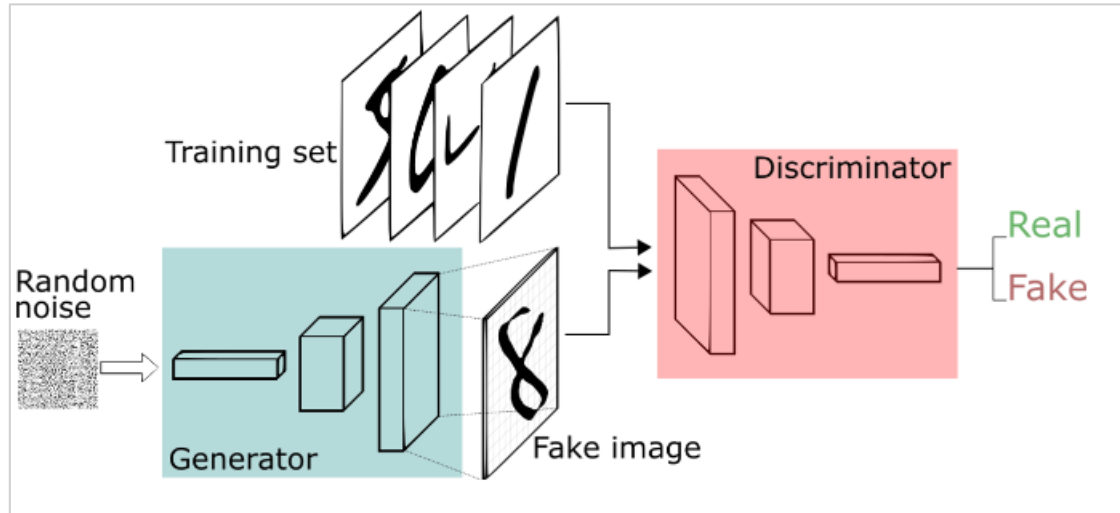
Boosting

→ 다수 적응

→ AdaBoost, XGBoost

Unit 04 | Coding Examples

PyTorch



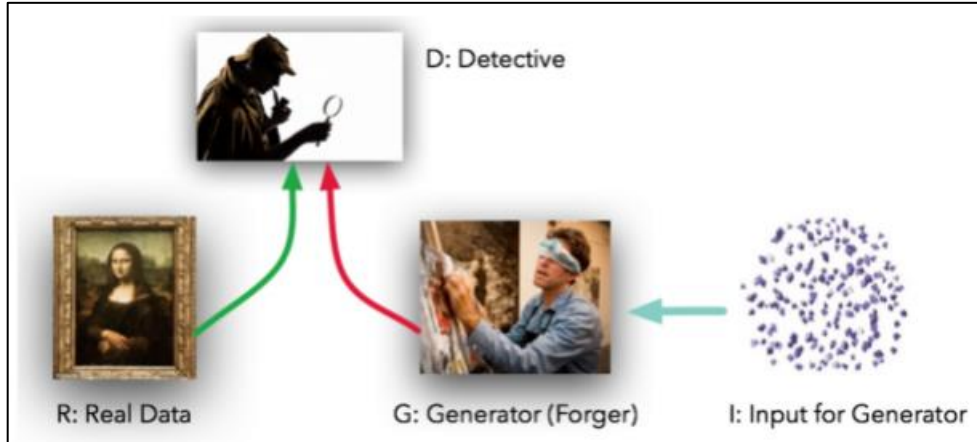
Deep Learning Model GAN (Generative Adversarial Network)

- 적대적 생성 네트워크로, 딥러닝 비전에서 시작해 Sound Generation, Text Generation 등으로 넓게 응용되고 있는 모델
- Generator 와 Discriminator 의 min-max loss 으로 학습을 진행

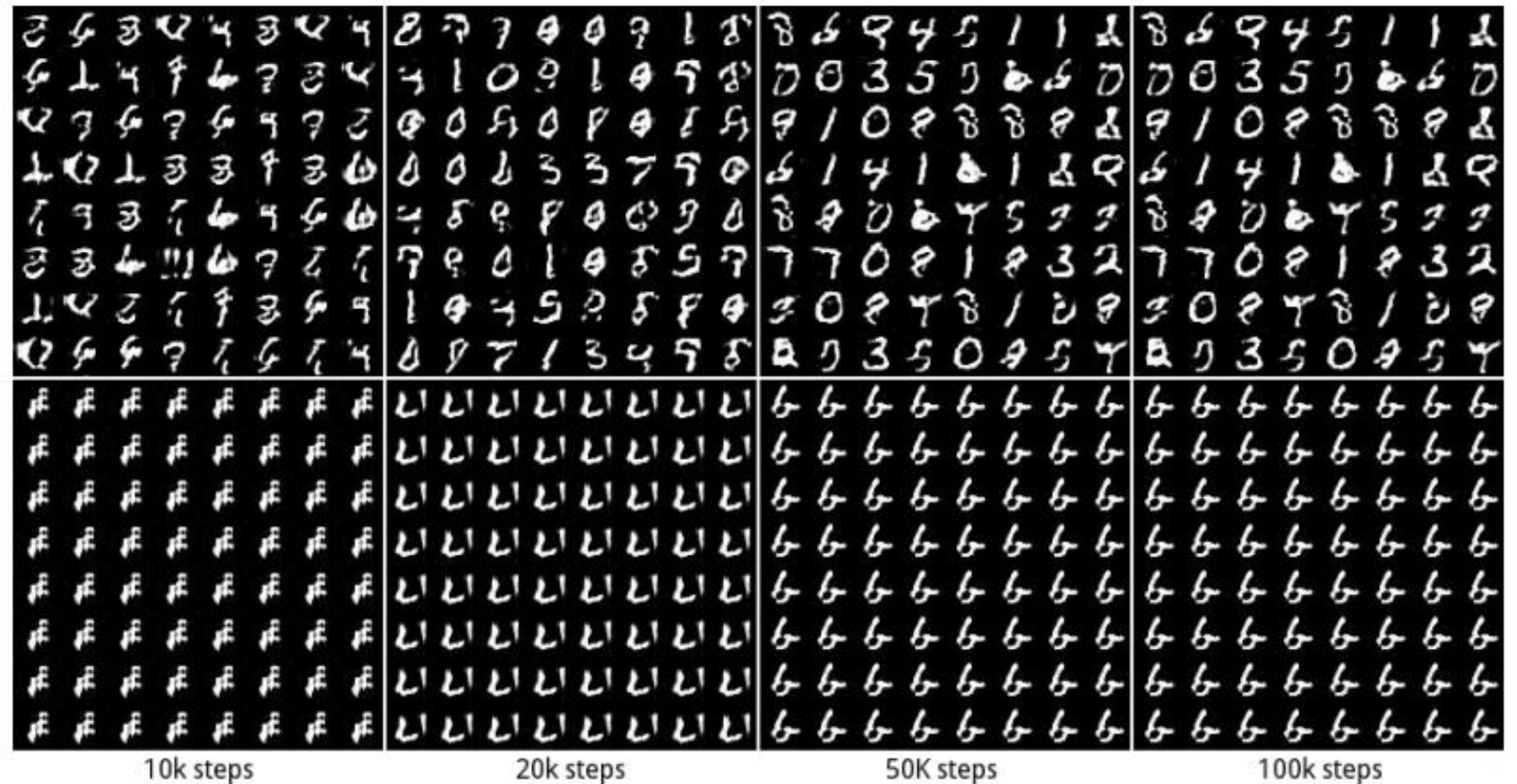
Unit 04 | Coding Examples

PyTorch

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))].$$



- 딥러닝 모델에서 **Loss Function**이 매우 중요한 역할
- GAN 에서
Generator (생성자)은 loss 을 최대화 하려고 하고,
Discriminator(판별자)은 loss 을 최소화 하려고 한다.
- G와 D 가 상호보완적으로 학습을 해야 한다.



Homework

‘Week1_Framework.ipynb’ 채우기.

- 1. Library 와 Framework 의 차이 간단하게 서술하기. (100자 내외)**
- 1. 딥러닝과 머신러닝의 관계 및 특징, 차이 간단하게 서술하기. (200자 내외)**
- 1. ipynb 코드 보고 이해하고 주석 달기.**

참고

<https://volcano-screen-ada.notion.site/5de439afd31f41509bf2a0fcd3322544>

Unit 04 | Coding Examples

sklearn

<https://colab.research.google.com/drive/1W92rbalOlhqkrjoxVwKMb5XRkUHgpKiV?usp=sharing>

PyTorch

<https://colab.research.google.com/drive/1IYVIUIGsUzWWoBoLPhzC9VbnWGA70tcG?usp=sharing>

감사합니다