

# 파이토치 딥러닝 프레임워크

# 딥러닝 구현 라이브러리

## 파이토치(Pytorch)

파이토치는 2017년 초에 공개된 딥러닝 프레임워크로 개발자들과 연구자들이 쉽게 GPU를 활용하여 인공 신경망 모델을 만들고 학습시킬수 있게 도와준다.

파이토치의 전신이라고 할수 있는 토치(Torch)는 루아 프로그래밍 언어로 되어 있지만, 파이토치는 파이썬으로 작성되어 파이썬의 언어특징을 많이 가지고 있다.

파이토치는 페이스북 인공지능 연구팀(Research)멤버들이 주로 관리하며, 독자적으로 운영되는 파이토치 포럼은 사람들이 질문을 올리면 프레임워크 개발자를 비롯한 많은 사람들이 답이 해주는 등 활발히 교류가 일어나고 있다.

## 파이토치 공식사이트 <https://pytorch.org>



**Andrej Karpathy** ✓  
@karpathy

I've been using PyTorch a few months now and I've never felt better. I have more energy. My skin is clearer. My eye sight has improved.

오전 3:56 · 2017년 5월 27일 · [Twitter Web Client](#)

492 리트윗 1.7천 마음에 들어요 수



**Sean Robertson** @sproberson · 2017년 5월 27일  
[@karpathy](#) 님에게 보내는 답글

Talk to your doctor to find out if PyTorch is right for you.

1

16

151



**Soumith Chintala** ✓ @soumithchintala · 2017년 5월 27일  
this goes hand in hand with the parody account [@debuggermassa](#)

1

22



## FROM RESEARCH TO PRODUCTION

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

Get Started >

OpenAI 에서 근무하다가 현재는 테슬라에서 일하고 있는 안드레이 카사피(Andrej Karpathy)는 파이토치를 쓰게 된 후 얼마나 몸과 마음이 건강해졌는지 트윗을 쓰기도 했다.

OpenAI는 2015년 말 일론 머스크가 세운 비영리 인공지능 연구소로 세계적으로 인정받은 연구자들이 있는 회사이다. 따라서 매우 다양한 모델과 프레임워크를 써봤을 텐데, 그 중 파이토치가 가장 편리하다고 말할 정도면 파이토치가 확실히 장점이 있다는 것을 알 수 있다.

# 텐서플로우와 파이토치의

## 차이점

텐서플로우와 파이토치의 차이점을 살펴보면 다음과 같다.

파이토치와 텐서플로우 둘다 연산에 GPU를 이용하는 프레임워크이다. 하지만 텐서플로우는 연산 그래프를 먼저 만들고 실제 연산할 때 값을 전달하여 결과를 얻는 'Define and Run' 방식이고, 파이토치는 그래프를 만듦과 동시에 값이 할당되는 'Define by Run' 방식이다. 텐서플로우의 '그래프를 먼저 정의하고 세션에서 실제로 값을 집어넣어 결과를 도출' 하는 패러다임은 사람에 따라 직관적으로 받아들이기 어려울 수 있고, 그래프를 정의하는 부분과 이를 돌리는 부분이 분리되기 때문에 전체적으로 코드 길이가 길어지게 된다.

반면 파이토치는 연산 그래프를 정의하는 것과 동시에 값도 초기화되어 연산이 이루어지는 'Define by Run' 이므로 연산 그래프와 연산을 분리해서 생각할 필요가 없다.

또한 연산 속도면에서도 차이가 있다.

연산그래프를 고정해놓고 값만 전달하는 텐서플로우가 더 빠른 환경일 수도 있지만 텐서플로우로 깃허브에 올라온 이슈에 따르면 실험에 많이 사용되는 모델로 벤치마킹한 결과 파이토치가 텐서플로우보다 2.5배 빠른 결과가 나왔다고 한다.

텐서플로우는 자체적으로 운영하는 포럼이 없고 구글 그룹도 편리하게 정리되어 있지 않다.

반면 파이토치는 자체 운영포럼이 있어서 질문을 올리면 파이토치 개발자들이 직접 답변을 달아주기도 한다.

# 텐서플로우와 파이토치의 차이점

텐서플로우와 파이토치의 차이점을 살펴보면 다음과 같다.

구분	Tensorflow	Pytorch
패러다임	Define and Run	Define by Run
그래프 형태	Static graph	Dynamic graph
현재 사용자	많음	적음
자체 운영 포럼	없음	있음
한국 사용자 모임	Tensorflow Korea(TF-KR)	Pytorch Korea(Pytorch-KR)

# Pytorch가 주목받는 이유

그렇다면 ! 어떤 이유로 최근 파이토치가 주목을 받는 것일까? 파이토치의 장점을 살펴보자.

- ▶ 설치가 간편하다
- ▶ 이해와 디버깅이 쉬운 직관적이고 간결한 코드로 구성되었다
- ▶ Define-by-Run 방식을 기반으로 한 실시간 결과값을 시각화한다
- ▶ 파이썬 라이브러리(Numpy, Scipy, Cython)와 높은 호환성을 가진다
- ▶ Winograd Convolution Alogithm 기본 적용을 통한 빠른 모델 훈련이 가능하다
- ▶ 모델 그래프를 만들 때 고정상태가 아니기 때문에 언제든지 데이터에 따라 조절이 가능하다(유연성 지님)

- ▶ Numpy 스러운 Tensor 연산이 GPU로도 가능하다
- ▶ Autograd System을 이용해 쉽게 DDN을 짤 수 있다
- ▶ 학습 및 추론 속도가 빠르고 다루기 쉽다

## Pytorch 내에 있는 패키지

패키지	기술
<b>torch</b>	강력한 GPU 지원 기능을 갖춘 Numpy와 같은 Tensor 라이브러리
<b>torch.autograd</b>	Torch에서 모든 차별화된 Tensor 작업을 지원하는 테이프 기반 자동 차별화 라이브러리
<b>torch.nn</b>	최고의 유연성을 위해 설계된 자동 그래프와 깊이 통합된 신경 네트워크 라이브러리
<b>torch.optim</b>	SGD, RMSProp, LBFGS, Adam 등과 같은 표준 최적화 방법으로 torch.nn과 함께 사용되는 최적화 패키지
<b>torch.multiprocessing</b>	파이썬 멀티 프로세싱을 지원하지만, 프로세스 전반에 걸쳐 Torch Tensors의 마법같은 메모리 공유 기능을 제공. 데이터 로딩 및 호그 워트 훈련에 유용.
<b>torch.utils</b>	편의를 위해 DataLoader, Trainer 및 기타 유틸리티 기능
<b>torch.legacy(.nn/optim)</b>	이전 버전과의 호환성을 위해 Torch에서 이식된 레거시 코드