

# 이미지 학습을 위한 딥러닝 프레임워크 비교분석\*

김종민\*, 이동휘\*\*

## 요약

딥러닝 프레임워크는 현재에도 계속해서 발전되어 가고 있으며, 다양한 프레임워크들이 존재한다. 딥러닝의 대표적인 프레임워크는 TensorFlow, PyTorch, Keras 등이 있다. 딥러닝 프레임워크는 이미지 학습을 통해 이미지 분류에서의 최적화 모델을 이용한다. 본 논문에서는 딥러닝 이미지 인식 분야에서 가장 많이 사용하고 있는 TensorFlow와 PyTorch 프레임워크를 활용하여 이미지 학습을 진행하였으며, 이 과정에서 도출한 결과를 비교 분석하여 최적화된 프레임워크를 알 수 있었다.

## A Comparative Analysis of Deep Learning Frameworks for Image Learning

Jong-min Kim\*, Dong-hwi Lee\*\*

### ABSTRACT

Deep learning frameworks are still evolving, and there are various frameworks. Typical deep learning frameworks include TensorFlow, PyTorch, and Keras. The Deep Learning framework utilizes optimization models in image classification through image learning. In this paper, we use the TensorFlow and PyTorch frameworks, which are most widely used in the deep learning image recognition field, to proceed with image learning, and compare and analyze the results derived in this process to know the optimized framework. was made.

**Key-words:** Deep Learning Frameworks, Image Learning, TensorFlow, PyTorch

접수일(2022년 09월 05일), 수정일(2022년 09월 21일),  
게재확정일(2022년 10월 31일)

\* 동신대학교 정보보안학과

\*\* 동신대학교 정보보안학과(교신저자)

★본 과제(결과물)는 2022년도 교육부의 재원으로 한국연구재단의 지원을 받아 수행된 지자체-대학 협력기반 지역혁신 사업의 결과입니다. (NRF-1345341782)

## 1. 서 론

딥러닝 프레임워크는 놀라울 정도로 빠른 속도로 발전해 오고 있으며, 지금 현재에도 계속해서 발전을 해가고 있다. 딥러닝 프레임워크는 다양한 분야에서 활용하고 있으며, 대학에서 개발되고 사용되어 왔던 딥러닝 프레임워크로서는 버클리 대 학의 '카페(Caffe)', 몬트리올 대학의 '티아노(Theano)'이다. 그리고 구글이 '텐서플로(TensorFlow)'를 오픈소스로 공개함에 따라 저변을 확대해 오고 있다[1].

오픈소스의 경우 실험에 대해 결과를 신속하게 낼 수 있다는 장점이 있으며, 예리 및 다양한 프레임워크에 적용이 가능하다는 장점도 가지고 있다. 이러한 오픈소스 기반의 딥러닝 프레임워크는 수백가지 이상이 존재하며, 그중 TensorFlow와 PyTorch가 주로 사용되어 지고 있다.

따라서, 본 연구에서는 TensorFlow와 PyTorch를 사용하여 이미지 분석을 하기 위해 학습을 진행해 두 프레임워크를 비교분석하여 이미지 학습에 최적화 된 프레임워크를 도출하고자 한다.

이러한 결과를 도출하기 위해서, 2장에서는 딥러닝 프레임워크, 3장에서는 제안하는 방법, 4장에서는 이미지 학습 프레임워크 비교분석 5장 결론으로 구성하였다.

## 2. 딥러닝 프레임워크

딥러닝 프레임워크란 딥러닝 모형을 쉽고 빠르게 개발할 수 있도록 도움을 주는 소프트웨어를 말하며[1], 딥러닝소프트웨어, 딥러닝 라이브러리, 머신러닝 라이브러리, 딥러닝 툴, 오픈소스 AI 툴 등 다양한 이름으로도 불린다. 딥러닝 프레임워크는 심층 신경망(DNN, Deep Neural Network) 모델 구현을 단순화하여, 다양한 심층 아키텍처를 쉽게 만들고 테스트할 수 있게 한다. 이 프레임워크의 대표적인 기능으로는 크고 깊은 DNN 컴퓨팅 모델을 만드는 것(모델 구축), 컴퓨팅 모델 안에서 기울기(gradient)를 계산하는 것(자동미분), cuDNN, cuBLAS 등을 포장하여 GPU에서 효율적으로 실행이 가능하게 하는 것(GPU 활용) 등이 있다. 오픈소스 소프트웨어로 공유되는 딥러닝 프레임워크의 대표적인 예로는, 엠엑스넷(MXNet, Amazon), 텐서플로(Tensor

Flow, Google), 씨엔터케이(CNTK, Microsoft), 패들패들(PaddlePaddle, Baidu), 카페2(Caffe2, Facebook), 파이토치(PyTorch, Facebook)등이 있다[2].

### 2.1 Tensorflow

텐서플로우는 구글이 개발하고 배포한 오픈소스 라이브러리이며, 자료의 흐름그래프를 이용하여 수치계산을 수행할 수 있다. 2015년 구글은 머신러닝을 이용하고 싶은 개발자나 연구자들에게 누구나 텐서플로우를 사용할 수 있게 텐서플로우를 포함한 튜토리얼과 예제 파일을 공개하였다[3].

텐서플로우는 대규모 환경에서 작동할 수 있는 머신 러닝을 위한 시스템이며, 데이터 플로우 그래프 기반으로 모델을 구축하며 그래프의 각 노드인 개별연산 즉, OpKernel을 상황에 맞게 서로 다른 디바이스에 배치하여 실행시킨다. 텐서플로우에 연결될 수 있는 디바이스는 멀티코어 CPU, GPU, 그리고 TPU 등이 있으며 만약 사용자가 어플리케이션 코드에서 디바이스를 명시적으로 지정하지 않는 경우에는 각 OpKernel을 사용자 머신에 연결되어 있는 디바이스 중 최적의 디바이스를 찾아 연결시킨다. 따라서 텐서플로우의 기본구조는 사용자가 입력한 원시데이터가 실행 환경에 따라 다양한 디바이스에서 처리되는 구조이다 [4].

### 2.2 PyTorch[5]

Feed Forward Aural Network는 최초의 인공 신경망이며, 단순한 구조로 입력층(input layer)에 데이터를 입력하고, 1개 이상의 은닉층(hidden layer)을 거쳐 출력층(out layer)를 통해 결과를 출력하는 구조이다. 이러한 구조에서는 많은 loop와 재귀함수로 구성되어 있는 복잡한 계산에서는 제한적일 수 밖에 없으며, 이 문제점을 해결하고자 PyTorch가 나오게 되었다.

PyTorch는 Torch를 기반으로 하며, 자연어 처리와 같은 어플리케이션을 위해 사용된다. GPU 사용이 가능하기 때문에 속도가 상당히 빠르며, 아직까지 TensorFlow의 사용자가 많지만, 비직관적인 구조와 난이도 때문에 PyTorch의 사용자가 늘어나고 있는 추세이다. 이는 Facebook의 인공지능 연구팀이 개발했으며, 간결하고 구현이 빠리되며, 텐서플로우보다 사용자가 익히기 훨

편 쉽다는 특징이 있다. 또한 코드를 직접 다룬다 사람들에게 설명해 주기도 효과적이다. 텐서플로와 PyTorch의 가장 큰 차이점은 딥러닝을 구현하는 패러다임이 다르다는 것이다. 텐서플로우는 Define-and-Run 프레임워크인 반면에, Pytorch는 Define-by-Run이다.

Define and Run는 코드를 직접 돌리는 환경인 세션을 만들고, placeholder를 선언하고 이것으로 계산 그래프를 만들고(Define), 코드를 실행하는 시점에 데이터를 넣어 실행하는(Run) 방식. 이는 계산 그래프를 명확히 보여주면서 실행시점에 데이터만 바뀌어도 되는 유연함을 장점으로 갖지만, 그 자체로 비직관적이다. 그래서 딥러닝 프레임워크 중 난이도가 가장 높은 편이다.

### 3. 제안하는 방법

#### 3.1 분석 환경

<표 1>은 이미지 학습을 위한 딥러닝 프레임워크를 비교 분석하기 위한 환경이다.

<표 1> 분석 환경

구분	version
OS	Windows 10
CPU	Intel i5-10400 2.90GHz
RAM	8GB
TensorFlow	2.7.0
PyTorch	1.10.0+cu111
학습모델	VGG16

#### 3.2 데이터 수집

이미지 학습을 위해 사용한 DATA는 bing\_image\_downloader의 파일을 크롤링하여 수집하였으며, (그림 1)는 이미지를 크롤링 하는 과정이다. 수집된 데이터는 사자, 호랑이, 고양이를 크롤링 하여 수집하였다.

```

[ ] query = "lion"
  downloader.download(query, limit=40, output_dir='/', adult_filter_off=True, force_replace=False, timeout=60)
  dataset.call(query, 30)

[ ] query = "cat"
  downloader.download(query, limit=40, output_dir='/', adult_filter_off=True, force_replace=False, timeout=60)
  dataset.call(query, 30)

[ ] query = "tiger"
  downloader.download(query, limit=40, output_dir='/', adult_filter_off=True, force_replace=False, timeout=60)
  dataset.call(query, 30)
    
```

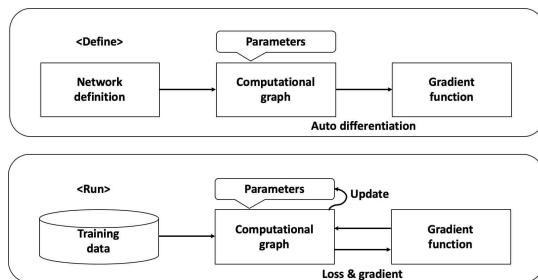
(그림 1) 이미지 데이터 크롤링 예제

### 3.3 이미지 학습

#### 3.3.1 TensorFlow 이미지 학습

TensorFlow를 사용하여 이미지를 학습하기 위해 사용한 모델은 VGG16 이며, 세부설정(include\_top=False)을 통해 VGG16 모델을 특정 추출로 사용 정의하였다. 데이터의 경우 3개로 특정하였으며(사자, 호랑이, 고양이) 특정 개수에 맞게 출력 노드도 맞게 설정해 주었다. 이후 학습에 필요한 라이브러리(Flatten, Dense, Dropout, ImageDataGenerator)를 불러와 이미지 데이터의 최댓값 1/255로 나누어 읽을 때, 자동으로 정규화 되게 하고 flow\_from\_directory를 이용하여 배치 사이즈만큼 categorical모드로 다양한 효과를 주며 읽어 주게 하였다.

(그림 2)는 Define-and-Run 프레임워크로서 TensorFlow 프레임워크에서 사용된다. 이 기반으로 설계된 프레임워크의 경우 정의와 실행의 2단계로 진행이 되며, 훈련전에 그래프를 먼저 정의하기 때문에 복잡한 신경망에서는 제한적이다.



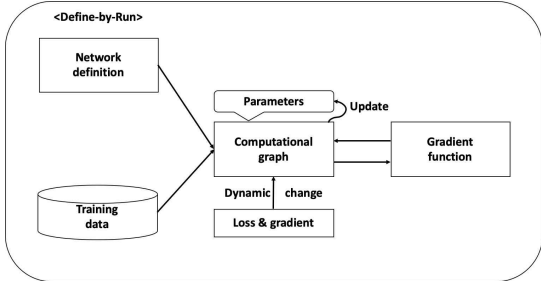
(그림 2) Define-and-Run(TensorFlow)[6]

#### 3.3.2 PyTorch 이미지 학습

PyTorch를 사용하여 이미지 학습을 하기 위해 computer vision용 라이브러리(Torchvision)를 불러온 후 모델에 최적화 하기 위해서 device를 초기화 진행한다. 다음으로 전이 학습을 위해서 VGG16 모델을 사용하여 마지막 레이어를 별도의 리뉴얼 레이어로 바꿔 학습하게 하였으며, 뉴런 수는 파이널 레이어 출력 클래스에 맞춰 설정(3개)해 주었다.

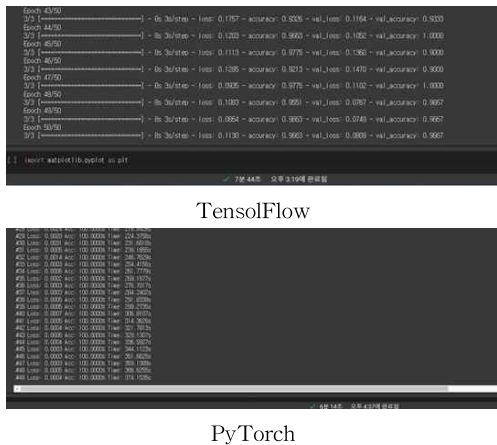
(그림 3)은 Define-by-Run 프레임워크로서 PyTorch 프레임워크에서 사용된다. 이 기반으로 설계된 프레임워크의 경우 정의와 실행의 2단계를 단일 단계로 결합하여 실행하기 때문에, 실행과 동시에 그래프를 생성

하는 방식으로 TensorFlow 방식보다 복잡한 신경망에서 유연하게 사용되어진다.



(그림 3) Define-by-Run(PyTorch)[6]

#### 4. 이미지 학습 프레임워크 비교분석



(그림 4) 프레임워크별 학습 시간

<표 2> 프레임워크 비교분석표

프레임워크	학습속도
TensorFlow	7분 44초
PyTorch	6분 14초

(그림 4)와 <표 2>는 TensorFlow와 PyTorch 프레임워크를 사용하여 똑같은 이미지를 학습한 결과이다.

TensorFlow는 7분 14분이 나왔으며, PyTorch의 경우 TensorFlow 보다 1분 30초 정도 빨리 학습을 완료한 것으로 나타났다.

#### 5. 결론

현재 딥러닝 프레임워크는 빠른 발전을 토대로 다양한 분야에서 적용되어 지고 있으며, 이러한 적용분야가 늘어남에 따라 빠르고 신속하게 분석할 수 있는 방법에 대해 계속해서 연구되어 지고 있다.

따라서 본 논문에서는 딥러닝 프레임워크 중 가장 많이 사용되고 있는 TensorFlow와 PyTorch를 이용하여 이미지 학습에서 어떠한 프레임워크가 빠르게 학습이 되는지를 비교 분석하였으며, 그 결과 PyTorch가 TensorFlow 보다 1분 30초 정도 빨리 학습한 것을 알 수 있었다.

TensorFlow와 PyTorch 프레임워크의 경우 가장 큰 차이점을 보여주는 것은 딥러닝을 구현할 때의 패러다임이다. TensorFlow는 Define-and-Run, PyTorch는 Define-by-Run 방식을 사용한다. 이 두 방식의 가장 큰 차이점은 정의와 실행에 있어 훈련 전 그래프를 먼저 정의 하느냐와 같이 동시에 정의하느냐의 차이가 있었다. Define-and-Run의 방식처럼 그래프를 먼저 정의할 경우 복잡한 신경망에 제한적이다. 하지만 PyTorch에서 사용되어진 Define-by-Run 방식은 이런 복잡한 신경망에 유연하게 대처할 수 있으며, 분석결과에서도 확인할 수 있듯이 각 프레임워크 패러다임의 차이에 따라 분석속도에 유의미한 결과가 나타났다.

본 논문의 결과를 토대로 딥러닝 개발에 대해 적합한 프레임워크를 적용할 수 있는 선행연구가 될 수 있기를 기대한다.

#### 참고문헌

[1] Yeo-Jin Chung , Sung-Mahn Ahn, Jiheon Yang, Jaejoon Lee, "Comparison of Deep Learning Frameworks:About Theano, TensorFlow, and Cognitive Toolkit", Journal of Intelligence Information Systems, Vol.23, No.2, pp. 1-17, 2017.

[2] Eunjoo Choi, Junyeong Lee, In-Goo Han, "Deriving adoption strategies of deep learning open source framework through case studies", Journal of

Intelligence Information Systems, Vol.26, No.4, pp. 27-65, 2020.

[3] Sungho Jung, Daeop Lee, Kyoungsang Lee, "Prediction of River Water Level Using Deep-Learning Open Library", Journal of The Korean Society of Hazard Mitigation, Vol.18, No.1, pp. 1-11, 2018.

[4] Jinhwa Kim, Seehwan Yoo, "Secure TensorFlow: Securing Machine Learning with H/W and S/W", Proceedings of the Korean Information Science Society Conference, pp. 1,967-1,969, 2019.

[5] <https://ko.wikipedia.org/wiki/PyTorch>.

————— [ 저자 소개 ] —————



김 종 민 (Jong-Min Kim)  
 2015년 산업보안학박사  
 현 재 동신대학교 정보보안학과  
 교수  
 email : dyuo1004@dsu.ac.kr



이 동 휘 (DongHwi Lee)  
 2007년 경기대학교 정보보호박사  
 2011년~2012년 University of Colorado  
 Denver, Dept. of  
 Computer Science and  
 Engineering  
 현 재 동신대학교 정보보안학과 교수  
 email : dhclub@dsu.ac.kr