

Machine Learning

Deep Learning Frameworks

Mostafa S. Ibrahim

Teaching, Training and Coaching for more than a decade!

Artificial Intelligence & Computer Vision Researcher

PhD from Simon Fraser University - Canada

Bachelor / MSc from Cairo University - Egypt

Ex-(Software Engineer / ICPC World Finalist)



© 2023 All rights reserved.

Please do not reproduce or redistribute this work without permission from the author

Deep learning frameworks

- Deep learning frameworks provide essential tools for designing, training, and deploying deep neural networks
- There are many tools and I highly recommend PyTorch
 - For learning
 - Also wide use cases in the production

Caffe Framework

- Developed by: **Berkeley** Vision and Learning Center
- Language: **C++**
- Very fast and efficient, suited for production deployments.
- Good for Image classification tasks
- **Less user-friendly API**

TensorFlow (TF)

- Developed by: Google Brain
- Language: Python, C++, JavaScript
- Highlights:
 - Extensive ecosystem (mobile, browser-based apps, and deployment)
 - Excellent scalability across multiple **GPUs and TPUs**.
 - **TensorBoard** for visualization.
 - **Robust support for production deployment.**
 - Integration with Keras as high-level API.
- Cons
 - **Steeper learning curve** compared to some other frameworks.
 - Graph-based computation can be **complex to debug** [TF v1.0]

Keras

- Developed by: François Chollet
- Language: Python
- Highlights:
 - High-level API built on top of TF
 - Excellent for beginners and quick prototyping
 - Simple, readable, and concise code.
- Downsides:
 - Not ideal for low-level operations.
 - **Less control over model optimization process**

PyTorch

- Developed by: Facebook's AI Research lab
- Language: Python, C++, CUDA
- Highlights:
 - Dynamic computation graph (eager execution) makes it more intuitive and **easier to debug**.
 - Excellent community and academic adoption.
 - **Integrated with Python libraries like NumPy.**
- Downsides:
 - Strong Ecosystem but not as TensorFlow.
 - Deployment in non-Python environments may require extra effort.

TensorFlow and PyTorch

- Together, both of the frameworks dominate the academia and market
- Both under continuous development for handling shortcomings
- Tensorflow offered v2 to solve many difficulties
 - But at that time, Pytorch already spread due to pythonic use / easy to code/debug
- So far, Tensorflow seems better for production apps, but this is changing also **overtime**

Feature	TensorFlow	PyTorch	Keras
Ecosystem	Strong	Strong	Moderate
Debugging	Moderate	Easy	Easy
Scalability	Excellent	Good	Moderate (Depends on Backend)
Flexibility	Good	Excellent	Moderate
Deployment	Excellent	Moderate	Good
Community Support	Strong	Strong	Strong
Pre-trained Models	Extensive	Extensive	Extensive
Languages Supported	Multiple	Multiple	Python

Important to learn identifying metics
before **comparisons**

“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”

