

PyTorch vs TensorFlow 2023-A Head-to-Head Comparison

PyTorch vs TensorFlow 2023-A Head-to-Head Comparison of the similarities and differences of the top deep learning frameworks.

Last Updated: 25 Nov 2022

GET ACCESS TO ALL PYTORCH PROJECTS

[VIEW ALL PYTORCH PROJECTS](#)



'Man and machine together can be better than the human'.

All thanks to deep learning frameworks like PyTorch, Tensorflow, Keras, Caffe, and DeepLearning4j for making machines learn like humans with special brain-like architectures known as [Neural Networks](#). The war of deep learning frameworks has two prominent competitors- PyTorch vs Tensorflow because the other frameworks have not yet been adopted widely. These frameworks simplify the process of humanizing machines with supremacy through accurate large-scale complex deep learning models. If you are new to [machine learning](#), it means that you have been wheeled by this incredible field of study and its limitless possibilities of building applications that have never been implemented without human intervention, *congratulations and welcome to the world of deep learning!*



Start Project

Table of contents

- [PyTorch vs Tensorflow 2023– Comparing the Similarities and Differences](#)
- [PyTorch vs TensorFlow – Google Trends](#)
- [PyTorch vs TensorFlow – Graph Generation and Definition – Dynamic vs Static](#)
- [PyTorch vs TensorFlow – Visualization – Tensorboard vs Visdom](#)
- [PyTorch vs TensorFlow –Model Deployment In Production](#)
- [PyTorch vs TensorFlow – Device Management](#)
- [PyTorch vs TensorFlow - Learning Curve](#)
- [PyTorch vs TensorFlow – Documentation, and Debugging](#)
- [PyTorch vs TensorFlow – Use Cases](#)
- [PyTorch vs Tensorflow- Which one should you choose?](#)

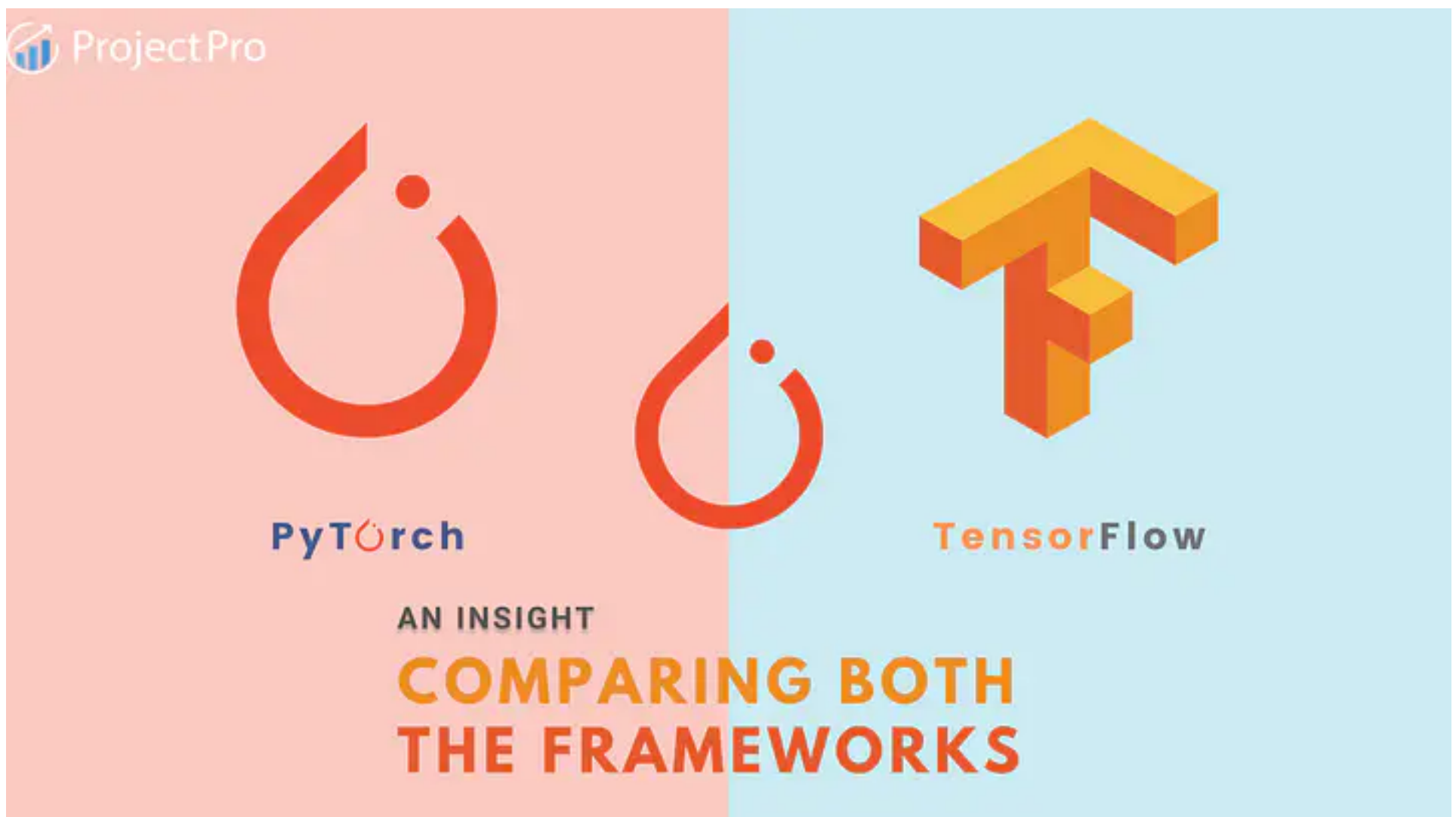
So, having played quite some time with neural networks you’ve finally decided to start your deep learning journey and become a deep learning engineer. Good call! Now, you probably want to dive into predicting who would take home the next Oscar, or are keen on generating faces of people who don’t exist at all with AI or simply want to detect apples and oranges on pictures. There are several machine learning and deep learning frameworks today that will help you achieve these goals. Each of these frameworks has a different purpose and is developed in a different manner for solving diverse business challenges. There are many deep learning frameworks but as a beginner, you will always have this question on “Which deep learning framework should I choose for my next [machine learning project](#)?”. The purpose of this article is to help you understand the similarities and differences between two of the most popular deep learning frameworks – PyTorch vs [Tensorflow](#). It does not matter whether you are a data scientist, researcher, student, machine learning engineer, or just a deep learning enthusiast, you’re definitely going to find the comparison of PyTorch and TensorFlow very useful.

PyTorch vs Tensorflow 2023– Comparing the Similarities and Differences

PyTorch vs. TensorFlow - A Head-to-Head Comparison



PyTorch and Tensorflow both are open-source frameworks with Tensorflow having a two-year head start to PyTorch. Tensorflow, based on Theano is Google’s brainchild born in 2015 while PyTorch, is a close cousin of Lua-based Torch framework born out of Facebook’s AI research lab in 2017.



You can read about the development of Tensorflow in the paper [“TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems.”](#) and PyTorch in this paper ["Automatic Differentiation in PyTorch."](#) PyTorch fits perfectly in the [machine learning ecosystem](#) as it is developed to be used in Python though it has a C++ interface. Compared to other declarative deep learning frameworks, PyTorch is popular for its imperative programming style which makes it more pythonic. PyTorch leverages the flexibility and popularity of the python programming language whilst maintaining the functionality and convenience of the native Torch library. In Tensorflow the computations are performed using high-performance C++ binaries but there are interfaces in several programming languages that ease the implementation of deep learning architectures for beginners. Tensorflow is in a relationship with TF 2.0 making its usability a lot more powerful. It comes with an intuitive high-level API tf.keras so one can enjoy all the low-level classes of TensorFlow together with the user-friendliness of Keras.

New Projects

End-to-End ML Model Monitoring using Airflow and Docker

[View Project](#)

Hands-On Approach to Regression Discontinuity Design Python

[View Project](#)

Talend Real-Time Project for ETL Process Automation

[View Project](#)

Learn to Create Delta Live Tables in Azure Databricks

[View Project](#)

Learn to Build a Siamese Neural Network for Image Similarity

[View Project](#)

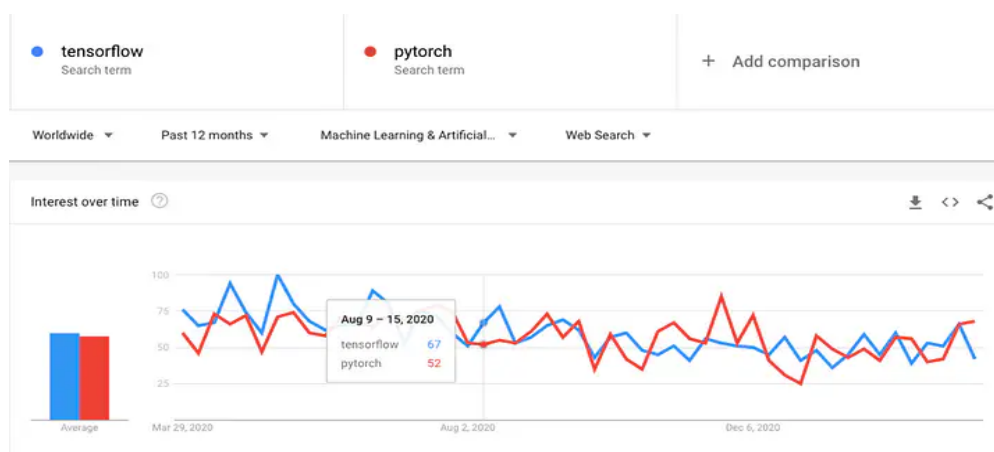
Build I and Sp Regres Model Python

[View](#)

[View all New Projects](#)

PyTorch vs TensorFlow – Google Trends

Just to show you a comprehensive picture of growth in demand and usage of PyTorch and Tensorflow deep learning frameworks, here is the Google trend graph for the keywords Tensorflow vs PyTorch -



You can also look at the number of job listings for PyTorch and Tensorflow to get a comprehensive picture of the demand for these deep learning skills –



pytorch

United States

Search

Jobs ▾

Date Posted ▾

Experience Level ▾

Company ▾

Pytorch in United States

4,442 results

Job Alert Off ☐



Principal ML Applied Scientist

Microsoft
Bellevue, WA



5 connections work here

Promoted



Data Scientist

Wpromote
Los Angeles Metropolitan Area



1 alum works here

Promoted



Machine Learning Engineer, Personal Intelligence

Apple
Cupertino, CA

\$138K – \$220K



493 alumni work here

14 hours ago

M
a
a
n
e



🔍 tensorflow

📍 United States

Search

Jobs ▾

Date Posted ▾

Experience Level ▾

Company ▾

Tensorflow in United States

6,143 results

Job Alert Off



Artificial Intelligence Researcher

New York City Metropolitan Area

Promoted • Apply now



Deep Learning Researcher - Speech Recognition

AssemblyAI

United States • Remote



Actively recruiting

Promoted • Apply now



Content Strategist

Y Media Labs

San Francisco Bay Area • Remote



4 alumni work here

1 week ago • Apply now

PyTorch vs TensorFlow – Graph Generation and Definition – Dynamic vs Static

Every deep learning framework depends on the generation of computation graphs for calculating the gradient values needed for gradient descent optimization. Having built the forward propagation graph, the deep learning frameworks tackle the backward differentiation. The reason for having computational graphs is to achieve parallelism and speed up the training process. There are usually two types of graphs – Static and Dynamic.



PyTorch

Dynamic Computation Graphs

Graphs can be built at runtime



TensorFlow

Graphs construction is **static**.
Graphs **need to be compiled** first.
Eager Execution in TF 2.0 provides dynamism.



PyTorch

Debugging PyTorch is same Python code.
pdb - Python Debugger can be used.



TensorFlow

tfdbg - Learn TensorFlow Debugger
Request variable from session to inspect.

 **ProjectPro**

In a static computational graph, you just define the neural network architecture and run it through a bunch of data to train the model and make predictions while a dynamic computational graph is defined implicitly once the forward propagation is executed. One just needs to write the forward computation with all the features and the computational graph is built on-demand dynamically offering flexibility. Dynamic computational graphs are easy to debug, unlike static computational graphs. The computational graphs form an important design differentiating factor between [TensorFlow](#) and PyTorch. PyTorch makes use of dynamic computational graphs while TensorFlow creates optimized static computational graphs. However, TensorFlow 2.0 has eager execution by default that makes the graph generation similar to dynamic computational graphs wherein operations are evaluated immediately without generating a computational graph to be run later. So, as of today, you can have both static and dynamic modes of graph generation in both frameworks.

[Click here to view a list of 50+ solved, end-to-end Big Data and Machine Learning Project Solutions \(reusable code + videos\)](#)

PyTorch 1.8 vs Tensorflow 2.x in 2021

What's New in TensorFlow 2.x in 2021

- With the increasing demand for deploying models in the browser, the latest version of Tensorflow introduced tensorflow.js that lets users deploy existing python models within the browser using Node. Tensorflow.js makes it possible to develop and train machine learning models using Javascript.
- Mobile and Web applications are the two most dominant types of applications, TensorFlow 2.x introduced a novel lightweight library TensorFlow Lite that allows model deployment on mobile and embedded devices by converting them into a compressed flat buffer.
- A major release in TensorFlow 2.x is TFX developed to overcome the traditional software production challenges like modularity, scalability, and maintainability. TFX (TensorFlow Extended) introduced in Tensorflow 2.x is an end-to-end platform to deploy and productionize machine learning pipelines.

Get Closer To Your Dream of Becoming a Data Scientist with 70+ Solved [End-to-End ML Projects](#)

What's New in PyTorch 1.8 in 2021

- With over 3000 commits, PyTorch 1.8 has introduced better features for code optimization, compilation, and front-end apis for scientific computing.
- The release of the PyTorch profiler makes debugging and troubleshooting of large-scale deep learning models easier than before.
- There have been a lot of improvements and updates to PyTorch Mobile to help users launch models on mobile devices faster.
- PyTorch Lightning (the Keras of PyTorch) has been released to ease and shorten the process of implementing neural networks easier.

PyTorch vs TensorFlow – Visualization – Tensorboard vs Visdom

Suppose that you are developing a deep learning model for image classification with 98% desired accuracy. You will have to track the loss and accuracy of your deep learning model at every epoch. Also, you will want to profile the loss and accuracy of your model with various hyperparameters like optimizers, learning rate, drop out, and others to fine-tune the deep learning model. Deep learning

models are pricey in terms of computation so to track the accuracy of the deep learning models on different data flow graphs with different values of hyperparameters, it is important to visualize the results and get a bird's eye view of the overall training process of the model.

Both PyTorch and TensorFlow support visualization tools for quick analysis making it easier to review the training process of the deep learning models and ease the process of debugging. These visualization tools are also great for presenting the results. TensorFlow's Tensorboard has an awesome in-built visualization tool with a suite of web apps for understanding a deep learning model through 5 different visualizations- graphs, scalars, audio, histograms, and images. PyTorch's native visualization tool visdom is flexible, customized, lightweight, and easy to use with support for PyTorch tensors. Visdom is not interactive and has very limited and minimalistic features for overviewing data. However, the latest release of PyTorch 1.2.0 has PyTorch built-in Tensorboard.

Ace Your Next Job Interview with Mock Interviews from Experts to Improve Your Skills and Boost Confidence!

Get Started

PyTorch vs TensorFlow –Model Deployment In Production

Developing and training a deep learning model is just half the job done. Deploying these trained deep learning models into production and managing them is the most difficult part. TensorFlow has an in-built model deployment tool TensorFlow Serving which is used by most of the Google projects. Having been tested on over 1000 Google projects, TensorFlow Serving is capable of handling millions of requests per second. It is used for deploying machine learning models on specialized gRPC servers and provides remote access to them. [Model deployment with TensorFlow](#) Serving is highly flexible and it also integrates perfectly with Kubernetes and Docker. Using TensorFlow serving, it is easy to update a deployed model and also rollback any earlier versions of the deployed model without having to shut down the server. TensorFlow Serving is designed for industrial production environments and is a good choice where performance is a concern. While a 10% faster runtime might not mean a lot to a researcher working with these frameworks but it definitely helps organizations save millions.

PyTorch users have got TorchServe, a flexible model deployment tool in April 2020. The model deployment tool for PyTorch is in its infancy and in the experimental stage, unlike TensorFlow Serving that offers many functionalities. However, with a basic set of features like a model archiver tool, a server, metrics, logging, an API endpoint specification, and model snapshots, TorchServe attracts most of the business cases. The latest version of TorchServe also supports HuggingFace, Nvidia Waveglow, AWS Cloud Formation, and others.

Free access to solved code examples can be found [here \(these are ready-to-use for your ML projects\)](#).

PyTorch vs TensorFlow – Device Management

There can be a huge difference in the performance of neural network models when managing devices. Both these frameworks apply neural networks perfectly, however, the way they execute is different. A TensorFlow operation has both GPU and CPU implementations but a GPU device would always be given a priority to run the operation unless you explicitly request it to run on another device. The GPU acceleration is automated in TensorFlow meaning there is no control over memory usage. So, even if one GPU is in use, it will consume the memory of all available GPUs.



On the contrary, PyTorch does not automate GPU usage and does not have a dedicated library for GPU users. It uses CUDA to specify the usage of CPU or GPU. The deep learning model does not run without CUDA specifications for CPU or GPU use so the training process enhances when using PyTorch as there is better control over the use of resources.

Explore Categories

[Data Science Projects in Python](#)

[Data Science Projects in R](#)

[Machine Learning Projects in Python](#)

[Deep Learning Projects](#)

[Neural Network Projects](#)

[Tensorflow Projects](#)

[H2O R Projects](#)

[IoT Projects](#)

[Keras Deep Learning Projects](#)

[NLP Projects](#)

[Pytorch](#)

[Data Science Projects in Banking and Finance](#)

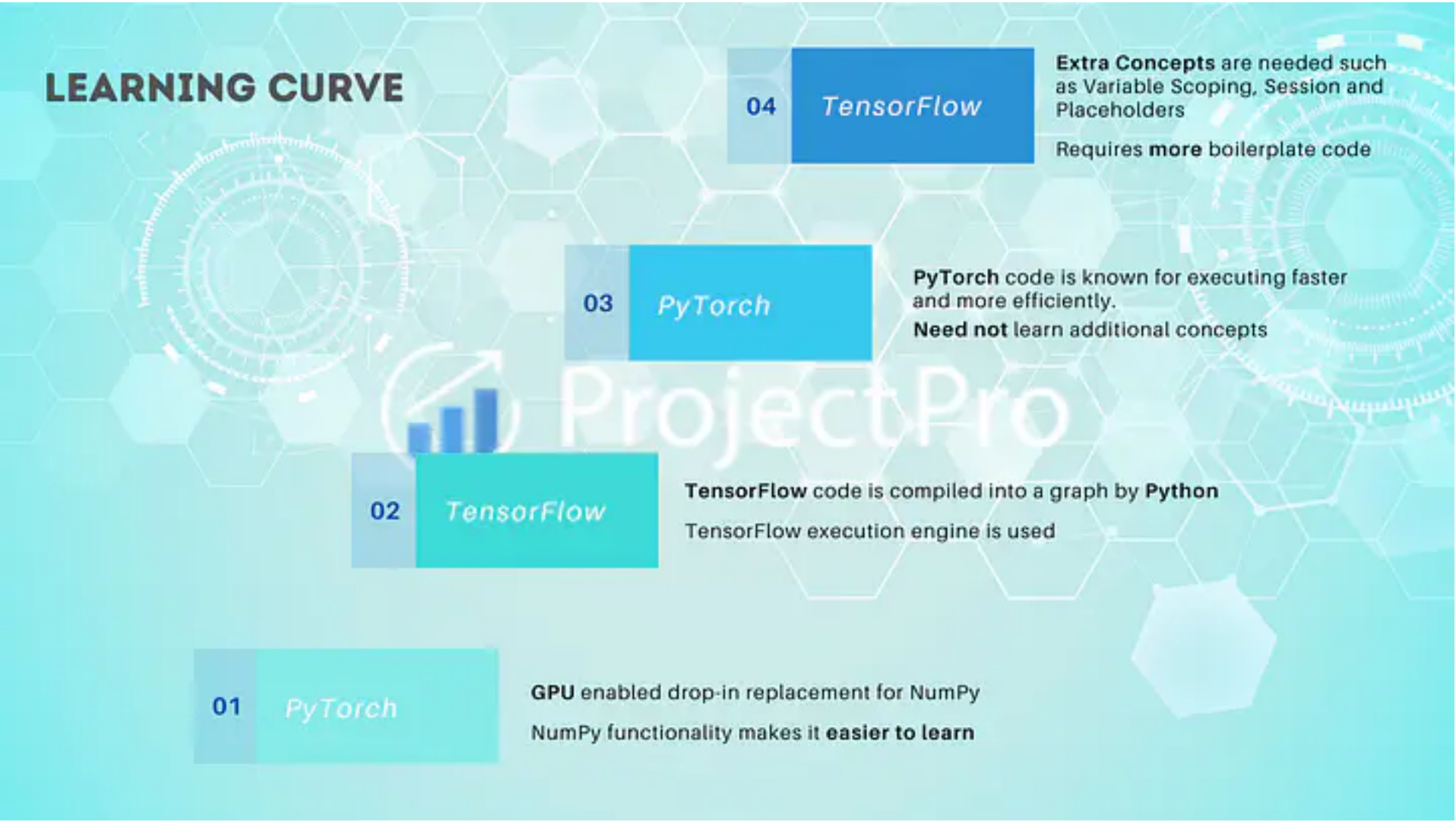
[Data Science Projects in Retail & Ecommerce](#)

[Data Science Projects in Entertainment & Media](#)

[Data Science Projects in Telecommunications](#)

PyTorch vs TensorFlow - Learning Curve

The learning curve for a deep learning framework depends on the previous expertise level and also on the purpose of using the framework. For machine learning beginners who are just getting started with deep learning, TensorFlow would be a little hard to comprehend because of the low-level implementations of the neural network structure. However, its very high-level Keras API makes it easy for beginners to learn the basic concepts, and later TensorFlow can be used for a better understanding of the implementations of neural network structure.



PyTorch is easy to learn when compared to TensorFlow because the syntax is very much similar to the traditional Python programming language and it is easier to experiment if you already know Python. With a more object-oriented style and straightforward data handling, the learning curve for PyTorch is easier compared to TensorFlow. Though it might not be feature-rich the ease of learning makes it one of the most loved deep learning frameworks among beginners, unlike TensorFlow where you need to understand the working of sessions and placeholders.

Get confident to build end-to-end projects.

Access to a curated library of 250+ end-to-end industry projects with solution code, videos and tech support.

PyTorch vs TensorFlow – Documentation, and Debugging

PyTorch users need not learn to use another debugger from scratch as debugging of PyTorch code can be done with the use of a standard python debugger. As we already discussed above that PyTorch defines computational graphs at run time so it is easier to use most of the regular python libraries and tools in PyTorch. So to debug PyTorch model code, one can use Python debugging tools like PyCharm, pdb, and ipdb. When it comes to debugging code in TensorFlow one needs to learn about the TF debugger and also about the variables that are requested from a TensorFlow session which is complex compared to debugging in PyTorch.



With a focussed community of developers, the documentation for both PyTorch and TensorFlow is very well organized and helpful for people getting started with these deep learning frameworks. There are tons of free tutorials, MOOCs, and example implementations available for both these frameworks. With lots of support and a huge community, both these frameworks are like magic wands to do some machine learning magic.

Get More Practice, More [Data Science and Machine Learning Projects](#), and More guidance. Fast-Track Your Career Transition with ProjectPro

PyTorch vs TensorFlow – Use Cases

When you are searching for a term or a phrase on Google, it automatically suggests the next word you are most likely to type, it's TensorFlow in action that predicts the next word for you. Whether it is Google Search, Google Translate, GMaps, Photos, Google Play, Android, or YouTube, TensorFlow is used by several internal Google projects and also for other applications like –

- Video Detection
- Image Recognition
- Voice and Speech Recognition
- Text-Based Applications

Common applications of PyTorch include -

- Handwritten Digit Recognition
- Text Generation
- Style Transfer
- Forecasting Time Sequences
- Image Classification

Recommended Reading

- [CNN vs RNN- Choose the Right Neural Network for Your Project](#)

PyTorch vs Tensorflow- Which one should you choose?

TensorFlow and PyTorch have become the de facto standards for the research and development of innovative custom deep learning applications because of the heavy lifting features they offer in terms of managing the underlying hardware, computation, and visualization. There were some drastic differences between the two frameworks earlier but as of today each of these frameworks has adopted all the good features from each other and have fared well in the battle. Considering the similarities between the two frameworks today, it is now possible to easily transition back and forth among the two frameworks.

Most Watched Projects

Snowflake Real Time Data Warehouse Project for Beginners-1

View Project

Linear Regression Model Project in Python for Beginners Part 1

View Project

Learn to Build a Polynomial Regression Model from Scratch

View Project

AWS Snowflake Data Pipeline Example using Kinesis and Airflow

View Project

Build Piecewise and Spline Regression Models in Python

View Project

Snowflake Real Time Data Warehouse Project for Beginners-2

View Project

View all Most Watched Projects

It is easy to learn PyTorch making it suitable for passion projects that require building prototypes in research while TensorFlow is a deep learning framework with production readiness. That being said, which deep learning framework should you choose to build your first deep learning model? The answer to this question depends on several factors but for people getting started with deep learning here is what our Project Advisors have to say – “Explore both Tensorflow and PyTorch before committing to learning any one of these deep learning frameworks in depth. Obviously, considering the best case you will be a master of both PyTorch and Tensorflow, however, at times it might not be possible to learn both. It is not necessary to know everything about each of these frameworks, just weigh your requirements before deciding to learn a deep learning framework in depth. Have fun while you discover the fantastic features of the framework rather than worrying about not being able to accomplish something too complex. Choose a deep learning framework based on the above listed best conditions for you to realize your deep learning projects”

If you are exploring a career in the deep learning industry get started with learning Tensorflow. However, if you are a researcher then it would be a good step to get started with PyTorch. Master real-world deep learning skills in Tensorflow and PyTorch with ProjectPro library of solved end-to-end deep learning projects that come with datasets, verified reusable solution code, 2-4 hours of video solution, documentation, and email technical support.

[PREVIOUS](#)

[NEXT](#)

Your Data Skills Need to Get Stronger

And We Have The Projects Ready

GET ACCESS TO SOLVED PROJECTS



Start Your First Project
Learn By Doing

Your phone

Your email

START PROJECT

Trending Project Categories

- Machine Learning Projects
- Data Science Projects
- Deep Learning Projects
- Big Data Projects
- Apache Hadoop Projects
- Apache Spark Projects
- Show more

Trending Projects

- Walmart Sales Forecasting Data Science Project
- BigMart Sales Prediction ML Project
- Music Recommender System Project
- Credit Card Fraud Detection Using Machine Learning
- Resume Parser Python Project for Data Science
- Time Series Forecasting Projects
- Show more

Trending Blogs

- Machine Learning Projects for Beginners with Source Code
- Data Science Projects for Beginners with Source Code
- Big Data Projects for Beginners with Source Code
- IoT Projects for Beginners with Source Code
- Data Analyst vs Data Scientist
- Data Science Interview Questions and Answers
- Show more

Trending Recipes

- Search for a Value in Pandas DataFrame
- Pandas Create New Column based on Multiple Condition
- LSTM vs GRU
- Plot ROC Curve in Python
- Python Upload File to Google Drive
- Optimize Logistic Regression Hyper Parameters
- Show more

Trending Tutorials

- PCA in Machine Learning Tutorial
- PySpark Tutorial
- Hive Commands Tutorial
- MapReduce in Hadoop Tutorial
- Apache Hive Tutorial -Tables
- Linear Regression Tutorial
- Show more

ProjectPro

© 2023 Iconiq Inc.

About us

Contact us

Privacy policy

User policy

Write for ProjectPro