Code Basic for python:
-> Python Playlist
    ->pandas(1-6)
    ->numpy
    ->matplotlib
Learn From python.org website

Deep learning Frameworks:

Top 8 Deep Learning Frameworks You Should Know in 2024

Lesson 6 of 28

By Simplilearn

Last updated on Nov 7, 2023

72610

Previous

Next

Table of Contents

TensorFlowKerasPyTorchTheanoDeeplearning4j (DL4J)View More
In today's world, more and more organizations are turning to machine learning and

artificial intelligence (AI) to improve their business processes and stay ahead of the

competition.

The growth of machine learning and AI has enabled organizations to provide smart

solutions and predictive personalizations to their customers. However, not all

organizations can implement machine learning and AI for their processes due to various

reasons.

This is where the services of various deep learning frameworks come in. These are interfaces, libraries, or tools, which are generally open-source that people with little to no knowledge of machine learning and AI can easily integrate. Deep learning frameworks can help you upload data and train a deep learning model that would lead to accurate and intuitive predictive analysis.

TensorFlow

Google's Brain team developed a Deep Learning Framework called TensorFlow, which supports languages like Python and R, and uses dataflow graphs to process data. This is very important because as you build these neural networks, you can look at how the data flows through the neural network.

TensorFlow's machine learning models are easy to build, can be used for robust machine learning production, and allow powerful experimentation for research.

With TensorFlow, you also get TensorBoard for data visualization, which is a large package that generally goes unnoticed. TensorBoard simplifies the process for visually displaying data when working with your shareholders. You can use the R and Python visualization packages as well.

Initial release:	November 9, 2015
Stable release:	2.4.1 / January 21, 2021
Written in:	Python, C++, CUDA
Platform:	Linux, macOS, Windows, Android, JavaScript
Type:	Machine learning library
Repository	github.com/tensorflow/tensorflow

License:	Apache License 2.0
Website	www.tensorflow.org

Keras

Francois Chollet originally developed Keras, with 350,000+ users and 700+ open-source contributors, making it one of the fastest-growing deep learning framework packages.

Keras supports high-level neural network API, written in Python. What makes Keras interesting is that it runs on top of TensorFlow, Theano, and CNTK.

Keras is used in several startups, research labs, and companies including Microsoft Research, NASA, Netflix, and Cern.

Other Features of Keras:

- User-friendly, as it offers simple APIs and provides clear and actionable feedback upon user error
- Provides modularity as a sequence or a graph of standalone, fully-configurable modules that can be combined with as few restrictions as possible
- Easily extensible as new modules are simple to add, making Keras suitable for advanced research

Initial release:	March 27, 2015
Stable release:	2.4.0 / June 17, 2020
Platform:	Cross-platform
Type:	Neural networks
Repository	github.com/keras-team/keras
License:	Massachusetts Institute of Technology (MIT)

Website	https://keras.io/

PyTorch

Adam Paszke, Sam Gross, Soumith Chintala, and Gregory Chanan authored PyTorch and is primarily developed by Facebook's AI Research lab (FAIR). It's built on the Lua-based scientific computing framework for machine learning and deep learning algorithms. PyTorch employed Python, CUDA, along with C/C++ libraries, for processing and was designed to scale the production of building models and overall flexibility. If you're well-versed with C/C++, then PyTorch might not be too big of a jump for you.

PyTorch is widely used in large companies like Facebook, Twitter, and Google.

Other Features of the Deep Learning Framework Include:

- It provides flexibility and speed due to its hybrid front-end.
- Enables scalable distributed training and performance optimization in research and production using the "torch distributed" backend.
- Deep integration with Python allows popular libraries and packages to be quickly write neural network layers in Python.

Initial release:	September 2016
Stable release:	1.7.1 / December 10, 2020
Platform:	IA-32, x86-64
Type:	Library for machine learning and deep learning
Repository	github.com/pytorch/pytorch
License:	Berkeley Software Distribution (BSD)

Website	https://pytorch.org/

Theano

The University de Montreal developed Theano, written in Python and centers around NVIDIA CUDA, allowing users to integrate it with GPS. The Python library allows users to define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays.

Initial release:	2007
Stable release:	1.0.5 / July 27, 2020
Platform:	Linux, macOS, Windows

Type:	Machine learning library
Repository	github.com/pytorch/pytorch
License:	The 3-Clause Berkeley Software Distribution (BSD)
Website	http://www.deeplearning.net/software/theano/

- NumPy • Theano has tight integration with NumPy for data computations
- GPU • Uses GPU's to perform data-intensive computations which are much faster than on a CPU
- • Has extensive unit-testing and self-verification that can detect and diagnose many types of errors

Deeplearning4j (DL4J)

A machine learning group that includes the authors Adam Gibson Alex D. Black, Vyacheslav Kokorin, Josh Patterson developed this Deep Learning Framework Deeplearning4j. Written in Java, Scala, C++, C, CUDA, DL4J supports different neural networks, like CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), and LSTM (Long Short-Term Memory).

After Skymind joined the Eclipse Foundation in 2017, DL4J was integrated with Hadoop and Apache Spark. It brings AI to business environments for use on distributed CPUs and GPUs.

Other Features of DL4J Include:

- A distributed computing framework as training with DL4J occurs in a cluster
- An n-dimensional array class using ND4J that allows scientific computing in Java and Scala
- A vector space modeling and topic modeling toolkit that is designed to handle large text sets and perform NLP

Initial release:	2014
Stable release:	1.0.0-beta6 / September 10, 2019

Platform:	Cross-platform
Type:	Natural Language Processing(NLP), Deep Learning, Machine Vision, Artificial Intelligence(AI)
Repository	github.com/deeplearning4j/deeplearning4j
License:	Apache License 2.0
Website	www.deeplearning4j.org

Caffe

Developed at BAIR or Berklee Artificial Intelligence Research and created by Yangqing Jia, Caffe stands for Convolutional Architecture for Fast Feature Embedding. Caffe is

written in C++ with a Python Interface and is generally used for image detection and classification.

Other Features and Uses of Caffe:

- Used in academic research projects, startup prototypes, and large-scale industrial applications in vision, speech, and multimedia
- Supports GPU- and CPU-based acceleration computational kernel libraries, such as NVIDIA, cuDNN, and IntelMLK
- Can process over 60M images per day with a single NVIDIA K40 GPU

Stable release:	1.0 / April 18, 2017
Type:	Library for Deep Learning
Repository	https://github.com/BVLC/caffe

Website	http://caffe.berkeleyvision.org/

Chainer

Developed by PreferredNetworks in collaborations with IBM, Intel, Microsoft, and Nvidia, Chainer is written purely in Python. Chainer runs on top of Numpy and CuPy Python libraries and provides several extended libraries, like Chainer MN, Chainer RL, Chainer CV, and many other libraries.

Other Features of Chainer:

- Supports CUDA computation
- Requires only a few lines of code to leverage a GPU
- uns on multiple GPUs with little effort
- Provides various network architectures, including feed-forward nets, convents, recurrent nets, and recursive nets

Author:	Seiya Tokui

Initial release:	June 9, 2015
Stable release:	7.7.0 / July 30, 2020
Platform:	Cross-Platform
Type:	Deep Learning library
Repository	https://github.com/chainer/chainer
License:	Massachusetts Institute of Technology (MIT)

Website	https://chainer.org/

Microsoft CNTK

Microsoft Research developed CNTK, a deep learning framework that builds a neural network as a series of computational steps via a direct graph. CNTK supports interfaces such as Python and C++ and is used for handwriting, speech recognition, and facial recognition.

Other Features of Microsoft CNTK Include:

- Designed for speed and efficiency, CNTK scales well in production using GPUs but has limited support from the community
- Supports both RNN and CNN type of neural models capable of handling image, handwriting, and speech recognition problems

Initial release:	January 25, 2016

Stable release:	2.7.0 / April 26, 2019
Type:	Library for Machine Learning and Deep Learning
Repository	https://github.com/Microsoft/CNTK
License:	Massachusetts Institute of Technology (MIT)
Website	https://www.microsoft.com/en-us/cognitive-toolkit/