Data Science – Deep Learning - Libraries

2. Deep Learning – Libraries

Contents

1.	. Tensorflow	2
	. Pytorch	
	. Keras	
	. Steps to create deep learning models with Keras	
	4.1. Define your model	
	4.2. Compile your model	
	4.3. Fit your model	
	4.4 Make predictions	

2. Deep Learning – Libraries

1. Tensorflow

- ✓ Tensorflow is a deep learning framework
- ✓ This library was created by Google in the year of 2015 and it is open source.
- ✓ TensorFlow is the most famous deep learning library.
- ✓ It is entirely based on Python programming language and use for numerical computation and data flow, which makes machine learning faster and easier.
- ✓ TensorFlow is based on graph computation

Logo



Tensorflow installation

pip install tensorflow

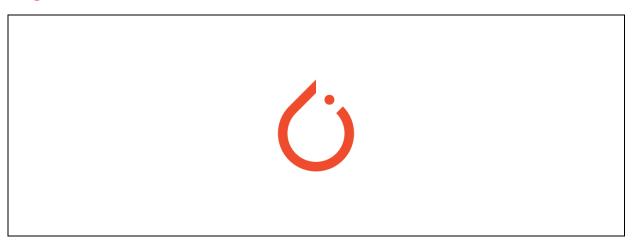
Update

- ✓ Tensorflow 2nd version onwards keras library was in built
- ✓ So, we no need to install keras separately

2. Pytorch

- ✓ Pytorch is a deep learning framework
- ✓ Pytorch is a deep learning library created by Facebook in the year of 2016 and it is an open source

Logo



Note

✓ CNTK is another deep learning framework created by Microsoft but not that much popular compare to tensorflow and pytorch

3. Keras

- ✓ Keras is an open-source high-level Neural Network library, which was written in Python, is capable enough to run on Theano, TensorFlow, or CNTK.
- ✓ It was developed by one of the Google engineers, François Chollet.
- ✓ It is user-friendly, extensible, and faster experimentation with deep neural networks.
- ✓ It supports Convolutional Networks and Recurrent Networks individually also their combination.



Data Science – Deep Learning - Libraries

4. Steps to create deep learning models with Keras

4.1. Define your model.

✓ Create a Sequential model and add configured layers.

4.2. Compile your model.

✓ Specify loss function and optimizers and call the compile() method on the model

4.3. Fit your model.

✓ Train the model on a sample of data by calling the fit () method on the model.

4.4. Make predictions

✓ Use the model to generate predictions on new data by calling the methods such as evaluate() or predict() on the model