Assignment 1.

Aim : Study of Deep learning Packages: Tensorflow, Keras, Theano and PyTorch. Document the distinct features and functionality of the packages.

Theory :

Tensorflow

Introduction:

The Tensorflow framework is an open end-to-end machine learning platform. It's a symbolic math toolkit that integrates data flow and differentiable programming to handle various tasks related to deep neural network training and inference. It enables programmers to design machine learning applications utilizing multiple tools, libraries, and open-source resources.

TensorFlow takes data in tensors, multi-dimensional arrays with more excellent dimensions. When dealing with enormous amounts of data, multi-dimensional arrays come in helpful. TensorFlow code is considerably easier to execute in a distributed way across a cluster of computers when utilizing GPUs because the execution mechanism is in the form of graphs.

TensorFlow Example

To give a concrete example, Google users can experience a faster and more refined search experience with AI. If the user types a keyword in the search bar, Google provides a recommendation about what could be the next word.

Google wants to use machine learning to take advantage of their massive datasets to give users the best experience. Three different groups use machine learning:

Researchers

Data Scientists

Programmers

They can all use the same toolset to collaborate with each other and improve their efficiency.

Google does not just have any data; they have the world's most massive computer, so Tensor Flow was built to scale. TensorFlow is a library developed by the Google Brain Team to accelerate machine learning and deep neural network research.

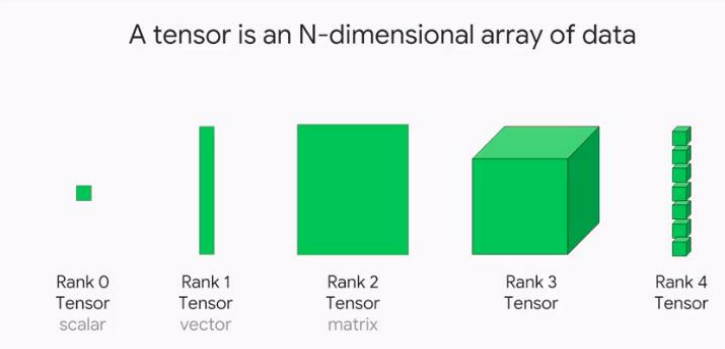
Components of TensorFlow

Tensorflow is named after the Tensor framework that it is built on. Tensors are used in every Tensorflow calculation. Any data can be represented by a tensor, an n-dimensional vector or matrix. The tensor values are all of the same data types and have a known (or partially known) shape. The shape of the data is determined by the dimensions of the matrix or array.

A tensor can be derived from either the input or output data of a calculation. TensorFlow performs all of its actions within a graph. The graph is made up of a sequence of sequential computations, and each operation is called an op node, and they are all interconnected.

The actions and relationships between the nodes are depicted in the graph.

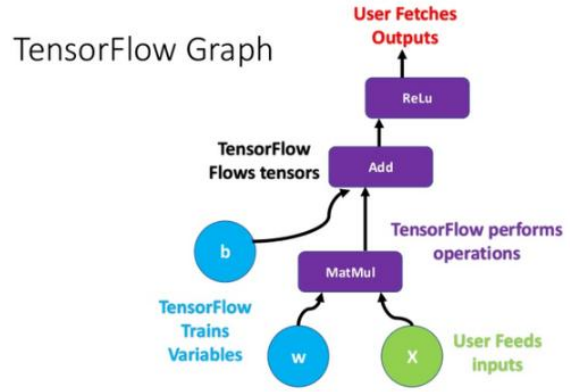
The values, on the other hand, are not displayed. The edge of the nodes in the Tensor is a means for populating the operation with data.



Graphs

TensorFlow is based on a graph-based architecture. This graph collects all of the training's series computations are collected and explained in this graph. The chart has a lot of advantages:

• The graph's portability allows computations to be saved for immediate or later use, and it was designed to run on numerous CPUs or GPUs and mobile devices.



The graph's computations are entirely done by linking tensors together. Thus, it may be saved and executed later.

The Design of TensorFlow and its Working

TensorFlow allows you to design dataflow graphs and structures to specify how data flows through a chart by receiving inputs as a multi-dimensional array called Tensor. It enables you to create a flowchart of operations that can be done on these inbound and outbound inputs.

This is available to programmers thanks to TensorFlow, which is written in Python. Python is easy to learn and use, and it provides straightforward methods for specifying how high-level abstractions should be linked. TensorFlow nodes and tensors are Python objects, as are TensorFlow applications.

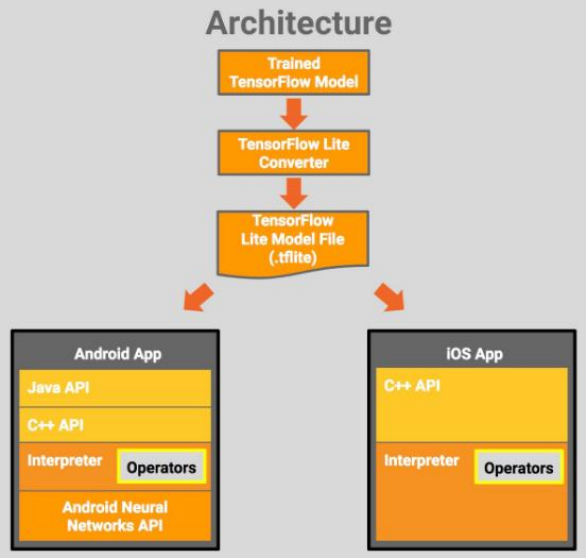
Python, on the other hand, does not do math operations. TensorFlow includes transformation libraries written as high-performance C++ binaries, and Python connects the components by routing data between them and providing high-level programming abstractions.

The architecture of TensorFlow is divided into three parts:

• Preparing the data

• Creating the model

• Prepare the model by training it and estimating it.



Tensorflow derives its name from the basic concept of taking input in a multi-dimensional array, commonly known as tensors. You can create a flowchart (called a Graph) of the processes you want to run on that input, and the data is entered at one end and output at the other.

Algorithms for TensorFlow

TensorFlow supports the following algorithms:

TensorFlow 1.10 includes a built-in API for:

• tf. estimator for linear regression,LinearRegressor

• Classification:tf.estimator,LinearClassifier

• tf. estimator.DNNClassifier for deep learning classification

• Wipe and deep learning: tf. estimator

DNNLinearCombinedClassifier

• tf. Estimator (booster tree regression).

BoostedTreesRegressor

Important features of TensorFlow −

It includes a feature of that defines, optimizes and calculates mathematical expressions easily with the help of multi-dimensional arrays called tensors.

It includes a programming support of deep neural networks and machine learning techniques.

It includes a high scalable feature of computation with various data sets.

TensorFlow uses GPU computing, automating management. It also includes a unique feature of optimization of same memory and the data used.

Keras

Keras is an open-source high-level Neural Network library, which is written in Python is capable enough to run on Theano, TensorFlow, or CNTK. It was developed by one of the Google engineers, Francois Chollet. It is made user-friendly, extensible, and modular for facilitating faster experimentation with deep neural networks. It not only supports Convolutional Networks and Recurrent Networks individually but also their combination.

It cannot handle low-level computations, so it makes use of the Backend library to resolve it. The backend library act as a high-level API wrapper for the low-level API, which lets it run on TensorFlow, CNTK, or Theano. It has an amazing industry interaction, and it is used in the development of popular firms likes Netflix, Uber, Google, E

Keras Backend

Keras being a model-level library helps in developing deep learning models by offering high-level building blocks. All the low-level computations such as products of Tensor, convolutions, etc. are not handled by Keras itself, rather they depend on a specialized tensor manipulation library that is well optimized to serve as a backend engine. Keras has managed it so perfectly that instead of incorporating one single library of tensor and performing operations related to that particular library, it offers plugging of different backend engines into Keras.

Keras consist of three backend engines, which are as follows:

TensorFlow

TensorFlow is a Google product, which is one of the most famous deep learning tools widely used in the research area of machine learning and deep neural network. It came into the market on 9th November 2015 under the Apache License 2.0. It is built in such a way that it can easily run on multiple CPUs and GPUs as well as on mobile operating systems. It consists of various wrappers in distinct languages such as Java, C++, or Python. xpedia, etc.

Theano

Theano was developed at the University of Montreal, Quebec, Canada, by the MILA group. It is an open-source python library that is widely used for performing mathematical operations on multi-dimensional arrays by incorporating scipy and numpy. It utilizes GPUs for faster computation and efficiently computes the gradients by building symbolic graphs automatically. It has come out to be very suitable for unstable expressions, as it first observes them numerically and then computes them with more stable algorithms.

CNTK

Microsoft Cognitive Toolkit is deep learning's open-source framework. It consists of all the basic building blocks, which are required to form a neural network. The models are trained using C++ or Python, but it incorporates C# or Java to load the model for making predictions.

Advantages of Keras

Keras encompasses the following advantages, which are as follows:

It is very easy to understand and incorporate the faster deployment of network models.

It has huge community support in the market as most of the AI companies are keen on using it.

It supports multi backend, which means you can use any one of them among TensorFlow, CNTK, and Theano with Keras as a backend according to your requirement.

Since it has an easy deployment, it also holds support for cross-platform. Following are the devices on which Keras can be deployed:

iOS with CoreML

Android with TensorFlow Android

Web browser with .js support

Cloud engine

Raspberry pi

It supports Data parallelism, which means Keras can be trained on multiple GPU's at an instance for speeding up the training time and processing a huge amount of data.

Disadvantages of Keras

The only disadvantage is that Keras has its own pre-configured layers, and if you want to create an abstract layer, it won't let you because it cannot handle low-level APIs. It only supports high-level API running on the top of the backend engine (TensorFlow, Theano, and CNTK).