Tensorflow

Introduction

- Google developed TensorFlowTM as an open-source software library for numerical computations with high throughput.
- As a result of its adaptable design, it may be easily deployed on a wide range of computing platforms (such as desktops, clusters of servers, mobile devices, and edge devices) and from a variety of sources.
- It was initially created by Google Brain researchers and was being used by Google employees.
- Incorporates deep neural networks for enhanced performance

Tensorflow's Benefits

- **Flexibility:** TensorFlow's architecture is very modular, allowing you to employ individual components or all of them together. Because of this, non-automatic migration to new models/versions, A/B testing experimental models, and canarying new models are all made possible through increased flexibility.

- **Portability:** TensorFlow's portability allows you to experiment with a concept on your laptop without the need for additional gear. Computers, laptops, and mobile devices may all use it because it runs on all of these types of hardware.

- **Research & Production:** Models can be trained and served to real clients in live mode using research and production software. Simply explained, rewriting codes isn't necessary, and industry researchers can put their ideas into products more quickly.

- **Auto Differentiation:** Gradient-based Machine Learning Algorithms Benefit from Auto Differentiation: It has automated differentiation capabilities.

- **Performance:** With its advanced support for threads, asynchronous computing and queues it helps you to get the most out of your hardware. Just distribute your TensorFlow graph's computation pieces across different devices and let the software take care of making copies. You can also run your computational graph using a variety of language options provided by the tool.

Usage of Tensorflow

- Deep Speech
    - Organization: Mozilla
    - Domain: Speech Recognition
    - Description: An implementation of TensorFlow inspired by Baidu's Deep Speech architecture.

- RankBrain
    - Organization: Google
    - Domain: Information Retrieval
    - Description: A large-scale deployment of deep neural networks for the purpose of ranking search results on www.google.com.

- Inception Image Classification Model
    - Organization: Google
    - Description: Research on extremely accurate computer vision models, beginning with the model that won the 2014 Imagenet picture classification challenge.

- SmartReply
    - Organization: Google
    - Description: A sophisticated LSTM model is used to produce email responses automatically.

- Massively Multitask Networks for Drug Discovery
    - Organization: Google and Stanford University
    - Domain: Drug discovery
    - Description: A deep neural network model with the purpose of identifying new drug candidates

- On device Computer vision for OCR
    - Organization: Google
    - Description: A computer vision model for optical character identification on-device in order to enable real-time translation.

Graph

- A computational graph is a collection of TensorFlow operations that have been connected to form a graph. The graph is made up of two distinct categories of objects.
- **Operations (or "ops"):** The graph's nodes. The term "operations" refers to mathematical operations that consume and produce tensors.
- Tensors are the graph's edges. These are the values that will be traversed by the graph. TensorFlow functions return tf.Tensors in the majority of cases.

Tensor

- TensorFlow's central data structure is the tensor. A tensor is a collection of primitive values that have been moulded into an array with any number of dimensions. The rank of a tensor is equal to the number of dimensions, and its shape is a tuple of numbers describing the length of the array along each dimension.
- The rank of a tensor is equal to the number of dimensions, and its shape is a tuple of numbers describing the length of the array along each dimension.

  [1., 2., 3.] # a rank 1 tensor; a vector with shape # a rank 1 tensor; a vector with shape [3]
  [[1., 2., 3.], [4., 5., 6.]] # a rank 2 tensor; a matrix with shape [2, 3]
  [[[1., 2., 3.]], [[7., 8., 9.]]] # a rank 3 tensor with shape [2, 1, 3]

Constants

- A constant is the most fundamental operation. The Python function that constructs the operation accepts as input a tensor value. The resulting operation does not require any inputs. When executed, it returns the value supplied to the function Object() { [native code] }.

  ```
  a = tf.constant(3.0, dtype=tf.float32)
  b = tf.constant(4.0)
  total = a + b
  print(a)
  print(b)
  print(total)
  ```

Output:
Tensor("Const 1:0", shape=(), dtype=float32)
Tensor("add:0", shape=(), dtype=float32)

Session

- Instantiate a tf.Session object, colloquially referred to as a session, to evaluate tensors. A session encapsulates the TensorFlow runtime's state and is used to execute TensorFlow operations.
- If a tf.Graph is comparable to a.py file, then a tf.Session is comparable to the Python executable.
- Multiple tensors may be passed to tf.
- Session.run. The run method manages any combination of tuples or dictionaries transparently.

Placeholders

- External inputs, referred to as placeholders, can be added to a graph via parameterization. A placeholder, similar to a function argument, is a promise to give a value later.

  x = tf.placeholder(tf.float32)
  x = tf.placeholder(tf.float32)
  z = x + y

Variables

- Additionally, we can initialise a variable in tensorflow. It is subject to alter over time. We use tf.get variable() for this purpose
- We use tf.get variable() for this purpose

File Types supported

- CSV (Comma separated values)
- TSV (tab separated values)
- Tfrecords (standard tensorflow format) : We have write a little programme that gets your data, stuffs it in an Example protocol buffer, serialises the protocol buffer to a string, and then writes the string to a TFRecords file using the tf.python io.TFRecordWriter
- Fixed length records (to read binary)

Conclusion:

In this article we have covered:

- Tensorflow introduction
- Benefits of tensorflow
- Usage of tensorflow
- Graph
- Tensor
- Constants
- Session
- Placeholders
- Variables
- File Types supported