Building Machine Learning models with TensorFlow
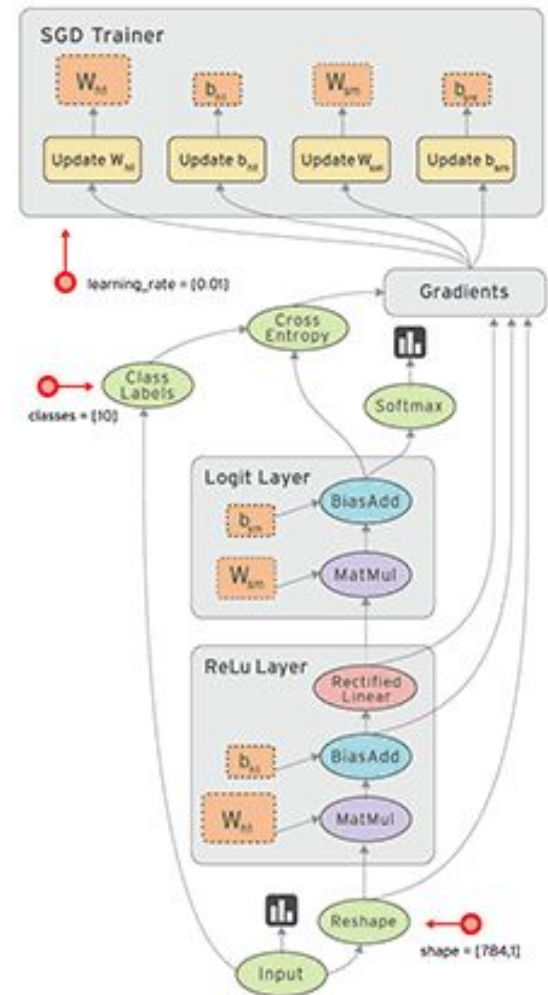
@SiAnakBatu

arif.romadhan@bukalapak.com

Bukalapak

BukaLaPak

TensorFlow Definition

TensorFlow is an open-source high-performance library for numerical computation that uses directed graph

TensorFlow toolkit hierarchy

High-level "out-of-box" API does distributed training → **tf.estimator**

Components useful when building custom NN models → **tf.layers, tf.losses, tf.metrics**

Python API gives you full control → **Core TensorFlow (Python)**

C++ API is quite low-level → **Core TensorFlow (C++)**

TF runs on different hardware → **CPU** **GPU** **TPU** **Android**

Cloud ML Engine

Bukalapak

Link access

1. https://github.com/arifromadhan19/tensorflow_mini_workshop (run in own laptop)

2. https://www.kaggle.com/arifromadhan19/kernels (run in kaggle kernel)

3. http://172.16.70.20:8890/tree/arif/tensorflow_mini_workshop (run in jupyter server)

TensorFlow toolkit hierarchy

13 Agustus 2019, @SiAnakBatu

The Python API

High-level "out-of-box" API does distributed training ⟶ **tf.estimator**

Components useful when building custom NN models ⟶ **tf.layers, tf.losses, tf.metrics**

Python API gives you full control ⟶ **Core TensorFlow (Python)**

C++ API is quite low-level ⟶ **Core TensorFlow (C++)**

TF runs on different hardware ⟶ **CPU** **GPU** **TPU** **Android**

Cloud ML Engine

Tensors and Operations

TensorFlow does have its own data structure for the purpose of performance and ease of use.

Tensor is the data structure used in Tensorflow and it is at the core of TensorFlow.

1. Constant
2. Variabel
3. Placeholder

https://www.easy-tensorflow.com/tf-tutorials/basics/tensor-types

The Python API
Build and run directed Graphs

```
...                                        Build
c = tf.add(a, b)
```

```
session = tf.Session()                     Run
numpy_c = session.run(c, feed_dict= ….)
```

a   b   ← Tensors

add ← Operation

c
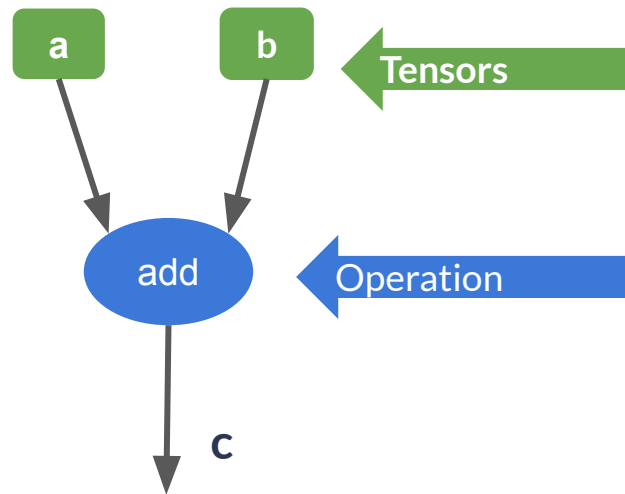
TensorFlow does lazy evaluation: you need to run the graph to get results

numpy

```
a = np.array([5, 3, 8])
b = np.array([3, -1, 2])
c = np.add(a, b)
print c

           [ 8  2 10]
```

TensorFlow

```
a = tf.constant([5, 3, 8])
b = tf.constant([3, -1, 2])
c = tf.add(a, b)                          Build
print c

         Tensor("Add_7:0", shape=(3,), dtype=int32)
```

```
with tf.Session() as sess:
  result = sess.run(c)
  print result                   Run

           [ 8  2 10]
```

tf Eager, however, allows you to execute operations imperatively

Let's do it
in Practice

Link : **https://www.kaggle.com/arifromadhan19/tensorflow**

http://172.16.70.20:8890/notebooks/arif/tensorflow_mini_wor

kshop/Simple_Neural_Network.ipynb

The High-level API

High-level "out-of-box" API does distributed training	→	**tf.estimator**
Components useful when building custom NN models	→	**tf.layers, tf.losses, tf.metrics**
Python API gives you full control	→	**Core TensorFlow (Python)**
C++ API is quite low-level	→	**Core TensorFlow (C++)**
TF runs on different hardware	→	**CPU** **GPU** **TPU** **Android**

Cloud ML Engine

Estimators

01   LinearRegressor

02   DNNRegressor

03   DNNLinearCombinedRegressor

04   LinearClassifier

05   DNNClassifier

06   DNNLinearCombinedClassifier

Working with Estimator

Set up machine learning model

1. Regression or classification?

2. What is the label?

3. What are the features?

Carry out ML steps

1. Train the model

2. Evaluate the model

3. Predict with the model

Square footage → **ML Model** → **Price**

Let's do it in Practice

Link :

http://172.16.70.20:8890/notebooks/arif/tensorflow_mini_workshop/2.%20Estimator%20in%20Classification%20Case.ipynb

Let's do it in Practice

Neural Network models

High-level "out-of-box" API does
distributed training
⟶ **tf.estimator**

Components useful when building
custom NN models
⟶ **tf.layers, tf.losses, tf.metrics**

Python API gives you full control
⟶ **Core TensorFlow (Python)**

C++ API is quite low-level
⟶ **Core TensorFlow (C++)**

TF runs on different hardware
⟶ **CPU** **GPU** **TPU** **Android**

Cloud ML Engine

Working with Neural Network models

Set up machine learning model

1. Regression or classification?

2. What is the label?

3. What are the features?

Square footage → **ML Model** → **Price**

Carry out ML steps

1. Train the model

2. Evaluate the model

3. Predict with the model

Let's do it
in Practice

—

Link :

https://www.kaggle.com/arifromadhan19/mnist-classification-tensorflow
https://www.kaggle.com/arifromadhan19/mnist-classification-keras

Bonus

Link :

https://developers.google.com/machine-learning/practica/image-classification/exercise-3

THANK ♥ YOU