COSC4337_100-TensorFlow-Basic-Syntax

1 TensorFlow Basics

[1]:
```
import tensorflow as tf
```

[2]:
```
# Make sure you are using 1.3 for exact sytnax matching!
print(tf.__version__)
```

```
1.15.0
```

1.1 Tensors

[3]:
```
hello = tf.constant('Hello')
```

[4]:
```
type(hello)
```

[4]:
```
tensorflow.python.framework.ops.Tensor
```

[5]:
```
world = tf.constant('World')
```

[6]:
```
result = hello + world
```

[7]:
```
result
```

[7]:
```
<tf.Tensor 'add:0' shape=() dtype=string>
```

[8]:
```
type(result)
```

[8]:
```
tensorflow.python.framework.ops.Tensor
```

[9]:
```
with tf.Session() as sess:
    result = sess.run(hello+world)
```

[10]:
```
result
```

[10]:
```
b'HelloWorld'
```

** Computations **

[11]:
```
tensor_1 = tf.constant(1)
tensor_2 = tf.constant(2)
```

[12]:
```
type(tensor_1)
```

[12]:
```
tensorflow.python.framework.ops.Tensor
```

[13]:
```
tensor_1 + tensor_2
```

[13]:
```
<tf.Tensor 'add_2:0' shape=() dtype=int32>
```

[14]:
```
sess
```

[14]:
```
<tensorflow.python.client.session.Session at 0x19e6d014ec8>
```

[15]:
```
sess.close()
```

1.2 Operations

[16]:
```
const = tf.constant(10)
```

[17]:
```
fill_mat = tf.fill((4,4),10)
```

[18]:
```
myzeros = tf.zeros((4,4))
```

[19]:
```
myones = tf.ones((4,4))
```

[20]:
```
myrandn = tf.random_normal((4,4),mean=0,stddev=0.5)
```

[21]:
```
myrandu = tf.random_uniform((4,4),minval=0,maxval=1)
```

[22]:
```
my_ops = [const,fill_mat,myzeros,myones,myrandn,myrandu]
```

1.3 Interactive Session

Useful for Notebook Sessions

[23]:
```
# Only run this cell once!
sess = tf.InteractiveSession()
```

[24]:
```
for op in my_ops:
    print(op.eval())
    print('\n')
```

```
10


[[10 10 10 10]
 [10 10 10 10]
```

```
 [10 10 10 10]
 [10 10 10 10]]


[[0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]]


[[1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]]


[[-8.2982250e-02 -4.7476265e-01  3.0409819e-01 -4.5937094e-01]
 [ 6.9448054e-01  6.6179568e-01 -3.5372025e-01 -3.9043581e-01]
 [-2.3814909e-01  6.3203096e-01  1.0370367e-01 -7.0188113e-02]
 [-3.1362766e-01  1.7542511e-04 -2.0142511e-01 -5.2723873e-01]]


[[0.6356571  0.4129572  0.92279005 0.54034007]
 [0.38770354 0.70097196 0.79907787 0.34641016]
 [0.00807869 0.7091018  0.7926662  0.6510047 ]
 [0.21936762 0.39348578 0.3323511  0.40567005]]
```

1.4 Matrix Multiplication

[25]:
```
a = tf.constant([ [1,2],
                  [3,4] ])
```

[26]:
```
a.get_shape()
```

[26]:
```
TensorShape([Dimension(2), Dimension(2)])
```

[27]:
```
b = tf.constant([[10],[100]])
```

[28]:
```
b.get_shape()
```

[28]:
```
TensorShape([Dimension(2), Dimension(1)])
```

[29]:
```
result = tf.matmul(a,b)
```

[30]:
```
result.eval()
```

[30]:
```
array([[210],
       [430]])
```

[ ]:

[ ]: