Chap 7 - TensorFlow Input

March 25, 2018

0.1 TensorFlow Input

To create a constant tensor object, we achieve it by:

```
import tensorflow as tf

hello_constant = tf.constant('Hello World!')
```

We use `tf.placeholder()` to create a non-constant tensor object.

0.1.1 tf.placeholder()

Over time, we want our TensorFlow model to take in *different datasets* with different parameters. That's why we can't use `tf.constant()`, but `tf.placeholder()`.

`tf.placeholder()` returns a tensor that gets its value from data passed to the `tf.session.run()` function, in which we set the input right before the session runs via `feed_dict`.

```
In [2]: # To not print out unwanted warnings
        import os
        os.environ["TF_CPP_MIN_LOG_LEVEL"]="3"

        import tensorflow as tf

        x = tf.placeholder(tf.string)
        with tf.Session() as sess:
            output = sess.run(x, feed_dict={x: 'Hello World'})
            print(output)
```

```
Hello World
```

The `feed_dict` parameter in `tf.session.run()` is used to set the placeholder tensor. In the above code, the tensor `x` is set to the string `Hello World`. It's also possible to set more than one tensor using `feed_dict`.

```
In [3]: x = tf.placeholder(tf.string)
        y = tf.placeholder(tf.int32)
        z = tf.placeholder(tf.float32)

        with tf.Session() as sess:
            output = sess.run(x, feed_dict={x: 'Sunday', y: 25, z: 22.14})
```

Note that the data passed to the `feed_dict` must match the tnesor type (or can be cast into that tensor type)

```
In [4]: # Case#1 - type mismatch
        x = tf.placeholder(tf.string)

        with tf.Session() as sess:
            output = sess.run(x, feed_dict={x: 20})


        ---------------------------------------------------------------------------

        TypeError                                 Traceback (most recent call last)

        <ipython-input-4-75e23b150fc9> in <module>()
          3
          4 with tf.Session() as sess:
        ----> 5     output = sess.run(x, feed_dict={x: 20})


        ~/tensorflow/lib/python3.5/site-packages/tensorflow/python/client/session.py in run(self
        903     try:
        904       result = self._run(None, fetches, feed_dict, options_ptr,
        --> 905                          run_metadata_ptr)
        906       if run_metadata:
        907         proto_data = tf_session.TF_GetBuffer(run_metadata_ptr)


        ~/tensorflow/lib/python3.5/site-packages/tensorflow/python/client/session.py in _run(sel
       1095                     type(subfeed_val)) +
       1096                 ' is not compatible with Tensor type ' + str(subfeed_dtype) +
       -> 1097                 '. Try explicitly setting the type of the feed tensor'
       1098                 ' to a larger type (e.g. int64).')
       1099


        TypeError: Type of feed value 20 with type <class 'int'> is not compatible with Tensor t


In [5]: # Case#2 - cast type is acceptable
        y = tf.placeholder(tf.int32)

        with tf.Session() as sess:
            output = sess.run(y, feed_dict={y: 20.0})
            print(output)

20
```

0.1.2 Quiz

Let's make the code return the number `123`.

```
In [6]: import tensorflow as tf

        def run():
            output = None
            x = tf.placeholder(tf.int32)

            with tf.Session() as sess:
                # TODO: Feed the x tensor 123
                output = sess.run(x, feed_dict={x: 123})

            return output

        run()
```

```
Out[6]: array(123, dtype=int32)
```