

[Tensorflow 학습 결과 저장 및 로드]

Tensorflow를 통해 특정 학습을 진행했을 때, 학습 결과를 저장하고 사용하게 된다.
(매번 학습 시킨 다음에 결과를 볼 수 없기 때문에)

이렇게 저장된 모델은 다시 Re-Training없이, 학습된 모델을 불러와서 결과를 확인 할 수 있다.

모델을 저장하기 위해서는 `tf.train.saver`를 이용한다.

위 코드와 같이 지금까지 학습된 Session를 특정 바이너리 파일로 저장을 한다.

saver를 이용하여 모델을 저장할 경우 2종류의 파일로 구분된다.

1) Meta graph - 'data/output_model.ckpt.meta'

meta파일은 Tensorflow graph에 대한 정보를 저장한다. 모든 Variable, Operation 등을 저장한다.

2) Checkpoint - 'data/output_model.ckpt.index', 'data/output_model.ckpt-00000-of-00001'

checkpoint file들은 학습을 통해 구해진 weight parameter들을 저장한다.

참고 : https://www.tensorflow.org/api_docs/python/tf/train/Saver#save

ts_ex01_saver.py

```
import tensorflow as tf
tf.set_random_seed(777)

W = tf.Variable(tf.random_normal([1]))
b = tf.Variable(tf.random_normal([1]))

X = tf.placeholder(tf.float32, shape=[None], name="input") #Restore 시 식별할 이름으로 사용될 수 있다.
Y = tf.placeholder(tf.float32, shape=[None], name="output") #Restore 시 식별할 이름으로 사용될 수 있다.

hypothesis = X * W + b
hypothesis = tf.identity(hypothesis, "hypothesis") #Restore 시 식별할 이름으로 사용될 수 있다.

cost = tf.reduce_mean(tf.square(hypothesis - Y))

optimizer = tf.train.GradientDescentOptimizer(learning_rate=0.01)
train = optimizer.minimize(cost)

sess = tf.Session()
sess.run(tf.global_variables_initializer())

trainX = [1, 2, 3]
trainY = [5, 10, 15]

for step in range(1001):
    cost_val, W_val, b_val, _ = sess.run([cost, W, b, train], feed_dict={X: trainX, Y: trainY})
    if step % 20 == 0:
        print(step, cost_val, W_val, b_val)

saver = tf.train.Saver()
save_path = saver.save(sess, 'data/output_model.ckpt')
```

ts_ex02_restore.py

```
import tensorflow as tf
tf.set_random_seed(777)

sess = tf.Session()
new_saver = tf.train.import_meta_graph('data/output_model.ckpt.meta')
new_saver.restore(sess, 'data/output_model.ckpt')

tf.get_default_graph()

x = sess.graph.get_tensor_by_name("input:0")
y = sess.graph.get_tensor_by_name("output:0")
hypothesis = sess.graph.get_tensor_by_name("hypothesis:0")
# "<op_name>:<output_index>"

result = sess.run(hypothesis, feed_dict={x:[1, 2, 3]})
print(result)
```