加载图：

model\_path = './tmp/finetune\_alexnet/n01440764/checkpoints/model\_epoch10.ckpt.**meta**'

saver = tf.train.import\_meta\_graph(model\_path)

方法一：【加载最后一次的模型】

log\_dir = './tmp/finetune\_alexnet/n01440764/checkpoints/ '

model\_dir = tf.train.get\_checkpoint(log\_dir)

saver.restore(sess,model\_dir)

方法二：【加载任意一次模型】

log\_dir = './tmp/finetune\_alexnet/n01440764/checkpoints/ **model\_epoch10.ckpt**'

#最后的命名跟保存时的命名相同。

Saver.restore(sess,log\_dir)

方法三：【定位到最新保存的模型】

log\_dir = './tmp/finetune\_alexnet/n01440764/checkpoints/ '

# 程序前面得有 Variable 供 save or restore 才不报错

# 否则会提示没有可保存的变量

saver = tf.train.Saver() 【不加载图，只加参数】

if ckpt and ckpt.model\_checkpoint\_path:

global\_step = ckpt.model\_checkpoint\_path.split('/')[-1].split('-')[-1]

saver.restore(sess, **ckpt.model\_checkpoint\_path**) **#这里可以加载任意模型。**

print('Loading success, global\_step is %s' % global\_step)

else:

print('No checkpoint file found')

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* # 连同图结构一同加载

ckpt = tf.train.get\_checkpoint\_state('./model/')

saver = tf.train.import\_meta\_graph(ckpt.model\_checkpoint\_path +'.meta')

with tf.Session() as sess:

    saver.restore(sess,ckpt.model\_checkpoint\_path)

# 只加载数据，不加载图结构，可以在新图中改变batch\_size等的值

# 不过需要注意，Saver对象实例化之前需要定义好新的图结构，否则会报错

saver = tf.train.Saver()

with tf.Session() as sess:

    ckpt = tf.train.get\_checkpoint\_state('./model/')

    saver.restore(sess,ckpt.model\_checkpoint\_path)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



.meta文件保存了当前图结构

.index文件保存了当前参数名

.data文件保存了当前参数值