

CSDN新首页上线啦，邀请你来立即体验！(http://blog.csdn.net/)

立即体验



博客 (//blog.csdn.net/?ref=toolbar) 学院 (//edu.csdn.net/?ref=toolbar)
下载 (//download.csdn.net/?ref=toolbar) GitChat (//gitbook.cn/?ref=csdn)
更多 ▾



登录 (https://passport.csdn.net/account/login?ref=toolbar) 注册 (http://passport.csdn.net/account/mobile/register?ref=toolbar&action=mobileRegister)
(https://passport.csdn.net/account/login?ref=toolbar) (https://passport.csdn.net/account/mobile/register?ref=toolbar&action=mobileRegister)

Tensorflow将模型导出为一个文件及接口设置

原创 2017年11月10日 21:13:44

259

在上一篇文章中《Tensorflow加载预训练模型和保存模型》(http://blog.csdn.net/huachao1001/article/details/78501928)，我们学习到如何使用预训练的模型。但注意到，在上一篇文章中使用预训练模型，必须至少的要4个文件：

| | |
|---|-----------------------------|
| 1 | checkpoint |
| 2 | MyModel.meta |
| 3 | MyModel.data-00000-of-00001 |
| 4 | MyModel.index |

这很不便于我们的使用。有没有办法导出为一个pb文件，然后直接使用呢？答案是肯定的。在文章《Tensorflow加载预训练模型和保存模型》(http://blog.csdn.net/huachao1001/article/details/78501928)中提到，meta文件保存图结构，weights等参数保存在data文件中。也就是说，图和参数数据时分开保存的。说的更直白一点，就是meta文件中没有weights等数据。但是，值得注意的是，meta文件会保存常量。我们只需将data文件中的参数转为meta文件中的常量即可！

1 模型导出为一个文件

1.1 有代码并且从头开始训练

Tensorflow提供了工具函数 tf.graph_util.convert_variables_to_constants () 用于将变量转为常量。看看官网的描述：

if you have a trained graph containing Variable ops, it can be convenient to convert them all to Const ops holding the same values. This makes it possible to describe the network fully with a single GraphDef file, and allows the removal of a lot of ops related to loading and saving the variables.

我们继续通过一个简单例子开始：

```
1 import tensorflow as tf
2
3 w1 = tf.Variable(20.0, name="w1")
4 w2 = tf.Variable(30.0, name="w2")
5 b1= tf.Variable(2.0,name="bias")
6 w3 = tf.add(w1,w2)
7
8 #记住要定义name，后面需要用到
9 out = tf.multiply(w3,b1,name="out")
10
11 # 转换Variable为constant，并将网络写入到文件
12 with tf.Session() as sess:
13     sess.run(tf.global_variables_initializer())
14     # 这里需要填入输出tensor的名字
15     graph = tf.graph_util.convert_variables_to_constants(sess, sess.graph_def, ["out"])
16     tf.train.write_graph(graph, '.', './checkpoint_dir/graph.pb', as_text=False)
```

执行可以看到如下日志：

| | |
|---|-------------------------------------|
| 1 | Converted 3 variables to const ops. |
|---|-------------------------------------|

可以看到通过 tf.graph_util.convert_variables_to_constants() 函数将变量转为了常量，并存储在graph.pb文件中，接下来看看如何使用这个模型。



huachao1001 (http://blog...)
(http://blog.csdn.net/huachao1001) 码云
0
(https://git/huachao?)
原创 57 粉丝 1246 喜欢 0

他的最新文章
更多文章 (http://blog.csdn.net/huachao1001)

Tensorflow加载预训练模型和保存模型
(http://blog.csdn.net/huachao1001/article/details/78501928)

TensorFlow中的队列 (http://blog.csdn.net/huachao1001/article/details/78083125)

相关推荐

tensorflow 保存和加载模型 -2 (http://blog.csdn.net/lujiaandong1/article/details/53301994)

如何用Tensorflow训练模型成pb文件和如何加载已经训练好的模型文件 (http://blog.csdn.net/u014432647/article/details/75276718)

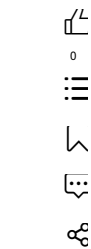
tensorflow存储图和训练的权重为.pb，然后读取.pb并使用 (http://blog.csdn.net/zhuqiqi/article/details/53376283)

keras模型导出到tensorflow，以及pb模型导出 (http://blog.csdn.net/starry_skyWWW/article/details/74209359)



博主专栏

- Android学习 (http://blog.csdn.net/column/details/Android.html) 129844
- AndroidStudio插件开发 (http://blog.csdn.net/column/details/AndroidStudio.html) 26093
- OpenGL (http://blog.csdn.net/column/details/OpenGL.html) 37303
- 在线课程



```
1 import tensorflow as tf
2 with tf.Session() as sess:
3     with open('./checkpoint_dir/graph.pb', 'rb') as graph:
4         graph_def = tf.GraphDef()
5         graph_def.ParseFromString(graph.read())
6         output = tf.import_graph_def(graph_def, return_elements=['out:0'])
7         print(sess.run(output))
```

运行结果如下：

| | |
|---|---------|
| 1 | [100.0] |
|---|---------|

回到 tf.graph_util.convert_variables_to_constants() 函数，可以看到，需要传入Session对象和图，这都可以理解。看看第三个参数 ["out"]，它是指定这个模型的输出 Tensor。

1.2 有代码和模型，但是不想重新训练模型

有模型源码时，在导出模型时就可以通过 tf.graph_util.convert_variables_to_constants() 函数来将变量转为常量保存到图文件中。但是很多时候，我们拿到的是别人的checkpoint文件，即meta、index、data等文件。这种情况下，需要将data文件里面变量转为常量保存到meta文件中。思路也很简单，先将checkpoint文件加载，再重新保存一次即可。

假设训练和保存模型代码如下：

```
1 import tensorflow as tf
2
3 w1 = tf.Variable(20.0, name="w1")
4 w2 = tf.Variable(30.0, name="w2")
5 b1= tf.Variable(2.0,name="bias")
6 w3 = tf.add(w1,w2)
7
8 #记住要定义name，后面需要用到
9 out = tf.multiply(w3,b1,name="out")
10
11 # 转换Variable为constant，并将网络写入到文件
12 with tf.Session() as sess:
13     sess.run(tf.global_variables_initializer())
14     saver = tf.train.Saver()
15     # 这里需要填入输出tensor的名字
16     saver.save(sess, './checkpoint_dir/MyModel', global_step=1000)
```

此时，模型文件如下：

| | |
|---|----------------------------------|
| 1 | checkpoint |
| 2 | MyModel-1000.data-00000-of-00001 |
| 3 | MyModel-1000.index |
| 4 | MyModel-1000.meta |

如果我们只有以上4个模型文件，但是可以看到训练源码。那么，将这4个文件导出为一个pb文件方法如下：

```
1 import tensorflow as tf
2 with tf.Session() as sess:
3
4     #初始化变量
5     sess.run(tf.global_variables_initializer())
6
7     #获取最新的checkpoint，其实就是解析了checkpoint文件
8     latest_ckpt = tf.train.latest_checkpoint("./checkpoint_dir")
9
10    #加载图
11    restore_saver = tf.train.import_meta_graph('./checkpoint_dir/MyModel-1000.meta')
12
13    #恢复图，即将weights等参数加入图对应位置中
14    restore_saver.restore(sess, latest_ckpt)
15
16    #将图中的变量转为常量
17    output_graph_def = tf.graph_util.convert_variables_to_constants(
18        sess, sess.graph_def, ["out"])
19    #将新的图保存到"/pretrained/graph.pb"文件中
20    tf.train.write_graph(output_graph_def, 'pretrained', "graph.pb", as_text=False)
```

执行后，会有如下日志：

| | |
|---|-------------------------------------|
| 1 | Converted 3 variables to const ops. |
|---|-------------------------------------|

接下来就是使用，使用方法跟前面一致：

http://edu.csdn.net/course/6134?utm_source=blog9

http://edu.csdn.net/course/73?utm_source=blog9

他的热门文章

- CoordinatorLayout的使用如此简单 (http://blog.csdn.net/huachao1001/article/details/51554608) 39499
- 自定义View，有这一篇就够了 (http://blog.csdn.net/huachao1001/article/details/51577291) 34521
- 酷炫的Activity切换动画，打造更好的用户体验 (http://blog.csdn.net/huachao1001/article/details/51659963) 25260
- 玩转AppBarLayout，更酷炫的顶部栏 (http://blog.csdn.net/huachao1001/article/details/51558835) 22250
- 从Android代码中来记忆23种设计模式 (http://blog.csdn.net/huachao1001/article/details/51536074) 17502



```
1 import tensorflow as tf
2 with tf.Session() as sess:
3     with open('./pretrained/graph.pb', 'rb') as graph:
4         graph_def = tf.GraphDef()
5         graph_def.ParseFromString(graph.read())
6         output = tf.import_graph_def(graph_def, return_elements=['out:0'])
7         print(sess.run(output))
```

打印信息如下：

| | |
|---|---------|
| 1 | [100.0] |
|---|---------|

2 模型接口设置

我们注意到，前面只是简单的获取一个输出接口，但是很明显，我们使用的时候，不可能只有一个输出，还需要有输入，接下来我们看看，如何设置输入和输出。同样我们分为有代码并且从头开始训练，和有代码和模型，但是不想重新训练模型两种情况。

2.1 有代码并且从头开始训练

相比1.1中的代码略作修改即可，第6行代码处做了修改：

```
1 import tensorflow as tf
2
3 w1 = tf.Variable(20.0, name="w1")
4 w2 = tf.Variable(30.0, name="w2")
5
6 #这里将b1改为placeholder，让用户输入，而不是写死
7 #b1= tf.Variable(2.0,name="bias")
8 b1= tf.placeholder(tf.float32, name='bias')
9
10 w3 = tf.add(w1,w2)
11
12 #记住要定义name，后面需要用到
13 out = tf.multiply(w3,b1,name="out")
14
15 # 转换Variable为constant，并将网络写入到文件
16 with tf.Session() as sess:
17     sess.run(tf.global_variables_initializer())
18     # 这里需要填入输出tensor的名字
19     graph = tf.graph_util.convert_variables_to_constants(sess, sess.graph_def, ["out"])
20     tf.train.write_graph(graph, '.', './checkpoint_dir/graph.pb', as_text=False)
```

日志如下：

| | |
|---|-------------------------------------|
| 1 | Converted 2 variables to const ops. |
|---|-------------------------------------|

接下来看看如何使用：

```
1 import tensorflow as tf
2 with tf.Session() as sess:
3     with open('./checkpoint_dir/graph.pb', 'rb') as f:
4         graph_def = tf.GraphDef()
5         graph_def.ParseFromString(f.read())
6         output = tf.import_graph_def(graph_def, input_map={'bias:0':4}, return_elements=['out:0'])
7         print(sess.run(output))
```

打印信息如下：

| | |
|---|---------|
| 1 | [200.0] |
|---|---------|

也就是说，在设置输入时，首先将需要输入的数据作为placeholder，然后在导入图 tf.import_graph_def() 时，通过参数 input_map={} 来指定输入。输出通过 return_elements=[] 直接引用tensor的name即可。

2.2 有代码和模型，但是不想重新训练模型

在有代码和模型，但是不想重新训练模型情况下，意味着我们不能直接修改导出模型的代码。但是我们可以通过 graph.get_tensor_by_name() 函数取得图中的某些中间结果，然后再加入一些逻辑。其实这种情况在上一篇文章已经讲了。可以参考上一篇文件解决，相比“有代码并且从头开始训练”情况局限比较大，大部分情况只能是获取模型的一些中间结果，但是也满足我们大多数情况使用了。

版权声明：本文为博主原创文章，未经博主允许不得转载。



相关文章推荐

tensorflow 保存和加载模型 -2 (http://blog.csdn.net/lujiandong1/article/details/53301994)

1、 我们经常在训练完一个模型之后希望保存训练的结果，这些结果指的是模型的参数，以便下次迭代的训练或者用作测试。Tensorflow针对这一需求提供了Saver类。 Saver类提供了...

lujiandong1 (http://blog.csdn.net/lujiandong1) 2016年11月23日 10:57 6850

如何用Tensorflow训练模型成pb文件和和如何加载已经训练好的模型文件 (http://blog.csdn.n...

这篇薄荷主要是讲了如何用tensorflow去训练好一个模型，然后生成相应的pb文件。最后会将如何重新加载这个pb文件。 首先先放出PO主的github： https://github.com/...

u014432647 (http://blog.csdn.net/u014432647) 2017年07月18日 10:28 5431



就刚刚，Python圈发生一件大事！

都说人生苦短，要学Python！但刚刚Python圈发生的这件事，你们怎么看？真相在这里...

(http://www.baidu.com/cb.php?c=IgF_pyfqnHmknjnPjc0lZ0qnfK9ujYzP1f4PjDs0Aw-5Hc3rHnYnHb0TAq15HfLPWRznjb0T1Ydn1RkPhuhn1mduHf1nhNh0AwY5HDdnHcdnHTvrjT0lgF_5y9YlZ0lQzq-uZR8mLPbUB48ugfElAqspynETZ-YpAq8nWqdlAdxTvqdThP-5yF_UvTkn0KzujYk0AFV5H00TZcq0KdpyfqnHRLPjnvnfKEpyfqnHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqn1c1n6)

tensorflow存储图和训练的权重为.pb，然后读取.pb并使用 (http://blog.csdn.net/zhuiquik/ar...

(1) tensorflow存储图和训练的权重 from __future__ import absolute_import, unicode_literals import input_dat...

zhuiquik (http://blog.csdn.net/zhuiquik) 2016年11月28日 13:24 8100

keras模型导出到tensorflow，以及pb模型导出 (http://blog.csdn.net/starry_skyWW/article/...

1 keras模型导出成tf模型 https://github.com/amir-abdi/keras_to_tensorflow/blob/master/keras_to_tensorflow...

starry_skyWW (http://blog.csdn.net/starry_skyWW) 2017年07月03日 23:38 2140

tensorflow保存graph和参数为pb文件 (http://blog.csdn.net/linmingan/article/details/72676...

from tensorflow.python.framework.graph_util import convert_variables_to_constants graph = convert_v...

linmingan (http://blog.csdn.net/linmingan) 2017年05月24日 14:28 2322



一学就会的 WordPress 实战课

学习完本课程可以掌握基本的 WordPress 的开发能力，后续可以根据需要开发适合自己的主题、插件，打造最个性的 WordPress 站点。

(http://www.baidu.com/cb.php?c=IgF_pyfqnHmknjfvP1m0lZ0qnfK9ujYzP1f4Pjnz0Aw-5Hc4nj6vPjm0TAq15Hf4rjn1n1b0T1dhuHnkujn3PvNhnW9bnhms0AwY5HDdnHcdnHTvrjT0lgF_5y9YlZ0lQzqMpgwBUvqQhP8QvIGIAPCmgfEmvq_lyd8Q1N9nHmvnj7hnHPWnjFhPAD1Pyn-5HDknWw9mhkEusKzujYk0AFV5H00TZcq0KdpyfqnHRLPjnvnfKEpyfqnHnsnj0YnskWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqP1b4Ps)

将TensorFlow的网络导出为单个文件 (http://blog.csdn.net/EncodeTS/article/details/54428...

有时候，我们需要将TensorFlow的模型导出为单个文件（同时包含模型架构定义与权重），方便在其他地方使用（如在c++中部署网络）。利用'tf.train.write_graph()'默认情况下只导...

EncodeTS (http://blog.csdn.net/EncodeTS) 2017年01月14日 20:03 2929

TensorFlow模型保存和提取方法 (http://blog.csdn.net/marsjhao/article/details/72829635)

一、 TensorFlow模型保存和提取方法 1. TensorFlow通过tf.train.Saver类实现神经网络模型的保存和提取。tf.train.Saver对象saver的save方法将Ten...

marsjhao (http://blog.csdn.net/marsjhao) 2017年06月01日 11:25 8753





Delphi7高级应用开发随书源码 (http://download.csdn.net/detail/chenxh/3)

http://download... 2003年04月30日 00:00 676KB



【Tensorflow】图和模型的保存机制与原理 (http://blog.csdn.net/zt312/article/details/7285...

核心定义：tf.train.Saver类保存 三个文件：模型文件列表checkpoint，计算图结构model.ckpt.meta，每个变量的取值model.ckpt 通过tf.train.import_...

 ztf312 (http://blog.csdn.net/ztf312) 2017年06月03日 20:04  2140


tensorflow从0开始（6）——保存加载模型 (http://blog.csdn.net/searobbers_duck/article/d...

目的 学习tensorflow的目的是能够训练的模型，并且利用已经训练好的模型对新数据进行预测。下文就是一个简单的保存模型加载模型的过程。保存模型 import tenso...

 searobbers_duck (http://blog.csdn.net/searobbers_duck) 2016年06月20日 16:52  50032



tensorflow 中导出/恢复模型Graph数据Saver (http://blog.csdn.net/tengxing007/article/det...

不得不说，在tensorflow中，这个问题一直困扰我好几天了，没有弄清graph个saver的关系。下面我就记录一下两者的用法以及应用场景：Graph图是tensorflow的核心，所有的操作...

 tengxing007 (http://blog.csdn.net/tengxing007) 2017年02月23日 15:09  1009



Tensorflow保存模型，恢复模型，使用训练好的模型进行预测和提取中间输出（特征） (http://...

前言：tensorflow中有operation和tensor，前者表示 操作，后者表示 容器，每个operation都是有一个tensor来存放值的，比如y=f(x)，operation是f...

 ying86615791 (http://blog.csdn.net/ying86615791) 2017年05月25日 17:22  4463



TensorFlow学习笔记（8）--网络模型的保存和读取 (http://blog.csdn.net/lwplwf/article/deta...

之前的笔记里实现了softmax回归分类、简单的含有一个隐层的神经网络、卷积神经网络等等，但是这些代码在训练完成之后就直接退出了，并没有将训练得到的模型保存下来方便下次直接使用。为了让训练结果可以复用...

 lwplwf (http://blog.csdn.net/lwplwf) 2017年03月16日 11:23  19935



inception V4 与resnet (http://blog.csdn.net/keyanxiaocaicai/article/details/66478367)

转载自：http://blog.csdn.net/lynnandwei/article/details/53736235

 keyanxiaocaicai (http://blog.csdn.net/keyanxiaocaicai) 2017年03月26日 19:49  756



TensorFlow学习笔记（11）--【Ubuntu】slim框架下的inception_v4模型的运行、可视化、...

模型：slim框架下的Inception_v4模型 Inception_v4的Checkpoint：http://download.tensorflow.org/models/inception_v...

 lwplwf (http://blog.csdn.net/lwplwf) 2017年07月27日 11:10  7241



CNN浅析和历年ImageNet冠军模型解析 (http://blog.csdn.net/AMDS123/article/details/7023...

版权声明 作者：黄文坚 本文为大数据杂谈4月13日微信社群分享内容整理。今天在这里我给大家讲解一些深度学习中卷积神经网络的原理和一些经典的网络结构 卷积神经网络原理浅析 ...

 AMDS123 (http://blog.csdn.net/AMDS123) 2017年04月18日 21:00  11184

Tensorflow 使用slim框架下的分类模型进行分类 (http://blog.csdn.net/cugxyy6/article/detai...

Tensorflow的slim框架可以写出像keras一样简单的代码来实现网络结构（虽然现在keras也已经集成在tf.contrib中了），而且models/slim提供了类似之前说过的object...

 cugxyy6 (http://blog.csdn.net/cugxyy6) 2017年10月27日 20:44  117


TensorFlow —— 30秒搞定物体检测 (http://blog.csdn.net/wangli0519/article/details/73611...

Google发布了新的TensorFlow物体检测API，包含了预训练模型，一个发布模型的jupyter notebook，一些可用于使用自己数据集对模型进行重新训练的有用脚本。使用该API可以快速...

 wangli0519 (http://blog.csdn.net/wangli0519) 2017年06月22日 22:04  5717

[Tensorflow]基于slim框架下inception模型的植物识别 (http://blog.csdn.net/wizard/article/d...

1.数据获取 python脚本根据关键字爬取对应的图片 #!/usr/bin/env python # encoding: utf-8 import urllib2 import re...

 wizard (http://blog.csdn.net/wizard) 2017年08月29日 22:46  1067

python3 下 tensorflow slim inceptionV4 问题修正与测试 (http://blog.csdn.net/sparkexper...

自从残差网络出来之后，好多人就想着能不能对模型进行结合测试一下。Google Research的Inception模型和Microsoft Research的ResidualNet模型两大图像识别...

 sparkexpert (http://blog.csdn.net/sparkexpert) 2017年04月18日 16:19  1259