**Win10系统下Anaconda环境下 TensorFlow的安装**

汪雄 2018/4/9

一．第一步Anaconda的安装和python环境的配置

1. Anaconda的安装

直接去百度上搜索，直接点击安装

1. 在Anaconda下安装多个版本的python环境

进入Anaconda promote的命令行模式

在命令行中输入

conda --version

可查看当前安装的Anaconda的版本

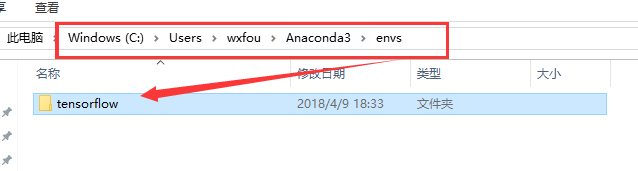
1. 创建一个3.5.2版本的python环境

在promote命令行中输入

 conda create –name tensorflow python=3.5.2

解释：该命令在envs环境下新建了一个tensorflow的文件夹，里面安装的是python3.5.2的安装环境

就会在安装目录下生成一个 tensorflow的文件夹，里面是python3.5.2的环境，如下图所示：



不同版本的切换

在promote命令行中输入 activate tensorflow

将envs目录下的tensorflow文件夹下的python环境激活

退出该环境，在promote环境中输入 deactivate tensorflow

二．TensorFlow环境的安装

**在python3.5环境中安装 CPU 版本的tensorflow**

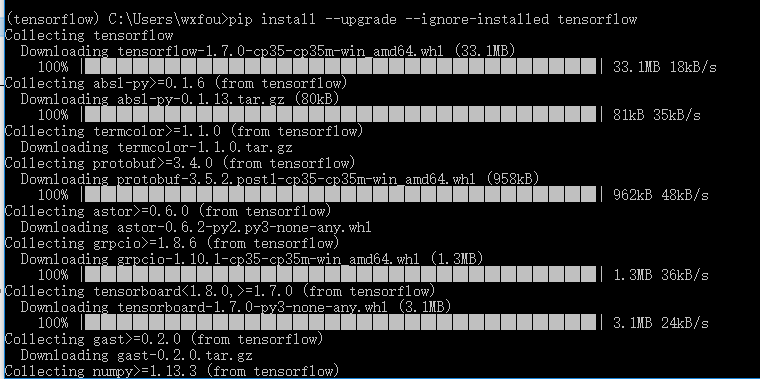
命令行参数为：

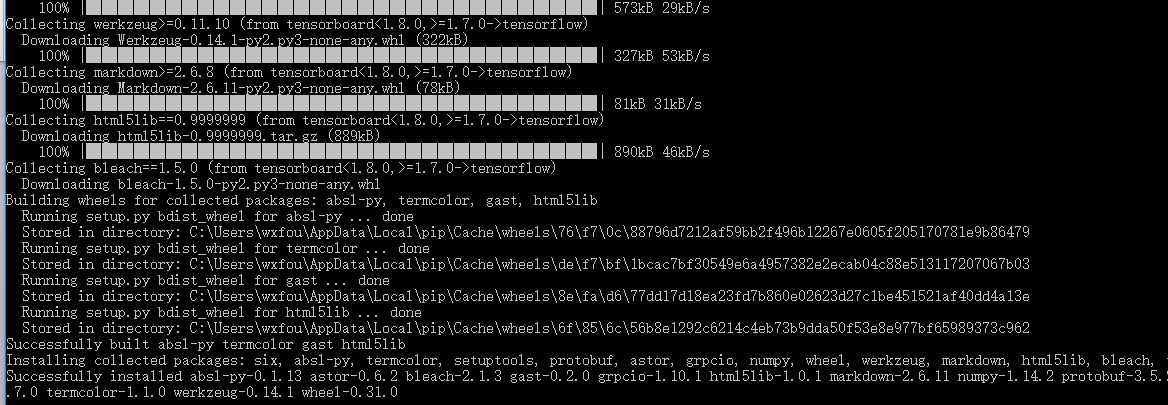
pip install --upgrade --ignore-installed tensorflow

查看tensorflow的版本信息

pip show tensorflow

安装成功后的截如下：





Tensorflow的卸载：

命令行中：

pip uninstall tensorflow

安装GPU版本的tensorflow

命令行安装：

pip install tensorflow-gpu

Tensorflow学习笔记

参考网友：<https://www.cnblogs.com/shihuc/p/6648130.html>

1. Tensorflow基本架构
2. 使用图（Graph）来表示计算任务
3. 在被称之为回话（Session）的上下文（Context）中执行图
4. 使用tensor表示数据
5. 通过变量（Variable）维护状态
6. 使用feed和fetch可以为任意的操作（arbitray operation）赋值或者从其中获取数据
7. Tensor

Tensorflow是一个编程系统，使用图来表示计算任务。图中的节点被称之为op。（Operation）。一个op获得0个或多个Tensor，执行计算，产生0个或多个Tensor。

每个Tensor是一个类型化的多维数组。例如，你可以将一小组图像集表示为一个四维浮点数组，这个四维数组分别是[batch，height，width，channels]。

一个Tensorflow图描述了计算的过程。为了进行并行计算，图必须在会话里被启动。

会话将图的op分发到诸如CPU或GPU之类的设备上，同时提供执行op的方法。这些方法执行后，将产生的tensor返回，在python语言中，返回的tensor时numpy ndarray对象；在C C++语言中，返回的是tensor是tensorflow::Tensor实例

1. Graph

Tensorflow程序通常被组织成一个构建阶段和一个执行阶段。在构建阶段，op的执

行步骤被描述成一个图，在执行阶段，使用会话执行图中的op。例如，通常在构建阶段创建一个图来表示和训练神经网络，然后在执行阶段反复执行图中的训练op。下面这个图，就是一个比较形象的说明。图中的每一个节点，就是一个op，各个op透过tensor数据流向形成的连接，构成了一个图。



构建图的第一步，是创建源op（source op）。源op不需要任何输入，例如常量（constant）。源op的输出被传递给其它op做运算。Python库中，op构造器的返回值代表被构造出的op的输出，这些返回值可以传递给其它op构造器作为输入。

Tensorflow Python库有一个默认图（default graph），op构造器可以为其增加节点。这个默认图对许多程序员来说已经足够用了。

四、Session

当图构建后了后，需要创建一个Session来运行构建好点的图，来实现逻辑，创建session的时候，若无任何参数，Tensorflow将启动默认的session。Session.run(xxx)是比较典型的使用方案，session运行结束后，返回时一个tensor。

Tensorflow的session有两大类，一种就是普通的session，即tensorflow.Session()，还有一种是交互式session，即tensorflow.InteractiveSession()。使用Tensor.eval()和Operation.run()方法替代Session.run()。这样可以避免使用一个变量来持有会话，为程序架构的设计添加了灵活性。

1. 远程访问Jupyter notebook

<https://www.cnblogs.com/wwwhza/p/8821117.html>