1. **概述**

本次实验选用Intel Movidius例程dogsvscats，目的是熟悉caffe的基本操作和DevCloud训练模型的操作过程。Dogsvscats使用GoogLeNet模型，用于识别图片中的猫和狗。本次实验主要分为3步：准备数据，连接DevCloud并传输数据，训练。

1. **准备数据**
2. 本地部署dogsvscats

在~下创建目录workspace：

mkdir -p ~/workspace

从github上clone代码：

cd ~/workspace

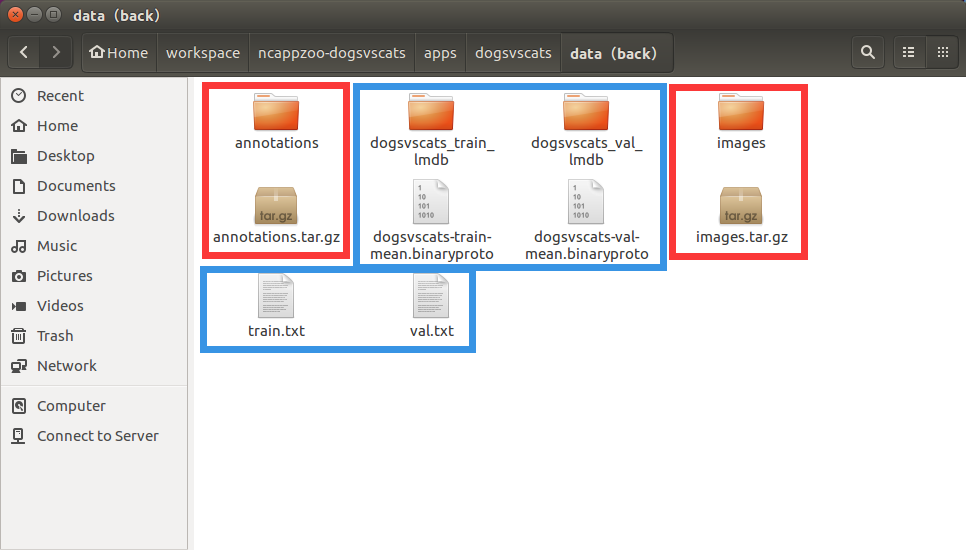
git clone <https://github.com/yuanyuanli85/ncappzoo.git>

cd ncappzoo

git checkout dogsvscats

1. 生成训练数据

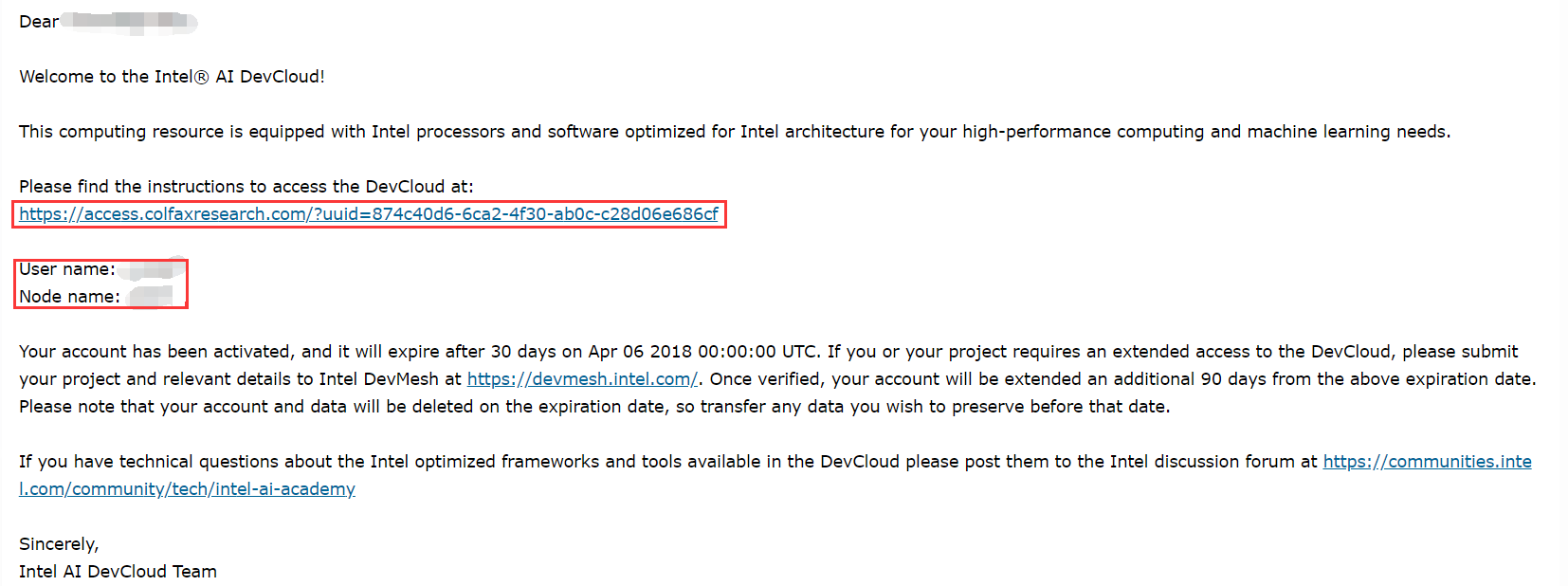
官方给的方法中，生成训练数据需要配置caffe环境，相当麻烦，如果有兴趣的同学可以参考<https://movidius.github.io/blog/deploying-custom-caffe-models/>，没兴趣的同学可以直接从我这儿拷数据。训练数据主要包括：



其中，红框标注的是原始训练数据，images是图片，annotations是标签；蓝框标注的是根据原始训练数据生成的适用于caffe的训练数据，蓝框中的文件夹保存的是lmdb格式的训练数据，两个binaryproto文件是mean均值文件，在训练中会用到；两个txt文件是图片和标签的列表。

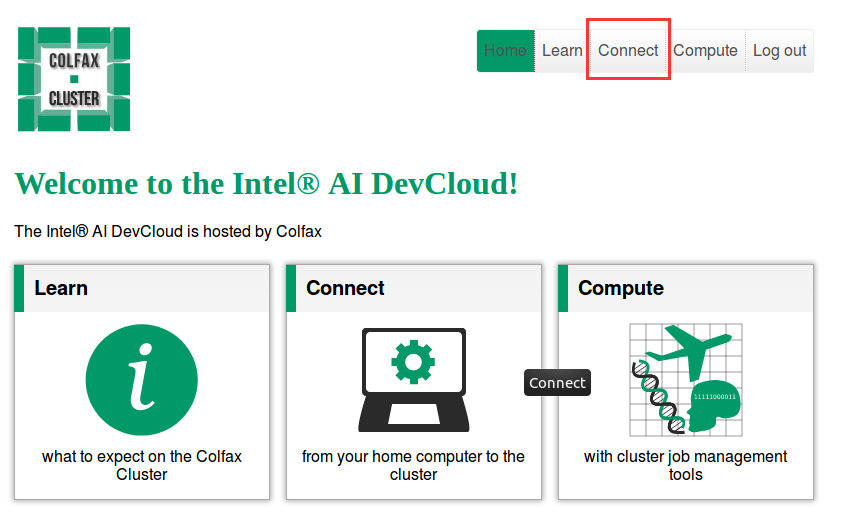
训练数据保存的位置在ncappzoo/apps/dogsvscats/data文件夹中，**与图中路径并不完全相同**。初始状态下是没有这个data文件夹的，需要自己新建。

1. **连接DevCloud**
2. 打开邮箱，找到大概这样一封Intel的邮件：

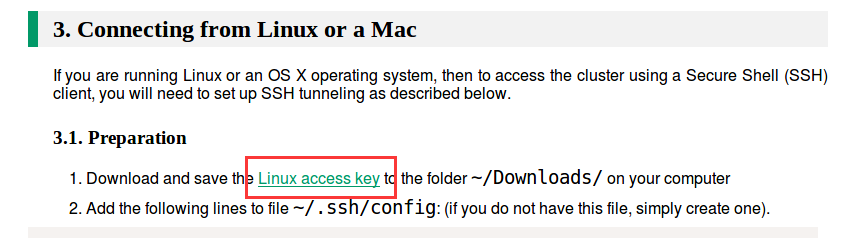


第一个红框内的链接是使用教程，教你怎么连接到devcloud，要仔细阅读下。第二个红框内是你的user name和node name，这个要记下来。

1. 从本地linux连接到DevCloud
2. 打开刚刚那封邮件中的链接（我用红框标注的），点击connect：



向下找到Connecting from Linux or a Mac，点击那部分内容中的Linux access key，下载access key，如下图：



1. 安装ssh：

sudo apt-get install ssh

然后进入~路径下，创建.ssh文件夹，用于存放linux access key：

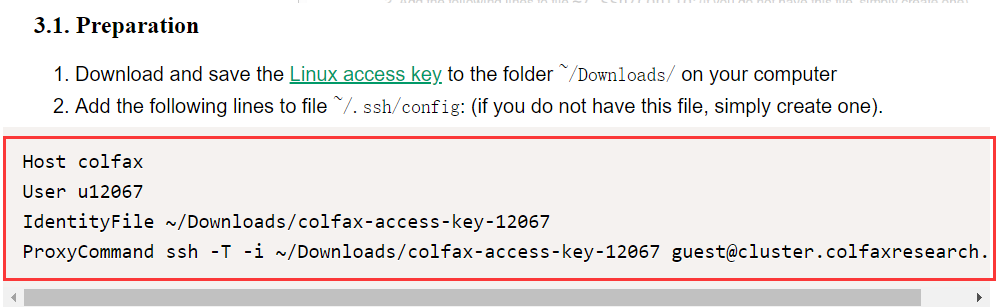
cd ~

mkdir .ssh

cd ~/.ssh/

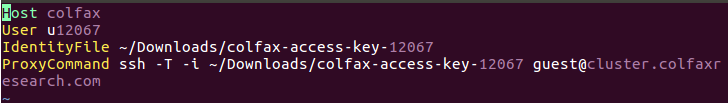
vim config

将网页中的下面这段内容（红框内的）拷贝到config文件中：



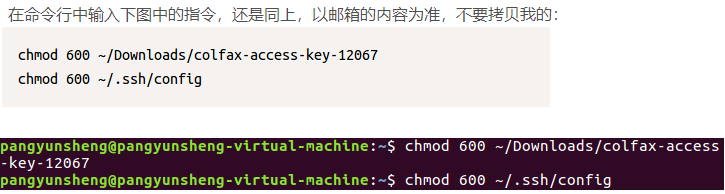
这一项涉及到vim的操作，不熟悉的同学可以自己百度一下。

注：不要照着我的抄上去，因为每个人的用户信息不同，下载的密钥也不同，根据你邮箱中的内容直接拷贝过去就行了。还是建议大家阅读一下这个教程，写的很详细，我这里也仅仅是将其翻译一遍。



1. 修改系统文件

在命令行中输入下图中的指令，还是同上，以邮箱的内容为准，不要拷贝我的：

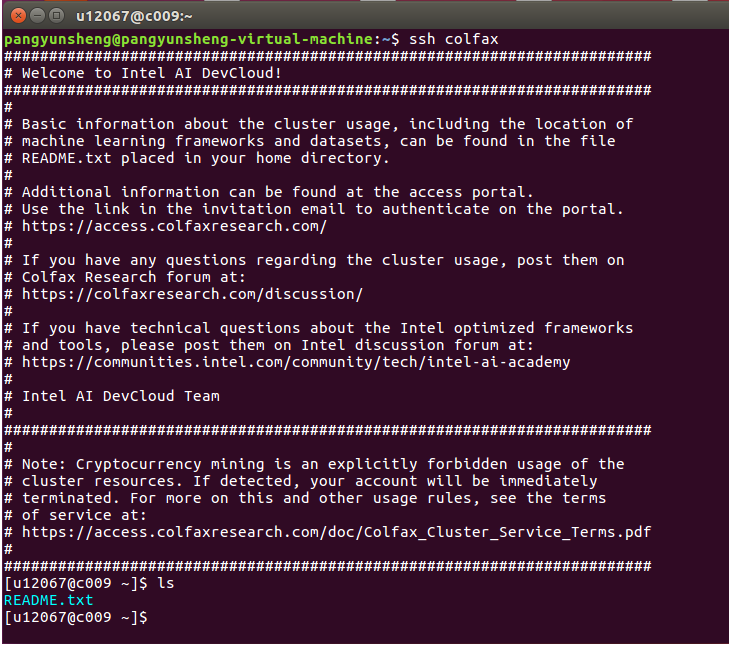


1. 登陆DevCloud

在命令行中输入下面指令（该指令就是登陆到DevCloud的指令，以后登陆都用这个指令）：

ssh colfax

显示如下：



说明我们已经成功登陆到了DevCloud。

1. 上传文件

现在我们已经登陆到了DevCloud，在这里我们可以简单地看做我们登陆到了另一台linux主机，然而这台主机的计算能力要远远大于我们的笔记本电脑。现在虽然登陆到了这台主机，但是主机内是空的，没有任何文件，我们需要把我们本地已经创建好的，需要训练用的文件上传到DevCloud。

1. 与（1）中相同，先clone代码：

在~下创建目录workspace：

mkdir -p ~/workspace

从github上clone代码：

cd ~/workspace

git clone https://github.com/yuanyuanli85/ncappzoo.git

cd ncappzoo

git checkout dogsvscats

1. 将本地的训练数据传输至DevCloud：

再打开一个命令行，在**本地linux下**输入下面的文件传输指令：

scp –r \

/home/reins/workspace/ncappzoo/apps/dogsvscats/data \

colfax:/home/u26506/workspace/ ncappzoo/apps/dogsvscats

其中，/home/reins/workspace/ncappzoo/apps/dogsvscats/data是本地的训练数据所在位置，同学们需要换成自己的data路径；后面的u26506是我使用的用户ID，读者也需要更换自己的用户ID。这个命令的意思是将本地的data文件夹传到DevCloud的路径下，以备训练使用。传输后可以在DevCloud下查看是否已经有了这个文件夹：



1. Clone caffe源码，因为训练时需要用到caffe的blvc\_googlenet：

cd ~

git clone <https://github.com/BVLC/caffe.git>

设置CAFFE\_PATH环境变量：

export CAFFE\_PATH=/home/u26506/caffe

这里的用户名要换成自己的。

本节主要参考<https://blog.csdn.net/pangyunsheng/article/details/79434302>和<https://www.cnblogs.com/WaitingForU/p/9091096.html>。

1. **训练**

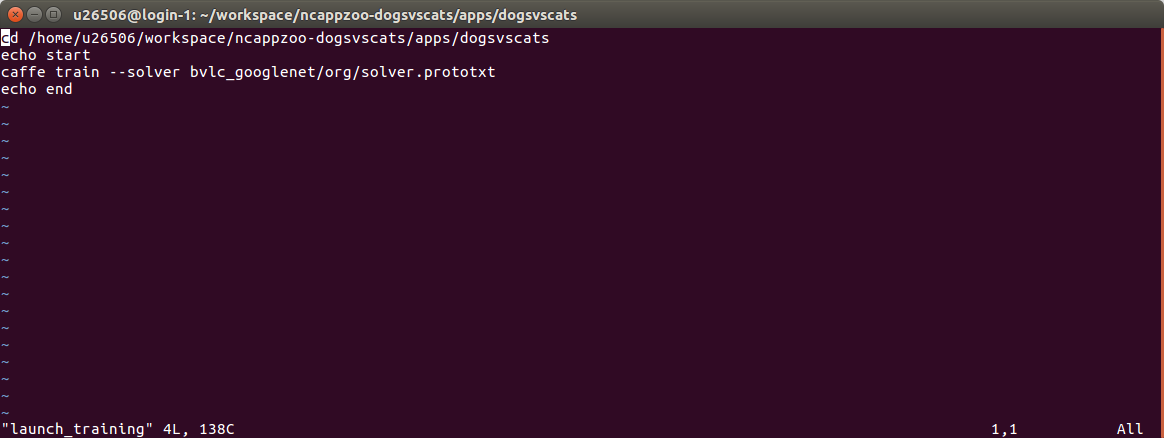
DevCloud的使用方法基本都在DevCloud的README.txt中写明，需要注意的是，登录进去后会发现自己在c009，这其实是login node，在这个状态下不能进行训练，因为CPU，内存，计算能力都受到了限制。如果要进行训练，需要用qsub命令。具体操作如下：

1. 创建脚本文件

在DevCloud的ncappzoo/apps/dogsvscats目录下创建一个脚本文件：

vim lanuch\_training

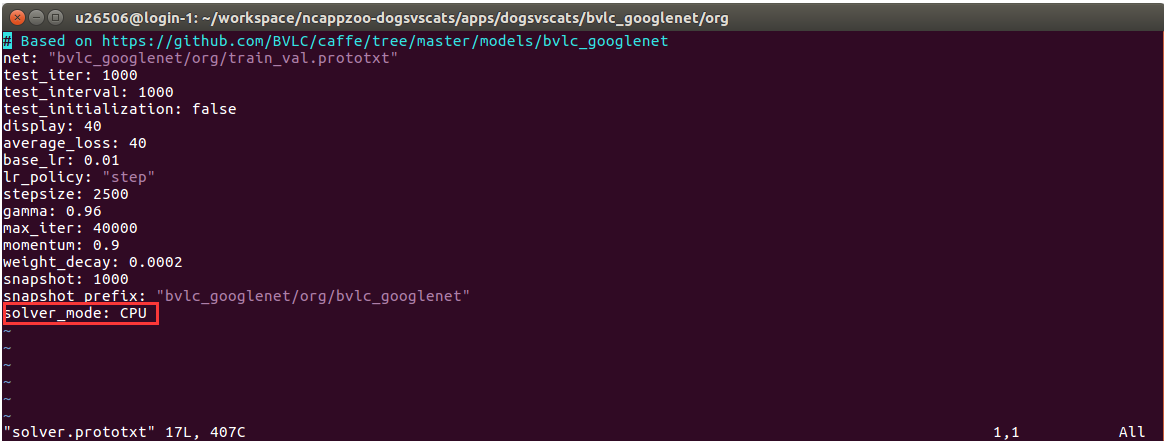
在脚本文件中加入下面内容（路径改为自己的）：



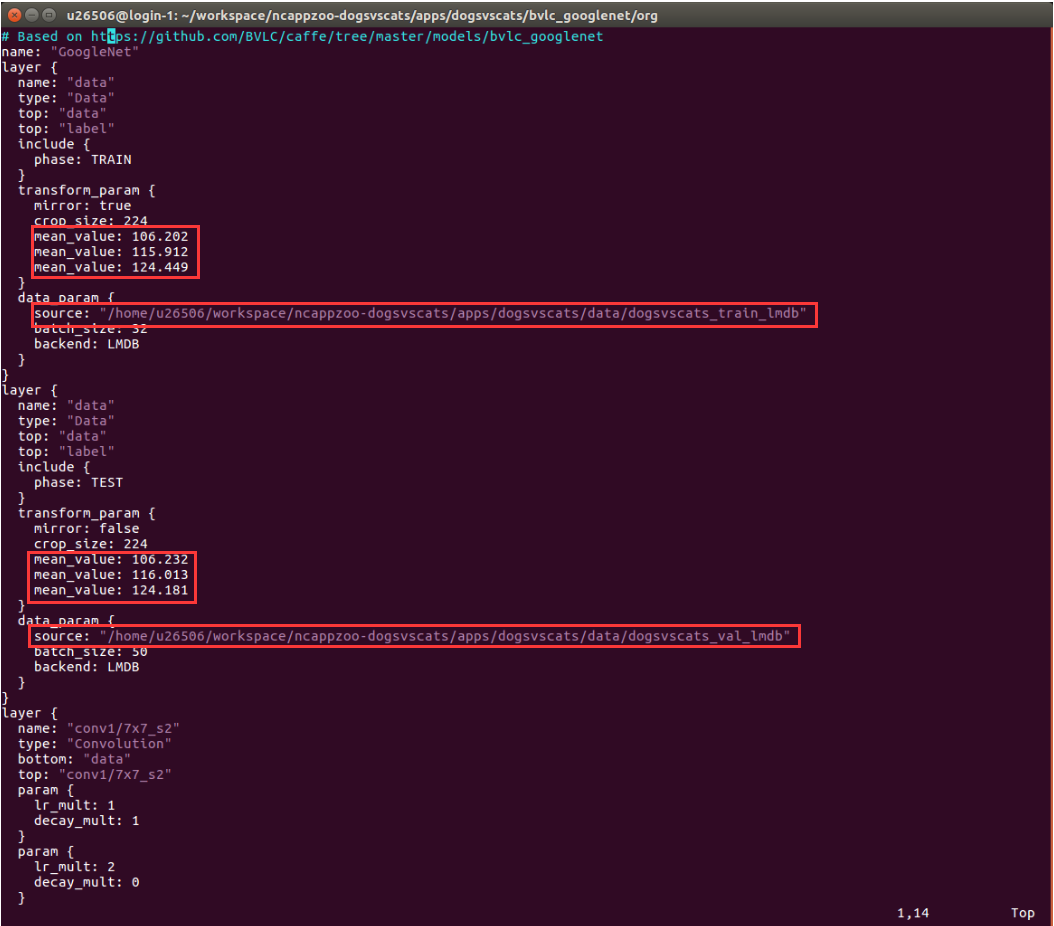
1. 修改网络结构和参数配置文件的部分路径

本次训练的网络结构和参数配置文件在ncappzoo/apps/dogsvscats/bvlc\_googlenet/org路径下，需要修改的是solver.prototxt和train\_val.prototxt文件。

首先看solver.prototxt，要把GPU改成CPU：



再看train\_val.prototxt，需要改mean\_value和路径（改成自己的路径，mean\_value可以改成一样的）：



1. 提交任务，开始训练

输入下面指令，提交任务：

qsub lanuch\_training

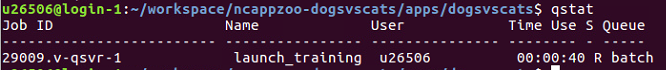
lanuch\_training文件是刚刚写的脚本文件。



到这里我们的任务就已经提交成功了，DevCloud现在就在帮我们训练。

1. 查看任务

在训练过程中，可以使用qstat指令查看任务：



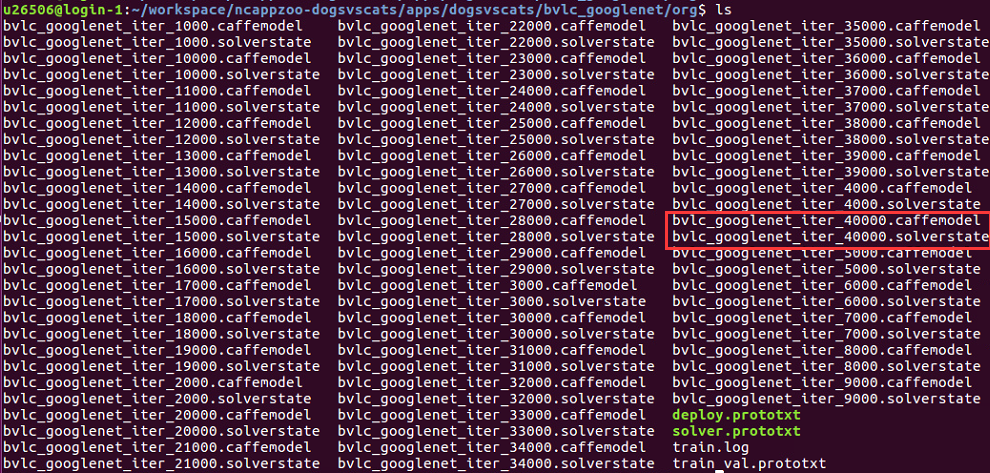
说明训练还没有完成。

如果输入qstat指令后没有任何输出，说明训练已经完成，我们可以查看训练日志以及训练出的模型。训练日志和模型默认保存在ncappzoo/apps/dogsvscats/bvlc\_googlenet/org路径下。

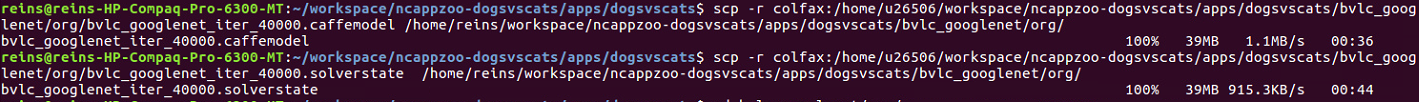
另外，训练有可能出错，如果发现qstat没有输出，而文件夹中出现了launch\_training.oxxxx和launch\_training.exxxx的文件，说明可能出错了，错误信息就保存在这两个文件中，用vim即可查看。

1. 训练完成

训练完成后，文件夹中应该有如下文件：



记录了每一轮训练的结果，其中数字最大的就是最终的结果。训练完成后，用scp命令将caffemodel和solverstate传回本地：



至此。DevCloud的任务便完成了。