# Install Tensorflow 1.8.0 with GPU from source on Ubuntu 18.04 Bionic Beaver 在Ubuntu 18.04仿生海狸上安装Tensorflow 1.8.0和GPU



When I first put Linux, the first thing I wanted to do was install the Tensorflow GPU. I found an excellent guide. But even with this in mind, I spent more than 40 hours on the installation. This article I want to save you time, sharing your own experience. Below you will find updated guide from python36.com  
当我第一次使用Linux时，我想做的第一件事就是安装Tensorflow GPU。我找到了一个很好的向导。但即使考虑到这一点，我在安装上花了40多个小时。这篇文章我想节省你的时间，分享你自己的经验。下面您将从python36.com找到更新的指南

Note: Have updated guide (all important changes are made at step 12). For me, previous version tensorflow and bazel build took 2h 3min. Updated version build 1hrs 32min. It worth it to update as you think?  
注意：已更新指南（所有重要更改都在步骤12中进行）。对我来说，之前的tensorflow和bazel版本花了2小时3分钟，更新后的版本花了1小时32分钟？

### Step 1: Update and Upgrade your system: 步骤1:更新和升级系统：

### Step 2: Verify You Have a CUDA-Capable GPU: 步骤2：验证您是否具有支持CUDA的GPU：

Note GPU model. eg. GeForce 840M  
注意GPU型号。例如，GeForce 840M

If you do not see any settings, update the PCI hardware database that Linux maintains by entering update-pciids (generally found in /sbin) at the command line and rerun the previous lspci command.  
如果没有看到任何设置，请通过在命令行输入update pciid（通常在/sbin中找到）来更新Linux维护的PCI硬件数据库，然后重新运行前面的lspci命令。

### Step 3: Verify You Have a Supported Version of Linux: 第3步：验证是否有受支持的Linux版本：

To determine which distribution and release number you’re running, type the following at the command line:  
要确定运行的发行版和发行版本号，请在命令行中键入以下命令：

The x86\_64 line indicates you are running on a 64-bit system which is supported by cuda 9.1.  
x86\_64行表示您正在cuda 9.1支持的64位系统上运行。

### Step 4: Install Dependencies: 步骤4:安装依赖项：

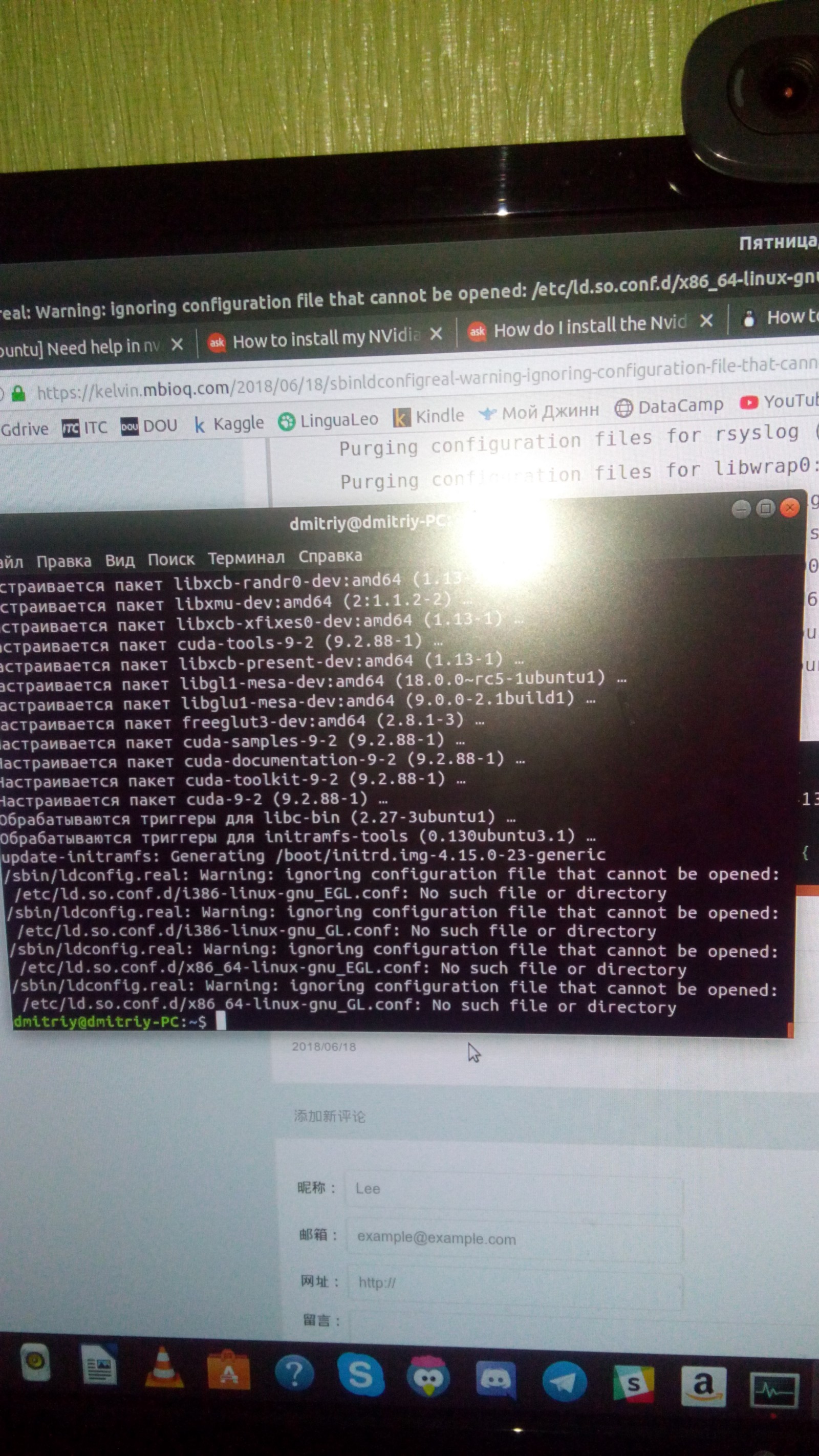
Required to compile from source:  
需要从源代码编译：

### Step 5: Install linux kernel header: 步骤5：安装linux内核头：

Goto terminal and type:  
转到终端和类型：

You can get like “4.15.0–23-generic”. Note down linux kernel version.  
你可以得到“4.15.0–23通用”。记下linux内核版本。

In Bionic Beaver (18.04) you have 4.15 kernel. With this version you have chance to stuck with cuda installation. I searched on Google for many hours and was not solved this problem.  
在仿生海狸（18.04）中有4.15个内核。有了这个版本，你就有机会坚持安装cuda。我在谷歌上搜索了好几个小时，但没有解决这个问题。

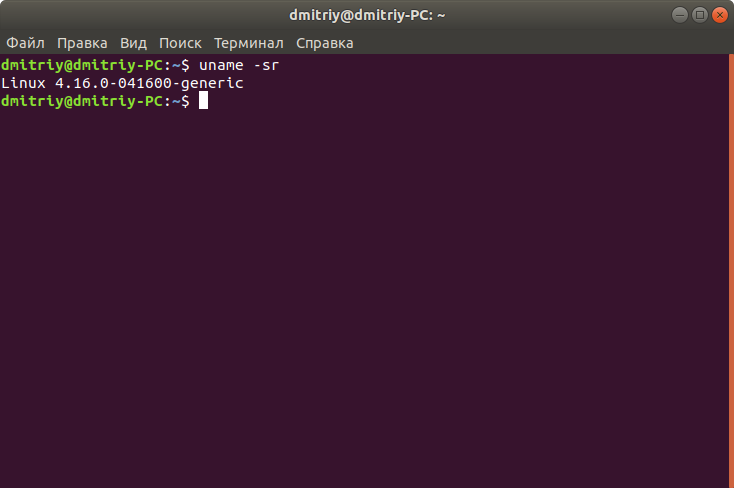


To avoid this problem install 4.16 kernel:  
要避免此问题，请安装4.16内核：

Once you’ve downloaded all the above kernel files, now install them as follows. Linux-headers will also be installed with this command:  
下载完以上所有内核文件后，现在按如下方式安装它们。Linux头文件也将与以下命令一起安装：

Once the installation is complete, reboot your machine and verify that the new kernel version is being used:  
安装完成后，重新启动计算机并验证是否正在使用新的内核版本：

You must get something like this:  
你必须得到这样的东西：



And that’s it. You are now using a much more recent kernel version than the one installed by default with Ubuntu.  
就这样。您现在使用的内核版本比Ubuntu默认安装的版本要新得多。

### Step 6: Install NVIDIA CUDA 9.2: 步骤6：安装NVIDIA CUDA 9.2：

Remove previous cuda installation(if you installed cuda before):  
删除以前的cuda安装（如果以前安装过cuda）：

Install cuda:  
安装cuda：

On step 6 when execute last command be careful!  
在第6步执行最后一个命令时要小心！

At the first try usually script stuck at unitramfs (every time when I launch it on 4.15 kernel I saw what this unitramfs files not found and cuda installed wrong and not correctly). On kernel 4.16 you don`t stuck with this problem.  
在第一次尝试时，脚本通常停留在unitramfs上（每次我在4.15内核上启动它时，我都看到这个unitramfs文件找不到，cuda安装错误且不正确）。在内核4.16中，你不会纠结于这个问题。

If this line not update for few minutes, open System Monitor and wait, when the load of the CPU cores will decrease. Don`t type immediately!  
如果这一行几分钟内没有更新，打开系统监视器并等待，这时CPU内核的负载将减少。不要马上打字！

Then try this in terminal:  
然后在终端中尝试此操作：

use ESC few times and then type: password + Enter + password + Enter..  
使用ESC几次，然后键入：password+Enter+password+Enter。。

If not helped:  
如果没有帮助：

use ESC few times and then type: password + Enter + password + Enter..  
使用ESC几次，然后键入：password+Enter+password+Enter。。

Be patient with yourself, typed password slowly. After 10 try use ESC and type again.  
耐心点，慢慢地输入密码。10分钟后，尝试使用ESC并再次键入。

If you install cuda on a fresh system, you need to type password «blindly » just once. Else be prepared to do this twice: when build kernel and when see this message:  
如果在新系统上安装cuda，则只需输入一次“blindly”密码。否则请准备两次：构建内核时和看到此消息时：

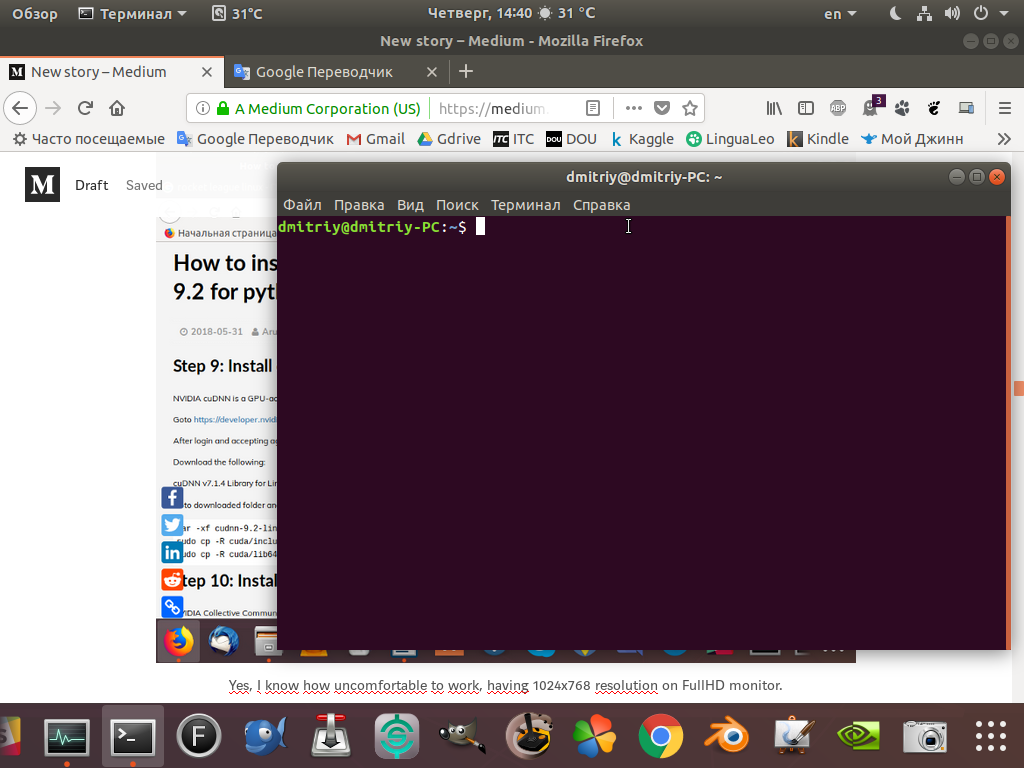
And you will succeed!  
你会成功的！

### Step 7: Reboot the system to load the NVIDIA drivers. 步骤7：重新启动系统以加载NVIDIA驱动程序。

### Step 8: Go to terminal and type: 步骤8：转到终端并键入：

Check driver version probably Driver Version: 396.26  
检查驱动程序版本可能是驱动程序版本：396.26

For now if you use nvidia-smi command you get temp for GPU and nothing more (no process found below). And you have low screen resolution because your nvidia-drivers not detect ligament (GPU-monitor).  
目前，如果您使用nvidia smi命令，您将获得GPU的temp，而没有更多（下面找不到进程）。你有低屏幕分辨率，因为你的nvidia驱动程序没有检测韧带（GPU显示器）。



But you can fix this with Xorg!  
但你可以用Xorg来解决这个问题！

Use this command to create Xorg:  
使用此命令创建Xorg：

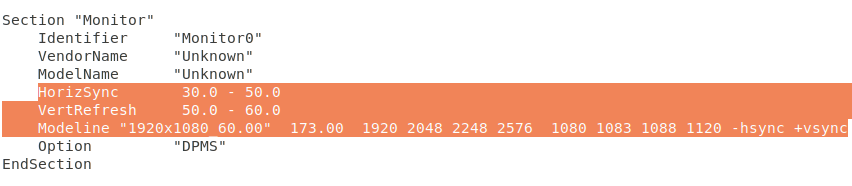
That`s create a file a file xorg.conf in path: (etc/X11/xorg.conf). To change resolution you need just reboot your system. And you can go to step 9.  
在路径（etc/X11/xorg.conf）中创建一个文件xorg.conf。要更改分辨率，只需重新启动系统。你可以进入第9步。

If this not helped change this file (xorg.conf). To do this use this command with parameters of your monitor. My command look like this:  
如果这没有帮助更改这个文件（xorg.conf）。要执行此操作，请将此命令与监视器的参数一起使用。我的命令是这样的：

Press Enter and you got  
按回车键

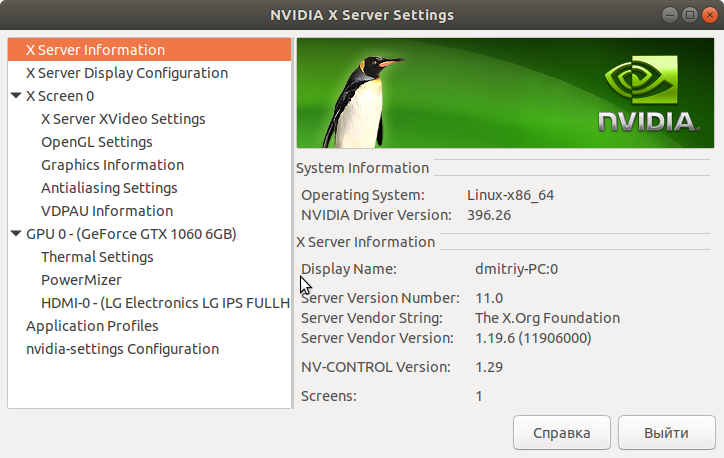
Just copy this and open xorg.conf:  
复制此文件并打开xorg.conf：

Paste here (instead of modeline). Change HorizSync and VertRefresh too:  
粘贴在此处（而不是modeline）。更改水平同步和垂直刷新：



And reboot after it. For now your screen resolution should be the same as before. For now you can type  
然后重启。现在你的屏幕分辨率应该和以前一样。现在你可以打字了

And see this:  
看看这个：



Now you can see temperature and other useful information about your GPU.  
现在你可以看到你的GPU的温度和其他有用的信息。

### Step 9: Install cuDNN 7.1.4: 步骤9：安装cuDNN 7.1.4：

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.  
NVIDIA cuDNN是一个GPU加速的深层神经网络原语库。

Goto and download Login and agreement required  
转到并下载所需的登录和协议

After login and accepting agreement.  
登录并接受协议后。

Download the following:  
下载以下内容：

cuDNN v7.1.4 Library for Linux  
适用于Linux的cuDNN v7.1.4库

Goto downloaded folder and in terminal perform following:  
转到下载的文件夹并在终端中执行以下操作：

### Step 10: Install NCCL 2.2.13: 步骤10：安装NCCL 2.2.13：

NVIDIA Collective Communications Library (NCCL) implements multi-GPU and multi-node collective communication primitives that are performance optimized for NVIDIA GPUs  
NVIDIA Collective Communications Library（NCCL）实现了为NVIDIA GPU优化性能的多GPU和多节点集合通信原语

Go to and attend survey to download Nvidia NCCL.  
前往并参加调查以下载Nvidia NCCL。

Download following after completing survey.  
完成调查后下载以下内容。

Go to downloaded folder and in terminal perform following:  
转到下载的文件夹并在终端中执行以下操作：

### Step 11: Install Dependencies 步骤11：安装依赖项

Isntall libcupti:  
Isntall libcupti公司：

Python related:  
与Python相关：

To install these packages for Python 2.7, issue the following command:  
要为Python2.7安装这些软件包，请发出以下命令：

To install these packages for Python 3.n, issue the following command:  
要为Python3.n安装这些软件包，请发出以下命令：

### Step 12: Configure Tensorflow from source: 步骤12：从源配置Tensorflow：

Download bazel(version 0.14):  
下载bazel（0.14版）：

Reload environment variables  
重新加载环境变量

Start the process of building TensorFlow by downloading latest tensorflow 1.8.0.  
通过下载最新的TensorFlow 1.8.0开始构建TensorFlow的过程。

Give python path in  
提供python路径

Press enter two times  
按回车键两次

Now you need compute capability which we have noted at step 1 eg. 5.0. Go to that and click on “CUDA-Enabled GeForce Products”. For example: if you have GPU on Pascal architecture your`s compute capability should be 6.1, if Maxwell — 5.2 and so on.  
现在你需要计算能力，我们已经在步骤1例如5.0中提到。点击“启用CUDA的GeForce产品”。例如：如果你在Pascal架构上有GPU，那么你的计算能力应该是6.1，如果Maxwell - 5.2等等。

Configuration finished!  
配置完成！

### Step 13: Build Tensorflow using bazel 步骤13：使用bazel构建Tensorflow

The next step in the process to install tensorflow GPU version will be to build tensorflow using bazel. This process takes a fairly long time.  
安装tensorflow GPU版本的下一步是使用bazel构建tensorflow。这个过程需要相当长的时间。

To build a pip package for TensorFlow you would typically invoke the following command:  
要为TensorFlow构建pip包，通常需要调用以下命令：

Note:if you got error like unsupported platform then make sure you are running correct pip command associated with the python you used while configuring tensorflow build.  
注意：如果出现不支持平台之类的错误，请确保运行的pip命令与配置tensorflow构建时使用的python相关。

This process will take a lot of time. It may take 1–2 hours or maybe even more.For example, on my i5–4590 it take 2 hrs 3 min. Be ready to wait!  
这个过程需要很多时间。可能需要1-2小时甚至更长时间。例如，在我的i5-4590上需要2小时3分钟。准备好等待！

Also if you got error like Segmentation Fault then try again it usually worked.  
另外，如果你有错误，如分割错误，然后再试一次，它通常是有效的。

The bazel build command builds a script named build\_pip\_package. Running this script as follows will build a .whl file within the tensorflow\_pkg directory:  
bazel build命令生成一个名为build\_pip\_package的脚本。按如下方式运行此脚本将在tensorflow\_pkg目录中生成一个.whl文件：

To build whl file issue following command:  
要生成whl文件，请发出以下命令：

To install tensorflow with pip:  
要使用pip安装tensorflow：

for existing virtual environment:  
对于现有的虚拟环境：

With a new virtual environment using virtualenv:  
使用virtualenv的新虚拟环境：

for python 2: (use sudo if required)  
对于python 2：（如果需要，请使用sudo）

for python 3: (use sudo if required)  
对于python 3:（如果需要，请使用sudo）

Note : if you got error like unsupported platform then make sure you are running correct pip command associated with the python you used while configuring tensorflow build.  
注意：如果出现不支持平台之类的错误，请确保运行的pip命令与配置tensorflow构建时使用的python相关。

You can check pip version and associated python by following command

### Step 14: Verify Tensorflow installation

Run in terminal

If the system outputs the following, then you are ready to begin writing tensorflow programs:

Success! You have now successfully installed tensorflow 1.8.0 on your machine.



This article based on awesome guide, which available [here](http://www.python36.com/how-to-install-tensorflow-gpu-with-cuda-9-2-for-python-on-ubuntu/). I am only supplemented it with the errors I encountered and how to avoid them.

Those answers maybe can help you during the process:

<https://www.tecmint.com/upgrade-kernel-in-ubuntu/>

<https://devtalk.nvidia.com/default/topic/1036167/stuck-trying-to-intall-nvidia-390-ubuntu-18-04-lts-/?offset=10>

<https://askubuntu.com/questions/647708/cannot-edit-xorg-conf-permissions>

<https://askubuntu.com/questions/4253/getting-screen-resolution-correct-with-nvidia-drivers>

<https://unix.stackexchange.com/questions/387735/how-to-set-a-custom-resolution-with-nvidia-drivers-installed>

Thanks for your attention!