Harish Masand Follow
Aug 17, 2021 · 3 min read · Listen

Installing Tensorflow with GPU, CUDA and cuDNN in Ubuntu 20.04

Deep Learning In-Depth — Get Started

This article covers the installation of GPU, CUDA, cuDNN and Tensorflow in Ubuntu 20.04. I have demonstrated below for Nvidia GeForce GTX1060m (m-Notebook) Graphic card, however you can follow below steps for any CUDA compatible GPU.

Most critical point is to install correct version of CUDA and cuDNN based on Graphic card driver installed in Step 1 below.

Prerequisite

- Ubuntu 20.04
- Nvidia GeForce GTX 1060m

Article Demonstrates installation for GTX1060m, however you can follow notes below each step, if you are installing for other CUDA compatible GPU.

You can find list of CUDA compatible GPU here: NVIDIA CUDA GPUs

Step 1: Install Graphic Card Driver

Assuming you are working on a fresh ubuntu 20.04 installation, first step is to install driver for your graphic card.

Run below in Terminal:

```
sudo add-apt-repository ppa:graphics-drivers/ppa
```

Run below in Terminal:

```
nvidia-smi
```

```
harish@harish-N85-87HP6:~$ nvidia-smi
Tue Aug 17 15:53:36 2021
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 460.91.03    Driver Version: 460.91.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  GeForce GTX 1060    Off  | 00000000:01:00.0 Off |                  N/A |
| N/A   61C    P2    26W /  N/A |    475MiB /  6078MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A       904      G   /usr/lib/xorg/Xorg                280MiB |
|    0   N/A  N/A      1284      G   /usr/bin/gnome-shell               84MiB |
|    0   N/A  N/A      2082      G   /opt/zoom/zoom                     29MiB |
|    0   N/A  N/A      2309      G   /usr/lib/firefox/firefox            1MiB |
|    0   N/A  N/A      2419      G   /usr/lib/firefox/firefox            1MiB |
|    0   N/A  N/A      2872      G   /usr/lib/firefox/firefox            1MiB |
|    0   N/A  N/A      2924      G   /usr/lib/firefox/firefox            1MiB |
|    0   N/A  N/A      3125      G   /usr/lib/firefox/firefox            1MiB |
|    0   N/A  N/A      7771      G   /usr/lib/firefox/firefox            1MiB |
|    0   N/A  N/A      8315      G   /usr/lib/firefox/firefox            1MiB |
|    0   N/A  N/A     10552      C   ...ffice/program/soffice.bin       61MiB |
+-----------------------------------------------------------------------------+
```

Note: Please Note the CUDA Version on Top Right. In my case it is 11.2. This is required to follow correct CUDA version.

Step 2: Install CUDA

CUDA is a parallel computing platform and application programming interface model developed by Nvidia to enable using GPU for general purpose processing.

Once you have installed the Nvidia Graphic card driver, install CUDA as per version from nvidia-smi output.

We will be installing CUDA 11.2. **Please use correct CUDA Version.**

```
sudo mv cuda-ubuntu2004.pin /etc/apt/preferences.d/cuda-repository-pin-600

wget https://developer.download.nvidia.com/compute/cuda/11.2.0/local_installers/cuda-repo-ubuntu2004-11-2-local_11.2.0-460.27.04-1_amd64.deb

sudo dpkg -i cuda-repo-ubuntu2004–11–2-local_11.2.0–460.27.04–1_amd64.deb

sudo apt-key add /var/cuda-repo-ubuntu2004–11–2-local/7fa2af80.pub

sudo apt-get update

sudo apt-get -y install cuda
```

Ref: CUDA Toolkit 11.2 Downloads

Note: You can find corresponding installation commands for different CUDA Versions here: CUDA Toolkit Archive

Step 3: Install cuDNN

cuDNN is GPU-accelerated library of primitives developed by Nvidia for deep neural networks.

Yet again you need to install the correct cuDNN version as per CUDA version installed in above step. I have chosen to download "Download cuDNN v8.2.1 (June 7th, 2021), for CUDA 11.x" >> "cuDNN Library for Linux (x86_64)" from cuDNN Archive as my CUDA Version is 11.2. Please select accordingly.

To download you will have to create an Nvidia account. Once downloaded use below instructions.

Run below in Terminal:

```
tar -xzvf cudnn-11.3-linux-x64-v8.2.1.32.tgz

sudo cp cuda/include/cudnn.h /usr/local/cuda/include
```

Set path in .bashrc file.

Run below in Terminal from home directory:

```
gedit .bashrc
```

Above command will open .bashrc in text editor. Please change the version in below path if you are using different CUDA Version. Once added, save and close the text editor.

```
export LD_LIBRARY_PATH=/usr/local/cuda-11.2/lib64:$LD_LIBRARY_PATH

export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH
```

Run below in Terminal from home directory:

```
source ~/.bashrc
```

Step 4: Install PIP

Python3.8 comes pre installed in Ubuntu 20.04. Run below command to install PIP.

Run below in Terminal:

```
sudo apt update

sudo apt install python3-pip
```

Step 5: Install tensorflow

Tensorflow by default comes with GPU support,so no need to install tensorflow-gpu specifically

Run below in Terminal:

```
pip install tensorflow
```

Step 6: Check tensorflow

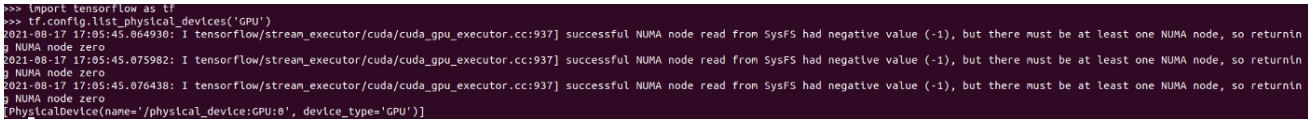
Check GPU is detected by Tensorflow


Run below from python command line:

```
import tensorflow as tf

tf.config.list_physical_devices('GPU')
```

You should be able to see "[PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU')]" as below



Note: If you are facing issues, it will most likely be due to version mismatch.

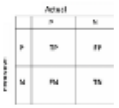
Happy Exploring Deep Learning In-Depth

Ispajic in MLearning.ai

Improving Crowd Wisdom with Machine Learning — part 1

Ayesha M in Analytics Vidhya

My ML Glossary: Part 1

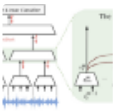


Max Larionov

I Built My Own Self-Driving Car. Part #3

Sumanth S R... in Analytics Vid...

Understanding the Gradient-Isolated Learning of Representations and intuiti...

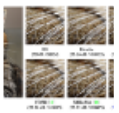


Learner1067 in Analytics Vidhya

Basic ANN using PyTorch

Mike Plotz

Paper Summary: Enhanced Deep Residual Networks for Single Image Super-...



ximnet

What does Microsoft LUIS do and how it affects your AI Chatbot quality?

Chu-Tak Li in Analytics Vidhya

Review of TPNet: Trajectory Proposal Network for Motion Prediction