# Tensorrt安装记录

环境：ubuntu 22.04 python 3.9 CUDA 12.3

以下全程需要在之后要使用的虚拟环境中进行

1. 从 https://developer.nvidia.com/tensorrt 下载与您的系统和CUDA版本相匹配的TensorRT安装包。
2. 切换到安装包所在路径

3.
   ```
   os="ubuntuxx04"
   ```
   ```
   tag="10.x.x-cuda-x.x"
   ```
   ```
   sudo dpkg -i nv-tensorrt-local-repo-${os}-${tag}_1.0-1_amd64.deb
   ```
   ```
   sudo cp /var/nv-tensorrt-local-repo-${os}-${tag}/*-keyring.gpg /usr/share/keyrings/
   ```
   ```
   sudo apt-get update
   ```

4.
   ```
   sudo apt-get install tensorrt
   ```
5. Ensure the pip Python module is up-to-date and the wheel Python module is installed before proceeding or you may encounter issues during the TensorRT Python installation.
   ```
   python3 -m pip install --upgrade pip
   ```
   ```
   python3 -m pip install wheel
   ```
6. Install the TensorRT Python wheel.
   ```
   python3 -m pip install --upgrade tensorrt
   ```

7.
   ```
   python3 -m pip install numpy onnx onnx-graphsurgeon
   ```

8.
   ```
   sudo su
   ```
   ```
   export PATH=/usr/local/cuda-12.3/bin:/usr/local/cuda/bin:$PATH
   ```
9. 回到虚拟环境中，再次
   ```
   export PATH=/usr/local/cuda-12.3/bin:/usr/local/cuda/bin:$PATH
   ```

10.
    ```
    pip install pycuda==2024.1
    ```

11.
    ```
    python3 -m pip install --upgrade setuptools pip
    ```

12.
    ```
    python3 -m pip install nvidia-pyindex
    ```
13. Verify the installation.

For the full TensorRT release

`dpkg-query -W tensorrt`

You should see something similar to the following:

```
tensorrt        10.2.0.x-1+cuda12.5
```

14. 最后验证安装的TensorRT可以在你的虚拟环境下使用

```
import tensorrt

print(tensorrt.__version__)

assert tensorrt.Builder(tensorrt.Logger())
```

15. 检测pycuda能否使用

```
import pycuda.autoinit

import pycuda.driver as drv

import numpy as np

import time

from pycuda.compiler import SourceModule

mod = SourceModule('''

__global__  void Text_GPU(float *A , float *B, float *K, size_t N){

  int bid = blockIdx.x;

  int tid = threadIdx.x;

  __shared__ float s_data[2];

  s_data[tid] = (A[bid*2 + tid] - B[bid*2 + tid]);

  __syncthreads();

  if(tid == 0)

  {

    float sum_d = 0.0;

    for(int i=0;i<N;i++)

    {

      sum_d += (s_data[i]*s_data[i]);

    }
```

>         K[bid] = exp(-sum_d);
>     }
> }
> ''')
>
> multiply_them = mod.get_function("Text_GPU")
> tic = time.time()
> A = np.random.random((1000,20)).astype(np.float32)
> B = np.random.random((1000,20)).astype(np.float32)
> K = np.zeros((1000,), dtype=np.float32)
> N = 20
> N = np.int32(N)
> multiply_them(
>         drv.In(A), drv.In(B), drv.InOut(K), N,
>         block=(20,1,1), grid=(1000,1))
> toc = time.time()
> print("time cost is:"+str(toc-tic))

直接运行一下，如果返回是这样的：

```
time cost is:0.009005308151245117
```

成功。