

TensorFlow2.4 GPU 版本的安装

(Win 10 版本)

1 TensorFlow2.4 支持的系统

- Ubuntu 16.04 or later
- Windows 7 or later
- macOS 10.12.6 (Sierra) or later (no GPU support)
- Raspbian 9.0 or later

2 安装 GPU 驱动和运行库

2.1 检查 GPU 驱动


确保电脑已经安装好了 Nvidia GPU 显卡驱动，官方驱动下载地址在这里：

<https://www.nvidia.com/drivers>。



同时需要确保电脑显卡的运算能力在 3.5 之上，可以在这里查看对应显卡的运算能力，

<https://developer.nvidia.com/cuda-gpus#compute>，例如笔者的电脑显卡为 GeForce GTX 1050，运算能力为 6.1，符合要求。



CUDA-Enabled GeForce and TITAN Products

GeForce and TITAN Products		GeForce Notebook Products	
GPU	Compute Capability	GPU	Compute Capability
NVIDIA TITAN RTX	7.5	GeForce RTX 2080	7.5
GeForce RTX 2080 Ti	7.5	GeForce RTX 2070	7.5
GeForce RTX 2080	7.5	GeForce RTX 2060	7.5
GeForce RTX 2070	7.5	GeForce GTX 1080	6.1
GeForce RTX 2060	7.5	GeForce GTX 1070	6.1
NVIDIA TITAN V	7.0	GeForce GTX 1060	6.1
NVIDIA TITAN Xp	6.1	GeForce GTX 980	5.2
NVIDIA TITAN X	6.1	GeForce GTX 980M	5.2
GeForce GTX 1080 Ti	6.1	GeForce GTX 970M	5.2
GeForce GTX 1080	6.1	GeForce GTX 965M	5.2
GeForce GTX 1070	6.1	GeForce GTX 960M	5.0
GeForce GTX 1060	6.1	GeForce GTX 950M	5.0
GeForce GTX 1050	6.1	GeForce 940M	5.0
GeForce GTX TITAN X	5.2	GeForce 930M	5.0

2.2 安装 GPU 运行库

对于 GPU 版本的 TensorFlow2.4 来说，因为调用了 Nvidia 显卡运行代码，因此需要安装 Nvidia 提供的运行库（GPU 加速工具）。GPU 加速工具有 2 个：CUDA 和 CuDNN。

TensorFlow 的版本一般和运行库的版本是绑定的，一定要配合使用，下载对应的版本，不要改动。

TensorFlow2.4 对应的 Nvidia 运行库版本如下：

CUDA 版本：11.0

cuDNN 版本：8.0

GPU

版本	Python 版本	编译器	构建工具	cuDNN	CUDA
tensorflow_gpu-2.4.0	3.6-3.8	MSVC 2019	Bazel 3.1.0	8.0	11.0
tensorflow_gpu-2.3.0	3.5-3.8	MSVC 2019	Bazel 3.1.0	7.6	10.1
tensorflow_gpu-2.2.0	3.5-3.8	MSVC 2019	Bazel 2.0.0	7.6	10.1
tensorflow_gpu-2.1.0	3.5-3.7	MSVC 2019	Bazel 0.27.1-0.29.1	7.6	10.1
tensorflow_gpu-2.0.0	3.5-3.7	MSVC 2017	Bazel 0.26.1	7.4	10
tensorflow_gpu-1.15.0	3.5-3.7	MSVC 2017	Bazel 0.26.1	7.4	10
tensorflow_gpu-1.14.0	3.5-3.7	MSVC 2017	Bazel 0.24.1-0.25.2	7.4	10
tensorflow_gpu-1.13.0	3.5-3.7	MSVC 2015 update 3	Bazel 0.19.0-0.21.0	7.4	10
tensorflow_gpu-1.12.0	3.5-3.6	MSVC 2015 update 3	Bazel 0.15.0	7.2	9.0
tensorflow_gpu-1.11.0	2.5-2.6	MSVC 2015 update 2	Bazel 0.15.0	7	9

2.2.1 安装 CUDA

1. 下载并安装 CUDA

下载地址为: <https://developer.nvidia.com/cuda-toolkit-archive>

Archived Releases

CUDA Toolkit 11.4.0 (June 2021), Versioned On/line Documentation
CUDA Toolkit 11.3.1 (May 2021), Versioned Online Documentation
CUDA Toolkit 11.3.0 (April 2021), Versioned Online Documentation
CUDA Toolkit 11.2.2 (March 2021), Versioned Online Documentation
CUDA Toolkit 11.2.1 (Feb 2021), Versioned Online Documentation
CUDA Toolkit 11.2.0 (Dec 2020), Versioned Online Documentation
CUDA Toolkit 11.1.1 (Oct 2020), Versioned Online Documentation
CUDA Toolkit 11.1.0 (Sept 2020), Versioned Online Documentation
CUDA Toolkit 11.0 Update1 (Aug 2020), Versioned Online Documentation
CUDA Toolkit 11.0 (May 2020), Versioned Online Documentation
CUDA Toolkit 10.2 (Nov 2019), Versioned Online Documentation
CUDA Toolkit 10.1 update2 (Aug 2019), Versioned Online Documentation
CUDA Toolkit 10.1 update1 (May 2019), Versioned Online Documentation
CUDA Toolkit 10.1 (Feb 2019), Online Documentation

Please Note: Due to an incompatibility issue, we advise users to defer updating to Linux Kernel 5.9+ until mid-November when an NVIDIA Linux GPU driver update with Kernel 5.9+ support is expected to be available.

Select Target Platform

Click on the green buttons that describe your target platform. Only supported platforms will be shown. By downloading and using the software, you agree to fully comply with the terms and conditions of the [CUDA EULA](#).

Operating System

Linux Windows

Architecture

x86_64

Version

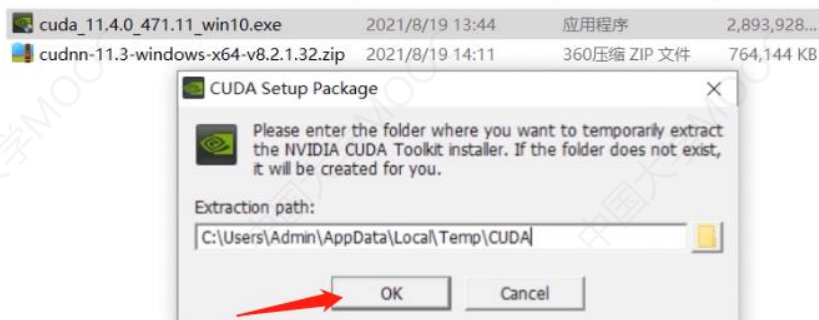
10 Server 2019 Server 2016

Installer Type

exe (local) exe (network)

CUDA 安装步骤

第一步



第二步



第三步



第四步



第五步



第六步



2.2.2 安装 cuDNN

1. 下载

安装好了 CUDA 之后，还需要安装 cuDNN。下载地址：

<https://developer.nvidia.com/cudnn>

下载 cuDNN 需要一个 NVIDIA 账号，自己注册一下就好了。

之后进入下载页面，下载 for CUDA 11.x 版本的 cuDNN。

cuDNN Archive

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

Download cuDNN v8.2.1 (June 7th, 2021), for CUDA 11.x
Download cuDNN v8.2.1 (June 7th, 2021), for CUDA 10.2
Download cuDNN v8.2.0 (April 23rd, 2021), for CUDA 11.x
Download cuDNN v8.2.0 (April 23rd, 2021), for CUDA 10.2

Library for Windows and Linux,

cuDNN Library for Linux (aarch64sbsa)

cuDNN Library for Linux (x86_64)

cuDNN Library for Linux (PPC)

cuDNN Library for Windows (x86)

2. 拷贝

下载完毕之后解压，得到如下文件：



(1)将 bin 目录下的文件复制到：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.4\bin

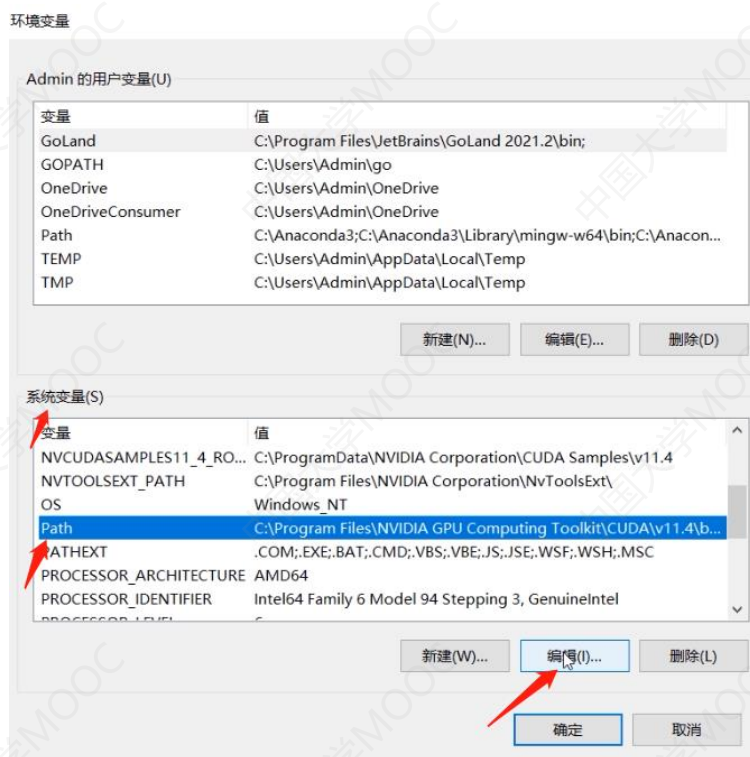
(2)将 include 目录下的文件复制到：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.4\include

(3)将 lib 目录下的文件复制到：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.4\lib\x64

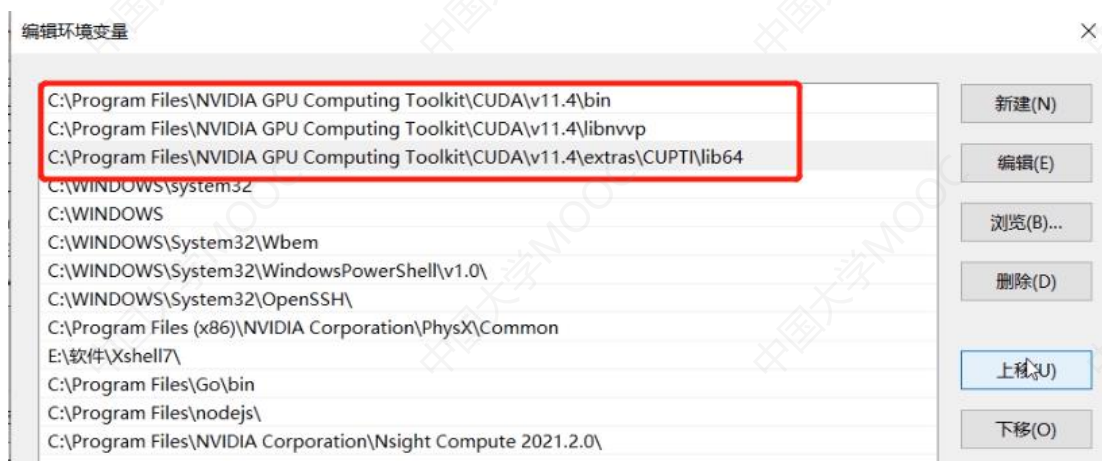
3. 配置环境变量



按照上面的步骤编辑 PATH 环境变量，添加以下路径：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v11.4\extras\lib64

这是配置后的 Path 环境变量：



3 安装 TensorFlow2.4 GPU 版本

1. 在 DOS 命令窗口中,创建并激活独立环境:

```
conda create -n tensorflow-gpu python==3.8
conda activate tensorflow-gpu
```

2. 安装相关软件包

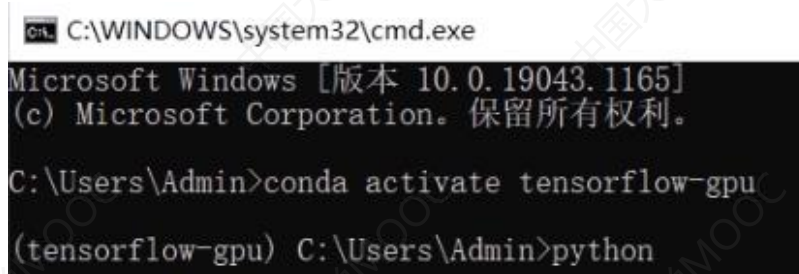
```
pip install numpy matplotlib Pillow scikit-learn pandas -i
https://pypi.tuna.tsinghua.edu.cn/simple
```

3. 安装 TensorFlow-gpu

```
pip install tensorflow-gpu==2.4.0 -i https://pypi.tuna.tsinghua.edu.cn/simple
```

4 测试 GPU 是否配置成功

打开控制台，激活环境：



输入测试语句：

```
Import tensorflow
tf.test.is_gpu_available()
tensorflow_test.is_gpu_available()
```

```
(tensorflow-gpu) C:\Users\Admin>python
Python 3.8.0 (default, Nov 6 2019, 16:00:02) [MSC v
Type "help", "copyright", "credits" or "license" for
>>> import tensorflow
2021-08-19 17:44:07.222173: I tensorflow/stream_exec
library_cudart64_110.dll
>>> tf.test.is_gpu_available()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'tf' is not defined
>>> tensorflow.test.is_gpu_available()
```

返回值为 True，则安装正确。

https://blog.csdn.net/qg_39507748/article/details/104273521

运行 tf2.x 出现错误: Blas GEMM launch failed 的解决方法