



devYurim

[About tags](#) Search[Home](#)[Python](#)[Data Science](#)[Data Base](#)[Development Enviroment](#)

텐서플로우(Tensorflow) GPU 버전 설치하기 - Windows 10

APRIL 30, 2018

연구실 내 컴퓨터에 드디어 GPU가..!! 로컬에서 Tensorflow GPU를 사용하기 위해 험난했던 GPU 버전 설치 방법을 남긴다...

반드시 tensorflow 공식 문서를 확인해야한다!!!

추천 cuda버전, cudnn버전, anaconda일때 파이썬 몇 버전 써야하는지, native pip 일때 파이썬 몇 버전을 써야하는지 적혀있다. 자신의 환경에 맞춰서 공식문서를 보고 파이썬 버전을 잘 선택해야한다. 또한, 선택하는 운영체제에 따라서도 버전이 다르다.

나는 **Windows 10 / Anaconda** 를 사용하여 설치할 예정이다.

- Tensorflow 공식 홈페이지 : <https://www.tensorflow.org/>

1. CUDA Toolkit 설치

텐서플로우 홈페이지에가서 install 버튼을 눌러보면 친절하게 **NVIDIA CUDA xx 설치하세요** 라고 나와있다.

Requirements to run TensorFlow with GPU support

If you are installing TensorFlow with GPU support using one of the mechanisms described in this guide, then the following NVIDIA software must be installed on your system:

- CUDA® Toolkit 9.0. For details, see [NVIDIA's documentation](#). Ensure that you append the relevant Cuda pathnames to the %PATH% environment variable as described in the NVIDIA documentation.
- The NVIDIA drivers associated with CUDA Toolkit 9.0.
- cuDNN v7.0. For details, see [NVIDIA's documentation](#). Note that cuDNN is typically installed in a different location from the other CUDA DLLs. Ensure that you add the directory where you installed the cuDNN DLL to your %PATH% environment variable.
- GPU card with CUDA Compute Capability 3.0 or higher for building from source and 3.5 or higher for our binaries. See [NVIDIA documentation](#) for a list of supported GPU cards.

If you have a different version of one of the preceding packages, please change to the specified versions. In particular, the cuDNN version must match exactly: TensorFlow will not load if it cannot find `cuDNN64_7.dll`. To use a different version of cuDNN, you must build from source.

나는 여기에 맞춰서 **CUDA 9.0** 버전을 설치하였다. 현재는 배포하는 버전은 9.1 버전이여서 **CUDA Toolkit Archive**에 가서 CUDA 9.0을 받았다.

- NVIDIA Cuda Toolkit 설치 : <https://developer.nvidia.com/cuda-toolkit-archive>

CUDA Toolkit 9.0 Downloads

Select Target Platform ⓘ

Click on the green buttons that describe your target platform. Only supported platforms will be shown.

Operating System

WindowsLinuxMac OSX

Architecture ⓘ

x86_64

Version

108.17Server 2016Server 2012 R2

Installer Type ⓘ

exe (network)exe (local)

Download Installers for Windows 10 x86_64

The base installer is available for download below.
There are 2 patches available. These patches require the base installer to be installed first.

> Base Installer

Download [1.4 GB] ⬇

Installation Instructions:

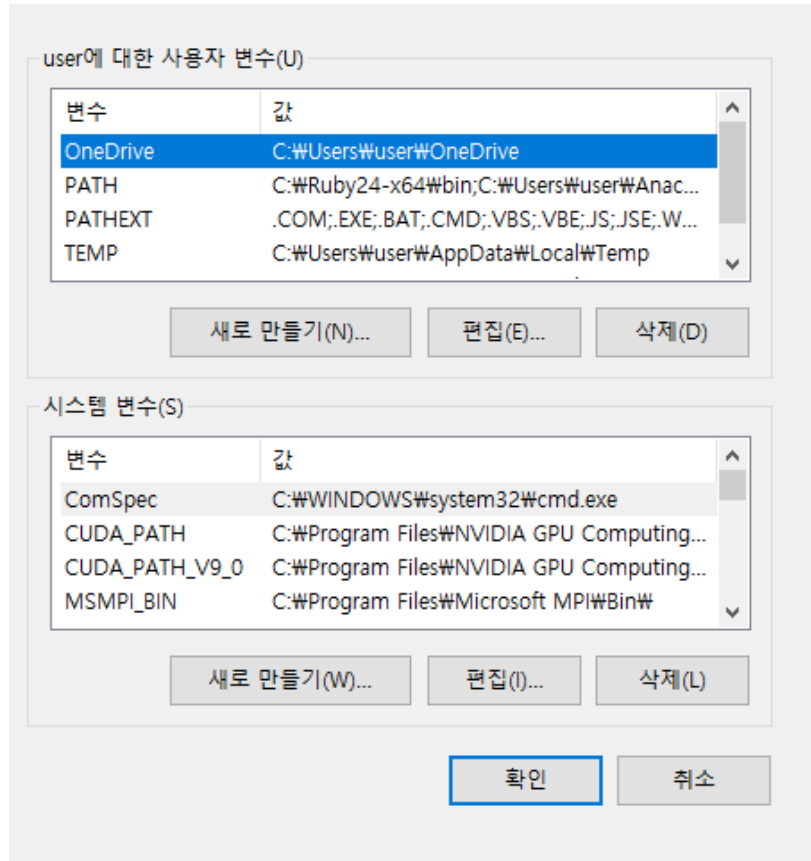
1. Double click cuda_9.0.176_win10.exe

2. Follow on-screen prompts

설치가 완료 되고 재부팅한다음에 **환경변수** 가 제대로 설치 되었나 확인해야한다.

- 시스템 환경변수 편집 -> 고급 -> 환경변수 클릭

환경 변수



위와 같이 **CUDA_PATH** 가 추가되어있으면 OK이다!

만약에 CUDA버전을 여러개 설치한다면 CUDA_PATH_V9_0과 같이 구분되며 최근 설치한 버전이 CUDA_PATH로 들어간다고 한다.

2. cuDNN 다운로드

이것도 역시 텐서플로우 홈페이지에 명시되어 있다. **cuDNN은 7.0.x버전** 으로 다운로드 받았다. cuDNN을 받으려면 DEVELOPER 홈페이지에 가입이 필요하다. 먼저 가입을 하고 다운로드 받는다.

- cuDNN 다운로드 : <https://developer.nvidia.com/rdp/cudnn-archive>

CUDA 버전에 유념해서 다운로드 받아야한다. 나는 **cuDNN v7.0.5 for CUDA 9.0** 을 받았다.

Download cuDNN v7.0.5 [Dec 5, 2017], for CUDA 9.0

압축파일을 압축 해제후 **cuda 폴더** 에 들어가보면 다음과 같은 **3개** 의 파일이 있다.

이름	수정한 날짜	유형	크기
bin	2018-04-30 오후...	파일 폴더	
include	2018-04-30 오후...	파일 폴더	
lib	2018-04-30 오후...	파일 폴더	
NVIDIA_SLA_cuDNN_Support	2017-11-12 오후...	텍스트 문서	39KB

이 세개 파일을 모두 CUDA_PATH 경로의 폴더에 추가해준다. CUDA_PATH 경로는 시스템 환경변수에 가면 확인 할 수 있다. 나는 아래와 같다.

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0

즉, **cuda 폴더 안의 내용을 CUDA_PATH 경로에 ctrl+c / ctrl+v 해서 복사** 하면 완료!

3. Anaconda를 이용해 Tensorflow 설치

나는 아나콘다(Anaconda)가 이미 설치되어 있는 상태이기 때문에 아나콘다를 이용하여 설치하였다. 아나콘다가 없다면 아나콘다를 설치하거나 python3.5버전(홈페이지에서 운영체제 별로 지원하는 파이썬 버전으로 설치 할 것)을 설치하면된다.

일단 현재 버전의 아나콘다는 python3.6을 기준으로하기 때문에 python3.5 버전을 가진 새로운 아나콘다 환경을 만들어 주어야 한다.

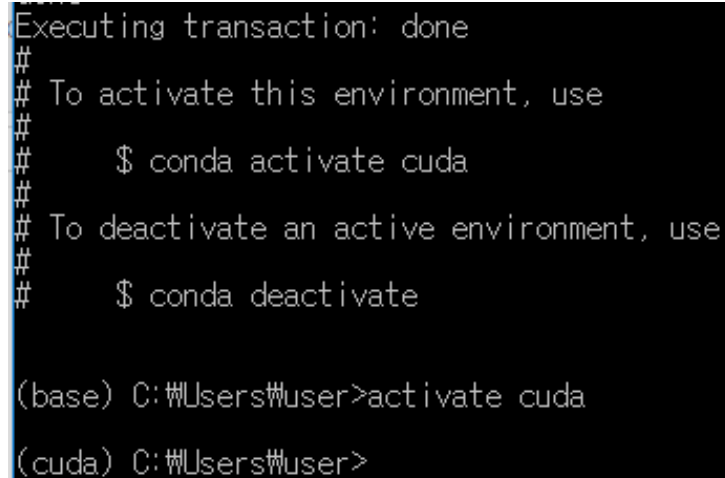
아직 python 3.6은 지원하지 않는 것으로 보이므로(2018.8.29 기준으로 지원) **반드시!!**

python3.5 버전 으로 환경설정을 해주어야 한다.

일단 윈도우 검색창에 Anaconda를 검색하여 **Anaconda Prompt** 를 켜다. 나는 환경이름을 cuda로 설정했다.

환경활성화는 Windows Powershell에서는 안되고 conda prompt에서만 가능하다!

```
conda create -n cuda pip python=3.5 /*환경 만들기(python 3.5) -> y/n 나오면 y 누를것.*/  
activate cuda /*환경 활성화*/
```



```
Executing transaction: done  
#  
# To activate this environment, use  
#  
# $ conda activate cuda  
#  
# To deactivate an active environment, use  
#  
# $ conda deactivate  
  
(base) C:\Users\User>activate cuda  
(cuda) C:\Users\User>
```

activate 하였을 때 다음과 같이 **(cuda)** 로 변경되어야 한다. 다음 아래의 명령어를 실행하여 **Tensorflow GPU 버전** 을 설치한다.

```
python -m pip install --upgrade pip /*일단 pip 를 업그레이드 시켜준다.*/  
pip install --ignore-installed --upgrade tensorflow-gpu /*텐서플로우 GPU 버전 설치*/
```

Tesorflow CPU 버전

은 아래와 같은 명령어로 설치하면 된다.

```
pip install --ignore-installed --upgrade tensorflow
```

설치가 끝나면 jupyter notebook도 설치하고 jupyter notebook을 켜서 코드가 잘 돌아가는지 확인한다.

```
In [3]: import tensorflow as tf

In [4]: hello = tf.constant('Hello! Tensorflow!')
        sess = tf.Session()
        sess.run(hello)

Out[4]: b'Hello! Tensorflow!'

In [ ]:
```

```
2018-04-30 22:29:04.884732: I T:\src\github\tensorflow\tensorflow\core\platform\cpu_feature_guard.cc:140] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2
2018-04-30 22:29:05.509826: I T:\src\github\tensorflow\tensorflow\core\common_runtime\gpu\gpu_device.cc:1356] Found device 0 with properties:
name: GeForce GTX 1050 Ti major: 6 minor: 1 memoryClockRate(GHz): 1.468
pciBusID: 0000:07:00:0
totalMemory: 4.00GiB freeMemory: 3.29GiB
2018-04-30 22:29:05.516205: I T:\src\github\tensorflow\tensorflow\core\common_runtime\gpu\gpu_device.cc:1435] Adding visible gpu devices: 0
2018-04-30 22:29:06.277174: I T:\src\github\tensorflow\tensorflow\core\common_runtime\gpu\gpu_device.cc:923] Device interconnect StreamExecutor with strength 1 edge matrix:
2018-04-30 22:29:06.280052: I T:\src\github\tensorflow\tensorflow\core\common_runtime\gpu\gpu_device.cc:923] 0
2018-04-30 22:29:06.282426: I T:\src\github\tensorflow\tensorflow\core\common_runtime\gpu\gpu_device.cc:942] 0: N
2018-04-30 22:29:06.284771: I T:\src\github\tensorflow\tensorflow\core\common_runtime\gpu\gpu_device.cc:1053] Created TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 with 3025 MB memory) => physical GPU (device: 0, name: GeForce GTX 1050 Ti, pci bus id: 0000:07:00:0, compute capability: 6.1)
```

위에 처럼 예시 코드가 잘 돌아가면 GPU 버전 설치 완료!!

아래의 명령어들은 참고 명령어 사항이다.

```
conda info --envs /*아나콘다 환경 확인하기*/
conda remove --name 삭제하려는환경이름 --all /*아나콘다 환경 삭제하기*/
```

```
(cuda) C:\Users\User>conda info --envs
# conda environments:
#
base                  C:\Users\User\Anaconda3
cuda                  * C:\Users\User\Anaconda3\envs\cuda
tensorflow            C:\Users\User\Anaconda3\envs\tensorflow
```

위와 같이 모든 아나콘다 환경이 표시가 된다. 내가 현재 사용하고 있는 환경은 *로 표시된다.

아나콘다 환경 활성화/비활성화는 아래와 같은 명령어로 실행 할 수 있다.

```
activate cuda /*cuda라는 이름을 가진 아나콘다 환경 활성화*/
deactivate cuda /*cuda라는 이름을 가진 아나콘다 환경 비 활성화*/
```

Reference

- https://www.tensorflow.org/install/install_windows
- <http://solarisailab.com/archives/1581>
- <http://hiuaa.tistory.com/39>
- <http://dwfox.tistory.com/85>

#Tensorflow

Powered by [Jekyll](#) with [Type Theme](#)

댓글 2건 <https://devYurim.github.io>

로그인

♡ 추천 5

Tweet

f 공유

인기순



토론 참여하기

다음으로 로그인

또는 디스커스에 가입하세요. (?)

이름



DOMINIC YOON • 2일 전

감사합니다.
많은 도움이 되었습니다.

^ | v • 답글 • 공유 >



Heerim Yun • 6달 전

설치하면서 계속 에러가 나서 어려웠는데, 이 포스트를 보고 설치완료 했습니다! 잘 동작하네요. 정말 감사합니다~

^ | v • 답글 • 공유 >

HTTPS://DEVYURIM.GITHUB.IO의 다른 댓글.

Docker 컨테이너 연결하기(Mariadb - Django/Python) - Ubuntu 16.04 LTS

댓글 2건 • 9달 전



yurim — ㅋㅋ 분노한거야?

지킬(Jekyll) 깃허브(Github) 블로그에 포스팅하기

댓글 한 건 • 9달 전



장수영 — 안녕하세요 지킬로 블로그 개설하고 나서도 어떻게 포스팅해야 하는지 막막했는데

Windows와 Ubuntu간의 공유 폴더(Samba) 생성

댓글 한 건 • 10달 전



황수영 — 쌤바!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Docker를 이용하여 Django 웹 데이터베이스 서버 셋팅

댓글 한 건 • 9달 전



황수영 — ♥♥ S2 갓.유.림 S2 ♥♥

PREVIOUS POST

Machine Learning(머신러닝) 시작하기

NEXT POST

Java JDK 설치하기 - Windows 10 / MAC OS