

TensorFlow Installation

A walkthrough to install TensorFlow for Python 3.6. If you already have TensorFlow on your machine and feel confident in your installation, you can go ahead and skip this notebook.

For the purposes of this tutorial, I'm recommending using Anaconda as your Python distribution and for managing your environments. If you'd rather use some other distribution or version of Python and feel confident in your abilities, you may also choose to just follow the instructions here: <https://www.tensorflow.org/install/>.

This tutorial is broken up by operating system (OS). Pick the instructions for the OS that you plan to use, and then check your install:

Contents

1. Ubuntu
2. Windows
3. Mac OS X
4. Verify TensorFlow Installation

Ubuntu

1. Download and install Anaconda

1.1 Go [here](#) and click on the Python 3.7 installer matching your system's architecture (likely 64-bit X86).

1.2 Navigate to your downloads directory in a terminal and run the installer:

```
cd ~/Downloads/  
bash Anaconda3-5.3.0-Linux-x86_64.sh
```

1.3 If the Anaconda installer gives the option to prepend itself to PATH, we recommend that you do, as it'll allow you to run the below conda commands from anywhere.

2. In a terminal, create a conda environment for TensorFlow:

```
conda create -n tensorflow python = 3.6
```

Note: This should install with Python 3.6. This is what we want for this tutorial, but if for some reason you'd like to use a different version, modifying "python=[x.x]" will instead create a environment with Python version [x.x] (eg 2.7, 3.5).

3. Activate your new environment:

```
source activate tensorflow
```

You will see your prompt prepend (tensorflow) to indicate that you are in your tensorflow environment.

Note: You will have to re-activate your environment each time you open a new terminal.

4. Install either the CPU version or GPU version of TensorFlow:

CPU: (Recommended as sufficient for this class)

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

GPU: (You'll also need to install some [Nvidia software](#))

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

Note: If you have GPUs available, you should use the GPU version for any serious research, as it is often 10-25x faster. For the purposes of this demo though, the CPU version is sufficient.

5. Additional packages to install (optional, but recommended):

These aren't strictly required to build TensorFlow models, but some handy packages that are used in some of the notebooks for this class:

```
pip install matplotlib jupyter tqdm
```

matplotlib is a Python library for MATLAB-style plotting, jupyter is an application for editing/running IPython notebooks in a web browser, and tqdm is nice package for generating progress bars.

Windows

1. Download and install Anaconda

1.1 Go [here](#) and click on the Python 3.7 installer matching your system's architecture (likely 64-bit X86).

1.2 Open the installer from your downloads folder and run it by double clicking on it.

1.3 Follow the installer's instructions. Make sure to enable the option to make Anaconda your default Python. Also, although the Anaconda installer recommends you don't let it prepend itself to PATH, doing so allows you to call the conda commands below from anywhere, which can be very convenient.

2. Open a Command Prompt and create a conda environment for TensorFlow:

```
conda create -n tensorflow python = 3.6
```

Note: This should install with Python 3.6. This is what we want for this tutorial, but if for some reason you'd like to use a different version, modifying "python=[x.x]" will instead create a environment with Python version [x.x] (eg 2.7, 3.5).

3. Activate your new environment:

```
activate tensorflow
```

You will see your prompt prepend (tensorflow) to indicate that you are in your tensorflow environment.

Note: You will have to re-activate your environment each time you open a new terminal.

4. Install either the CPU version or GPU version of TensorFlow:

CPU: *(Recommended as sufficient for this class)*

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

GPU: (You'll also need to install some [Nvidia software](#))

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

Note: If you have GPUs available, you should use the GPU version for any serious research, as it is often 10-25x faster. For the purposes of this demo though, the CPU version is sufficient.

5. Additional packages to install (optional, but recommended):

These aren't strictly required to build TensorFlow models, but some handy packages that are used in some of the notebooks for this class:

```
Shell pip install matplotlib jupyter tqdm
```

matplotlib is a Python library for MATLAB-style plotting, jupyter is an application for editing/running IPython notebooks in a web browser, and tqdm is nice package for generating progress bars.

Mac OS X

1. Download and install Anaconda

1.1 Go [here](#) and click on the command line Python 3.7 installer.

1.2 Navigate to your downloads directory in a terminal and run the installer:

```
cd ~/Downloads/  
bash Anaconda3-5.3.0-MacOSX-x86_64.sh
```

1.3 If the Anaconda installer gives the option to prepend itself to PATH, we recommend that you do, as it'll allow you to run the below conda commands from anywhere.

2. In a terminal, create a conda environment for TensorFlow:

```
conda create -n tensorflow python=3.6
```

Note: This installs with Python 3.6. This is what we want for this tutorial, but if for some reason you'd like to use a different version, modifying "python=[x.x]" will instead create a environment with Python version [x.x] (eg 2.7, 3.5).

3. Activate your new environment:

```
source activate tensorflow
```

You will see your prompt prepend (tensorflow) to indicate that you are in your tensorflow environment.

Note: You will have to re-activate your environment each time you open a new terminal.

4. Install the CPU version of TensorFlow:

CPU:

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

GPU: Unfortunately, TensorFlow with GPU compatibility is no longer supported on Mac OS X

Note: If you have them available, you should use a machine with GPUs, as it is often 10-25x faster. For the purposes of this demo though, the CPU version is sufficient.

5. Additional packages to install (optional, but recommended):

These aren't strictly required to build TensorFlow models, but some handy packages that are used in some of the notebooks for this class:

```
Shell pip install matplotlib jupyter tqdm
```

matplotlib is a Python library for MATLAB-style plotting, jupyter is an application for editing/running IPython notebooks in a web browser, and tqdm is nice package for generating progress bars.

Verify TensorFlow Installation

1. While in your tensorflow environment, enter the Python shell with:

```
python
```

Verify that your python version says Python 3.6.[x].

2. Enter this program into your Python shell:

```
import tensorflow as tf
hello = tf.constant('Hello, TensorFlow!')
sess = tf.Session()
print(sess.run(hello))
```

Possible warnings that The TensorFlow library wasn't compiled to use * instructions can be ignored. These warnings state that building from source can lead to speed improvements, but these won't make a dramatic difference for these demos.

Your command line should return:

```
b'Hello, TensorFlow!'
```