TensorFlow Installation

2019 - 2020

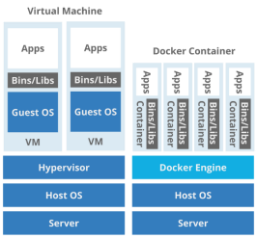
Ando Ki, Ph.D.
adki@future-ds.com

Contents

- TensorFlow installation on Ubuntu
  - TensorFlow installation with Virtualenv on Ubuntu
  - Running TensorFlow code

TensorFlow installation on Ubuntu

- Refer to 'https://www.tensorflow.org/install/install_linux'
- Types of TensorFlow
  - TensorFlow with CPU support only ← *install this for the first time*
  - TensorFlow with GPU support: for nVIDIA GPU
- Install mechanism
  - virtualenv ← *install this way*
    - a tool to create isolated Python environment
      - To deal with version dependency problems
  - "native" pip
    - Pip is package manager
  - Docker
  - Anaconda: a python distribution
    - Conda is a packaging tool and installer that aims to do more than what pip does
    - Anaconda is package manager + environment manager + additional scientific libraries



- virtualenv creates a folder which contains all the necessary executables to use the packages that a Python project would need.

TensorFlow installation with Virtualenv on Ubuntu

- Step 1: Install pip and virtualenv (for Python 2.7)
- Step 2: Create a virtual environment
- Step 3: Activate virtualenv
- Step 4: Install TensorFlow under the virtualenv
- Step 5: Test TensorFlow (see next slide)

TensorFlow requires Python 2.7
TensorFlow requires Python 3.3 or later

TensorFlow installation with Virtualenv on Ubuntu

Refer to 'https://www.tensorflow.org/install/install_linux'

```
$ sudo apt-get install python-pip python-dev python-virtualenv
$ virtualenv --system-site-packages ~/tensorflow
$ source ~/tensorflow/bin/activate # bash, sh, ksh, or zsh
(tensorflow)$ python --version
Python 2.7.6
(tensorflow)$ pip install --upgrade tensorflow
(tensorflow)$ pip list | grep tensorflow
(standard input): tensorflow (1.12.0)
(tensorflow)$ python -c 'import tensorflow as tf; print(tf.__version__)'
1.12.0
```

- Install pip and virtualenv
- Create a virtual env
- Activate the virtualenv
- Check version of Python
- Virtualenv prompt
- Install Tensorflow on the active virtual env
- Check version of TensorFlow

To uninstall TensorFlow, simply remove the directory as follows,
$ /bin/rm –rf ~/tensorflow

When "(tensorflow)$ pip install –upgrade tensorflow" fails,
(tensorflow)$ pip install --upgrade ***tfBinaryURL***

ubuntu

- Install Tensorflow on the active virtual env; see bottom for URL

tfBinaryURL for Tensorflow CPU only with Python 2.7 for Ubuntu 14.04
https://storage.googleapis.com/tensorflow/linux/cpu/tensorflow-1.2.1-cp27-none-linux_x86_64.whl

5

TensorFlow installation with Virtualenv on VirtualBox/Ubuntu 14.04

Refer to 'https://www.tensorflow.org/install/install_linux#InstallingVirtualenv'

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install python-pip python-dev python-virtualenv
$ virtualenv --system-site-packages ~/tensorflow
$ source ~/tensorflow/bin/activate # bash, sh, ksh, or zsh
(tensorflow)$ easy_install -U pip
(tensorflow)$ pip install tensorflow==1.12
(tensorflow)$ pip list | grep tensorflow
(standard input): tensorflow (1.12.0)
(tensorflow)$ python -c 'import tensorflow as tf; print(tf.__version__)'
1.12.0
```

To uninstall TensorFlow, simply remove the directory as follows,
$ /bin/rm –rf ~/tensorflow

When "(tensorflow)$ pip install –upgrade tensorflow" fails,
(tensorflow)$ pip install --upgrade ***tfBinaryURL***

ubuntu VirtualBox

tfBinaryURL for Tensorflow CPU only with Python 2.7 for Ubuntu 14.04
https://storage.googleapis.com/tensorflow/linux/cpu/tensorflow-1.2.1-cp27-none-linux_x86_64.whl

6

Running TensorFlow code

```
$ source ~/tensorflow/bin/activate
(tensorflow)$ vi hello.py

(tensorflow)$ python hello.py
Hello, TensorFlow!
(tensorflow)$ deactivate

$
```

Activate the virtualenv
Note that '~/tensorflow' is where the Python virtual environment has bee built.

Prepare TensorFlow code

Running a tensorflow script

Exit from virtualenv

Get back to normal shell

```
# Python 'hello.py'
import os
os.environ['TF_CPP_MIN_LOG_LEVEL']='2'

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

Deactivate warnings

Tensorflow testing after installation

Running TensorFlow code

- This example shows how to run TensorFlow
  - Step 1: go to your project directory
    - [user@host] cd $(PROJECT)/codes/tensorflow-project/hello
  - Step 2: see the codes
  - Step 3: run Python under virtual environment
    - (do not forget to run '$ source ~/tensorflow/bin/activate')
    - [user@host] python hello.py

```
[user@host] cd $(PROJECT)/codes/tensorflow-project/hello
[user@host] python hello.py
```

Refer: Install TensorFlow without GPU on Windows

- Step 1: install 'Visual C++ 2015 redistributable (x64 version)'
  - https://www.microsoft.com/en-us/download/details.aspx?id=53587
- Step 2: install Phthon 3.5.2 64-bit version (Wdinsows x86-64 execuable installer)
  - https://www.python.org/downloads/release/python-352/
- Step 3: check phthon with command widwon
  - c:\user\adki> python
  - >>> quit()
- Step 4: invoke Windows command window with administrator to install TensorFlow with pip
  - C:\windows\system32> pip install --upgrade https://storage.googleapis.com/tensorflow/windows/cpu/tensorflow-0.12.0rc1-cp35-cp35m-win_amd64.whl
- Step 5: testing TensorFlow
  - c:\user\adki> import tensorflow as tf
  - c:\user\adki> hello = tf.constant('Hello, TensorFlow!')
  - c:\user\adki> sess = tf.Session()
  - c:\user\adki> print(sess.run(hello))
  - c:\user\adki> quit()

9

㈜퓨쳐디자인시스템
34051 대전광역시 유성구 문지로 193, KAIST 문지캠퍼스, F723호
(042) 864-0211~0212 / contact@future-ds.com / www.future-ds.com

Future Design Systems, Inc.
Faculty Wing F723, KAIST Munji Campus, 193 Munji-ro, Yuseong-gu, Daejeon 34051, Korea
+82-042-864-0211~0212 / contact@future-ds.com / www.future-ds.com

FUTURE
Design Systems