1_installation

February 23, 2019

1   Installing Keras and TensorFlow on OSX

1.1   References

- [Installing Keras on Mac OS (High Sierra and below), Python 3.4-3.6](#)
- [Troubleshooting the loading of datasets from Keras](#)
- [Francois Chollet on Running Keras from AWS](#)
- [Jupyter Notebook tutorial from DataCamp](#)

1.2   Steps

1. First, install Python 3.6 on your machine by following the instructions on the [offical Python website](#). Note that Keras will not run on the current version of Python (version 3.7); you must install a version of Python between 3.4 and 3.6

2. Next, download the Mac package manager `Homebrew` by following the instructions on the [Homebrew website](#). When this is done, the `brew` command will be available in `Terminal`, allowing you to install the packages available on the Homebrew repository

3. Open up the `Terminal` application on your Mac and enter the following commands into your Terminal window. These commands install:

    - the `graphviz` library to allow for the visualisation of graph/networks
    - the Python packages `virtualenv` and `virtualenvwrapper` which allows for the organisation of the packages and dependencies needed to run Keras

```
# Terminal commands to install python and virtualenv
# Note that `$` is the command prompt.
# DON'T COPY THIS CHARACTER WHEN YOU COPY THE COMMANDS BELOW
$ brew install graphviz
$ pip3 install virtualenv virtualenvwrapper
```

4. Find out the location of your Python installation by running the following `Terminal` command. Take note of this location as we will use it in the next step. Note that in my case, the `Terminal` is indicating that the system path along which my Python 3 installation can be found is **/Library/Frameworks/Python.framework/Versions/3.6/bin/python3**. Also take note of its **parent path /Library/Frameworks/Python.framework/Versions/3.6/bin**. Note that these paths might be different for you when you run the `which` command

```
$ which python3
> /Library/Frameworks/Python.framework/Versions/3.6/bin/python3
```

5. Edit the file found at ~/.bash_profile. This step is necessary for virtualenv and virtualenvwrapper to run correctly. You can use any code editor such as Visual Studio Code or Sublime Text to do this

```
# edit the file ~/.bash_profile

# use the path for your Python3 installation found in step 4 here
# to set the VIRTUALENVWRAPPER_PYTHON variable
export VIRTUALENVWRAPPER_PYTHON=/Library/Frameworks/Python.framework/Versions/3.6/bin/python3

# use the parent path found in step 4 here to set the PYTHONBIN variable
export PYTHONBIN=/Library/Frameworks/Python.framework/Versions/3.6/bin

source $PYTHONBIN/virtualenvwrapper.sh
```

6. Load the edited .bash_profile filed in Terminal to pick up the file edits

```
$ source ~/.bash_profile
```

7. Create a virtual environment named dlpy (you can use any name you wish but I have chosen dlpy to represent "Deep Learning Python")

```
# make a new virtual environment named `dlpy`
$ mkvirtualenv dlpy -p python3

# if the command above succeeds, your command prompt should now display differently,
# indicating that the `dlpy` virtual environment is now active in your Terminal
# session as shown below
# Mine has changed from `$` (might be different for you) to `(dlpy)`
(dlpy)
```

8. Install the Python 3 packages you need to run Deep Learning models on your computer by entering the following command into Terminal:

```
# install python modules for deep learning to this virtual environment
(dlpy) pip3 install numpy scipy matplotlib h5py pydot-ng tensorflow keras
      jupyter scikit-learn seaborn
```

9. Start the Jupyter notebook by entering the following command into Terminal. Entering the keyboard command control-C in the Terminal closes the Jupyter Notebook

```
# Start Jupyter notebook. Change the string "dlpy" if you have
# chosen another name for your Python3 virtual environment
(dlpy) ~/.virtualenvs/dlpy/bin/jupyter-notebook
```

10. To deactivate the virtual environment dlpy after you are done, enter the Terminal command below:

```
# to deactivate the `dlpy` virtual environment
(dlpy) deactivate

# to reactivate the `dlpy` virtual environment when you need to
# install other Python 3 packages/modules to this environment
$ workon dlpy
```

11. To start up Jupyter Notebook using Python3 within the `dlpy` environment where you have installed `Keras` and its dependencies from Step 8, simply navigate to the system path where you have your previous work (`.ipynb` files stored) and rerun the following command from Step 9. You should now be able to run the Jupyter notebooks from the Deep Learning talk given at WorkCentral on 16 Feb 2019.

```
$ ~/.virtualenvs/dlpy/bin/jupyter-notebook
```