

Docker Demystified for Data Scientists: What's the fuss about and how to get started



Shaheen Gauher
PhD, Data Scientist
@Shaheen_Gauher 

Session Goals

- Get started with using Docker for your AI work
 - Basic docker Concepts
 - Few docker commands
 - cntk hello world using cntk image
 - Tensorflow hello world using tensorflow image
 - Custom image with Dockerfile

What is Docker

- Developing code in one environment and deploying in another
- Packaging the application into 'Containers'.
- *Instead of running the code we run the Container.*
- Application code, the libraries and dependencies needed to run the application
- Portable, self sufficient, run anywhere
- Develop, ship, deploy

What are Images and Containers

Images

- Templates for docker containers

- Combination of a file system and parameters

- Basic image, sophisticated image

- Download from docker hub or build it

- Create multiple containers from a single image

Containers

- Running (or stopped) instances of some image

- Fully functional, isolated

Some commands

```
docker pull      # get image from docker hub
docker images    # list all images
docker ps -a     # list all containers
docker ps        # list running containers
docker stop      # stop a running container
docker start     # start a stopped container
docker cp        # copy from the container's file
system to the   local machine and vice versa
```

Some commands

```
docker run -i -t image_name [command]
```

```
docker run -d -p image_name [command]
```

-d, --detach=false Run container in background (in detached mode) and print container ID.
-t, --tty=false Allocate a pseudo-TTY
-i, --interactive Keep STDIN open even if not attached
-p, --publish Mapping the port number of the Docker host to the port number on our
localhost to access the application running on that port in the container. (port 8888 is default for
Jupyter Notebook application)
[command] e.g. /bin/bash echo 'hello'

```
docker exec            # to execute a command in a  
running container
```

```
docker commit        # create a new image from a  
container's change
```

```
docker build        # build an image from a dockerfile
```

Some commands

== To delete all docker images ==

```
docker rmi $(docker images -q) # must delete  
containers first else will get - image is being  
used message
```

```
docker rmi $(docker images -q "dangling=true") #  
remove all images
```

```
docker rmi the_imageid # remove a single image,  
the_imageid should be the image id
```

Some commands

== To delete all docker containers ==

`docker rm $(docker ps -a -q)` # removes all stopped containers. You cannot remove a running container. Stop the container before attempting removal or force remove

`docker rm -f $(docker ps -a -q)`

`docker stop $(docker ps -qa)` # to stop all containers.

CNTK

<https://hub.docker.com/r/microsoft/cntk/>

➤ ↺

🔒 <https://hub.docker.com/r/microsoft/cntk/>

📖 ☆ ☆≡ 📝


page

Enter text to search

No results

< >

Options ▾

 Search

Explore Help [Sign up](#) [Sign in](#)

PUBLIC | AUTOMATED BUILD

microsoft/cntk ☆

Last pushed: 5 days ago

Repo Info

Tags

Dockerfile

Build Details

Short Description

CNTK images from github.com/Microsoft/CNTK-docker (See information on CNTK at <https://cntk.ai/>)

Full Description

See how to use the images in [CNTK Wiki](#).

The following tags default to images based on ubuntu:14.04 (CPU), nvidia/cuda:8.0-runtime-ubuntu14.04 (v.2.x GPU) and nvidia/cuda:7.5-runtime (v.1.x GPU).

Version 2 Images

Python 3.4 and 3.5


GPU ([NVIDIA Docker](#) is required)

- 2.3-gpu-python3.5-cuda8.0-cudnn6.0 - latest (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.2-gpu-python3.5-cuda8.0-cudnn6.0 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.1-gpu-python3.5-cuda8.0-cudnn6.0 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.0-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.0.rc3-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.0.rc2-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.0.rc1-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.0.beta15.0-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.0.beta12.0-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-14.04/version_2/gpu/runtime/python-3/Dockerfile)
- 2.0.beta11.0-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-14.04/version_2/0 beta11.0/gpu/runtime/Dockerfile)
- 2.0.beta10.0-gpu-python3.5-cuda8.0-cudnn5.1 (/ubuntu-


Docker Pull Command

`docker pull microsoft/cntk`

Owner

 microsoft

Source Repository

 [Microsoft/CNTK-docker](#)

CNTK Example

~\$ `docker pull microsoft/cntk` # This will get the latest image, which today means latest available GPU runtime configuration.

~\$ `docker pull microsoft/cntk:2.1-cpu-python3.5` # To get a specific configuration you need to add a tag. This will get you CNTK 2.1 CPU runtime configuration set up for Python 3.5.

~\$ `docker images`

REPOSITORY		TAG	IMAGE ID	
CREATED	SIZE			
microsoft/cntk		2.1-cpu-python3.5	57f6b9f1b27c	2
months ago	6.74GB			

CNTK Example

Jupyter Notebook
Container Shell

```
docker run -it -p 8888:8888 --name cntkdemo1 microsoft/cntk:2.1-cpu-python3.5 bash -c  
"source /cntk/activate-cntk && jupyter-notebook --no-browser --port=8888 --ip=0.0.0.0 --  
notebook-dir=/cntk/Tutorials --allow-root"
```

```
PS C:\Users\gshaheen> docker run -it -p 8888:8888 --name cntkdemo1 microsoft/cntk:2.1-cpu-python3.5 bash -c "source /cntk/activate-cntk && jupyter-  
notebook --no-browser --port=8888 --ip=0.0.0.0 --notebook-dir=/cntk/Tutorials --allow-root"
```

```
*****
```

CNTK is activated.

Please checkout tutorials and examples here:

/cntk/Tutorials

/cntk/Examples

To deactivate the environment run

source /root/anaconda3/bin/deactivate

```
*****
```

[I 22:54:38.373 NotebookApp] Writing notebook server cookie secret to /root/.local/share/jupyter/runtime/notebook_cookie_secret

[I 22:54:38.434 NotebookApp] Serving notebooks from local directory: /cntk/Tutorials

[I 22:54:38.434 NotebookApp] 0 active kernels

[I 22:54:38.434 NotebookApp] The Jupyter Notebook is running at: http://0.0.0.0:8888/?token=60ab283d7c4da6ae5111db3354f6e70e2074976bdacf67ea

[I 22:54:38.434 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).

[C 22:54:38.435 NotebookApp]

Copy/paste this URL into your browser when you connect for the first time,

to login with a token:

http://0.0.0.0:8888/?token=60ab283d7c4da6ae5111db3354f6e70e2074976bdacf67ea

^C[I 22:54:58.851 NotebookApp] interrupted

Serving notebooks from local directory: /cntk/Tutorials

0 active kernels

The Jupyter Notebook is running at: http://0.0.0.0:8888/?token=60ab283d7c4da6ae5111db3354f6e70e2074976bdacf67ea

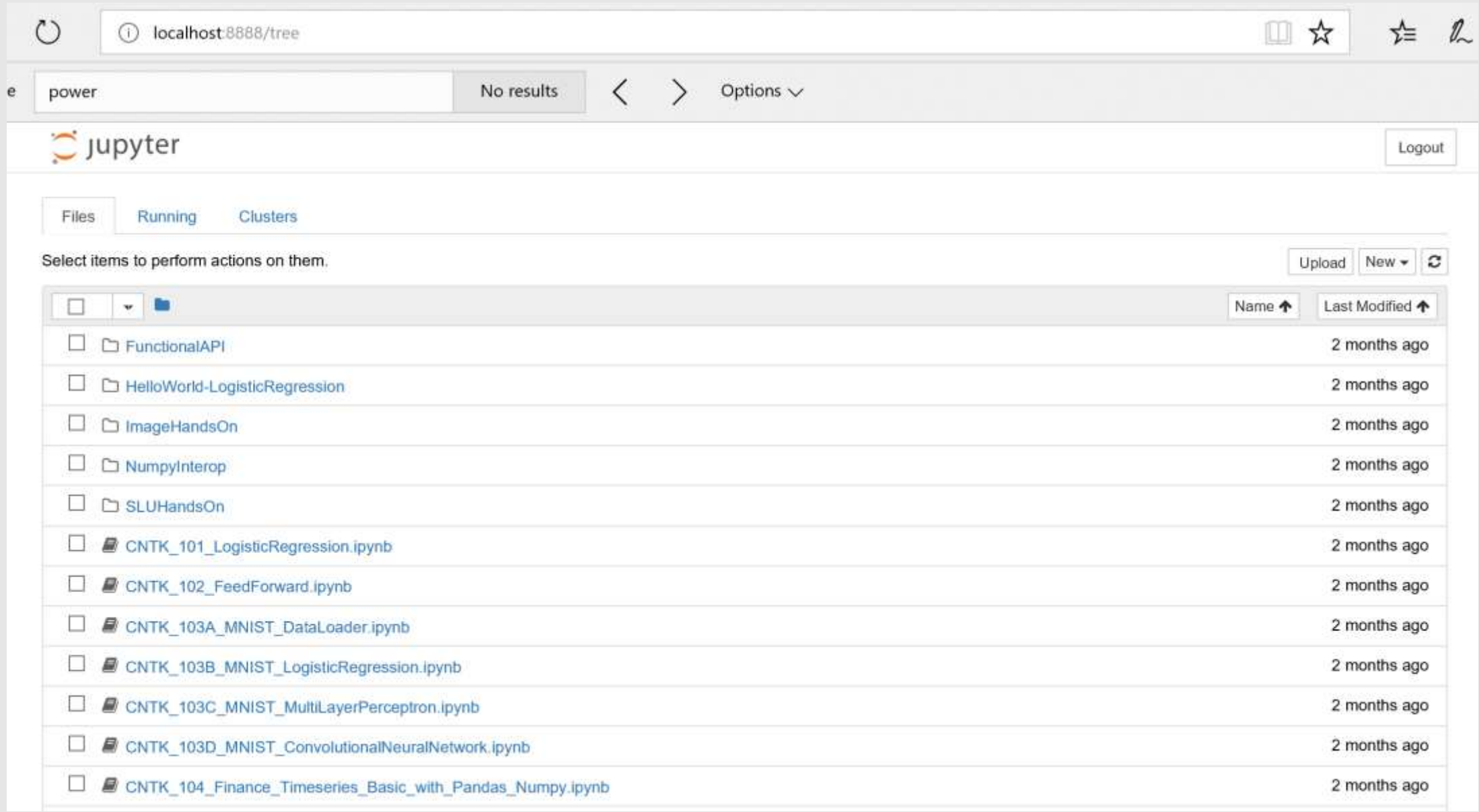
Shutdown this notebook server (y/[n])? y

[C 22:55:00.647 NotebookApp] Shutdown confirmed

[I 22:55:00.648 NotebookApp] Shutting down kernels

PS C:\Users\gshaheen>

CNTK Example – Jupyter Notebook



The screenshot displays the Jupyter Notebook web interface. At the top, the browser address bar shows 'localhost:8888/tree'. Below the address bar, there is a search bar containing the text 'power', which has returned 'No results'. The Jupyter logo is visible on the left, and a 'Logout' button is on the right. The main content area has tabs for 'Files', 'Running', and 'Clusters', with 'Files' being the active tab. A message 'Select items to perform actions on them.' is shown above a table of files. The table has columns for 'Name' and 'Last Modified'. The files listed are:

Name	Last Modified
FunctionalAPI	2 months ago
HelloWorld-LogisticRegression	2 months ago
ImageHandsOn	2 months ago
NumpyInterop	2 months ago
SLUHandsOn	2 months ago
CNTK_101_LogisticRegression.ipynb	2 months ago
CNTK_102_FeedForward.ipynb	2 months ago
CNTK_103A_MNIST_DataLoader.ipynb	2 months ago
CNTK_103B_MNIST_LogisticRegression.ipynb	2 months ago
CNTK_103C_MNIST_MultiLayerPerceptron.ipynb	2 months ago
CNTK_103D_MNIST_ConvolutionalNeuralNetwork.ipynb	2 months ago
CNTK_104_Finance_Timeseries_Basic_with_Pandas_Numpy.ipynb	2 months ago

CNTK Example

Jupyter Notebook
Container Shell

CNTK Example

Jump inside the container

Perform actions – download and install software, configure files etc.

Project – Need cntk and lightgbm. e.g. Transfer learning using resnet50 and lightgbm classifier. Need git. e.g. Do a tutorial on github and want to clone the repository.

Commit to save state

Image with cntk and lighgbm

CNTK Example – Install Lightgbm

```
~$ docker run -it --name cntkdemo2 microsoft/cntk:2.1-cpu-python3.5 /bin/bash
```

```
# pip install lightgbm
# mkdir mylightgbmex
```

Ctrl + p + q

To leave a container running

```
~$ docker ps
```

should see the running container cntkdemo2 listed

```
PS C:\Users\gshaheen> docker run -it --name cntkdemo2 microsoft/cntk:2.1-cpu-python3.5 /bin/bash
```

```
*****
```

```
Welcome to Microsoft Cognitive Toolkit (CNTK) v. 2.1
```

```
Activating CNTK environment...
```

```
(Use command below to activate manually when needed)
```

```
source "/cntk/activate-cntk"
```

```
*****
```

```
CNTK is activated.
```

```
Please checkout tutorials and examples here:
```

```
/cntk/Tutorials
```

```
/cntk/Examples
```

```
To deactivate the environment run
```

```
source /root/anaconda3/bin/deactivate
```

```
*****
```

```
(/root/anaconda3/envs/cntk-py35) root@327e271aa28b:~# pip install lightgbm
```

```
Collecting lightgbm
```

```
Downloading lightgbm-2.0.11-py2.py3-none-manylinux1_x86_64.whl (624kB)
```

```
100% |#####| 624kB 1.6MB/s
```

```
Requirement already satisfied (use --upgrade to upgrade): scikit-learn in ./anaconda3/envs/cntk-py35/lib/python3.5/site-packages (from lightgbm)
```

```
Requirement already satisfied (use --upgrade to upgrade): scipy in ./anaconda3/envs/cntk-py35/lib/python3.5/site-packages (from lightgbm)
```

```
Requirement already satisfied (use --upgrade to upgrade): numpy in ./anaconda3/envs/cntk-py35/lib/python3.5/site-packages (from lightgbm)
```

```
Installing collected packages: lightgbm
```

```
Successfully installed lightgbm-2.0.11
```

```
You are using pip version 8.1.2, however version 9.0.1 is available.
```

```
You should consider upgrading via the 'pip install --upgrade pip' command.
```

```
(/root/anaconda3/envs/cntk-py35) root@327e271aa28b:~#
```

```
PS C:\Users\gshaheen> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAME
327e271aa28b	microsoft/cntk:2.1-cpu-python3.5	"/bin/bash"	2 minutes ago	Up 2 minutes		cntkdemo2

```
PS C:\Users\gshaheen>
```

CNTK Example – Install Lightgbm

Copy files from local folder to a folder in the container (can be running or stopped)

```
~$ docker cp
C:\Users\gshaheen\Documents\Docker\mylightgbmex\regression.train
cntkdemo2:/root/mylightgbmex/regression.train

~$ docker cp
C:\Users\gshaheen\Documents\Docker\mylightgbmex\regression.test
cntkdemo2:/root/mylightgbmex/regression.test

~$ docker cp
C:\Users\gshaheen\Documents\Docker\mylightgbmex\mylgbm_ex.py
cntkdemo2:/root/mylightgbmex/mylgbm_ex.py
```

Jump back into the running container, check in directory mylightgbmex for the 3 files that were copied

CNTK Example – Install Lightgbm

```
~$ docker exec -it cntkdemo2  
/bin/bash  
# cd mylightgbmex  
# ls  
# vi mylgbm_ex.py  
# python mylgbm_ex.py  
# ls  
# exit  
  
~$ docker cp  
cntkdemo2:/root/mylightgbmex/m  
odel.txt  
C:\Users\gshaheen\Documents\Do  
cker\mylightgbmex\model.txt
```

```
(/root/anaconda3/envs/cntk-py35) root@f2495dbe6dfc:~# ls  
anaconda3  mylightgbmex  mylightgbmex  
(/root/anaconda3/envs/cntk-py35) root@f2495dbe6dfc:~# cd mylightgbmex/  
(/root/anaconda3/envs/cntk-py35) root@f2495dbe6dfc:~/mylightgbmex# ls  
mylgbm_ex.py  regression.test  regression.train  
(/root/anaconda3/envs/cntk-py35) root@f2495dbe6dfc:~/mylightgbmex# python  
mylgbm_ex.py  
Load data...  
Start training...  
[1]   valid_0's auc: 0.741301 valid_0's l2: 0.243481  
Training until validation scores don't improve for 5 rounds.  
[2]   valid_0's auc: 0.743929 valid_0's l2: 0.240045  
[3]   valid_0's auc: 0.766794 valid_0's l2: 0.236636  
[4]   valid_0's auc: 0.770253 valid_0's l2: 0.232959  
[5]   valid_0's auc: 0.767326 valid_0's l2: 0.229684  
[6]   valid_0's auc: 0.76964  valid_0's l2: 0.226942  
[7]   valid_0's auc: 0.780291 valid_0's l2: 0.223972  
[8]   valid_0's auc: 0.783104 valid_0's l2: 0.220928  
[9]   valid_0's auc: 0.787869 valid_0's l2: 0.217949  
[10]  valid_0's auc: 0.790538 valid_0's l2: 0.21512  
[11]  valid_0's auc: 0.793013 valid_0's l2: 0.212605  
[12]  valid_0's auc: 0.793956 valid_0's l2: 0.210519  
[13]  valid_0's auc: 0.793973 valid_0's l2: 0.208728  
[14]  valid_0's auc: 0.790659 valid_0's l2: 0.207585  
[15]  valid_0's auc: 0.788901 valid_0's l2: 0.206254  
[16]  valid_0's auc: 0.792521 valid_0's l2: 0.20433  
[17]  valid_0's auc: 0.793973 valid_0's l2: 0.20294  
[18]  valid_0's auc: 0.798246 valid_0's l2: 0.201385  
[19]  valid_0's auc: 0.798004 valid_0's l2: 0.20008  
[20]  valid_0's auc: 0.798197 valid_0's l2: 0.198988  
Save model...  
Start predicting...  
The rmse of prediction is: 0.446081045061  
(/root/anaconda3/envs/cntk-py35) root@f2495dbe6dfc:~/mylightgbmex# ls  
model.txt  mylgbm_ex.py  regression.test  regression.train  
(/root/anaconda3/envs/cntk-py35) root@f2495dbe6dfc:~/mylightgbmex#
```

Custom Image

Image with CNTK and lightgbm

```
~$ docker commit cntkdemo2 cntkwlgbm:version1
~$ docker images          # should see cntkwlgbm:version1
~$ docker run -it cntkwlgbm:version1 /bin/bash
```

```
PS C:\Users\gshaheen\Documents\MyGitCollection\LightGBM\examples\regression> docker commit b6d2cf617103 cntkwlgbm:version1
sha256:7bd54b55d78f6da58ccb78ada29269161f5379d9329536d8bcb12b4afec003df
PS C:\Users\gshaheen\Documents\MyGitCollection\LightGBM\examples\regression> docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
cntkwlgbm	version1	7bd54b55d78f	14 seconds ago	4.95GB
tensorflow/tensorflow	latest	a61a91cc0d1b	3 weeks ago	1.25GB
microsoft/cntk	2.2-cpu-python3.5	57f6b9f1b27c	2 months ago	6.74GB
microsoft/cntk	2.1-cpu-python3.5	08ebab3ccf79	3 months ago	4.95GB

```
PS C:\Users\gshaheen\Documents\MyGitCollection\LightGBM\examples\regression>
```

Curse of the Golden Image



<http://vignette3.wikia.nocookie.net/disney/images/5/55/Pirates-of-the-Caribbean-The-Curse-of-the-Black-Pearl-Teaser-Poster.jpg/revision/latest?cb=2016061915323>

Image with CNTK and lightgbm

Build an Image

Dockerfile

requirements.txt

Context

Dockerfile

Set of instructions, directions, and commands that describe an environment configuration

Replace the process of doing everything manually and repeatedly.

At the end of dockerfile execution - Image

Image with CNTK and lightgbm

```
gshaheen@MININT-VFC6HSL MINGW64 ~/Documents/Docker/mycntkwlgbm
$ ls
Dockerfile  mylgbm_ex.py  regression.test  regression.train  requirements.txt
```

```
$ cat Dockerfile
FROM microsoft/cntk:2.1-cpu-python3.5
RUN apt-get update \
    && apt-get install -y git
COPY . /root/mylightgbmex
RUN /root/anaconda3/envs/cntk-py35/bin/pip install -r
/root/mylightgbmex/requirements.txt
CMD [ "echo 'Hello'"]
```

```
$ cat requirements.txt
wheel
lightgbm==2.0.2
```

docker build -t="Image_Name" .

Image with cntk and lightgbm

```
$~ cd C:\Users\gshaheen\Documents\Docke\mylightgbmex  
$~ docker build -t="mycntkwlgbmimage" .    #takes few minutes  
$~ docker images    # should see mycntkwlgbmimage
```

```
PS C:\Users\gshaheen\Documents\Docke\mycntkwlgbm> docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
mycntkwlgbmimage	latest	625ac61d7ebf	7 seconds ago	5.01GB
cntkwlgbm	version1	f305e4a0dc28	2 hours ago	4.95GB
mxnet/python	latest	b1309bbbfd32	2 weeks ago	1.33GB
tensorflow/tensorflow	latest	a61a91cc0d1b	4 weeks ago	1.25GB
microsoft/cntk	2.1-cpu-python3.5	08ebab3ccf79	4 months ago	4.95GB

```
PS C:\Users\gshaheen\Documents\Docke\mycntkwlgbm>
```


Image with cntk and lightgbm

```
$~ docker run -it  
mycntkwlgbmimage /bin/bash  
#cd /root/mylightgbmex/  
# ls  
Dockerfile  mylgbm_ex.py  
regression.test  
regression.train  
requirements.txt  
# python  
# python mylgbm_ex.py  
# ls  
Dockerfile  model.txt  
mylgbm_ex.py  
regression.test  
regression.train  
requirements.txt
```

```
*****  
(/root/anaconda3/envs/cntk-py35) root@4f8f940b1cb8:~# ls  
anaconda3  mylightgbmex  openmpi  
(/root/anaconda3/envs/cntk-py35) root@4f8f940b1cb8:~# cd mylightgbmex/  
(/root/anaconda3/envs/cntk-py35) root@4f8f940b1cb8:~/mylightgbmex# ls  
Dockerfile  mylgbm_ex.py  regression.test  regression.train  requirements.txt  
(/root/anaconda3/envs/cntk-py35) root@4f8f940b1cb8:~/mylightgbmex# python mylgbm_ex.py  
Load data...  
Start training...  
[1] valid_0's auc: 0.741301 valid_0's l2: 0.243481  
Training until validation scores don't improve for 5 rounds.  
[2] valid_0's auc: 0.743929 valid_0's l2: 0.240045  
[3] valid_0's auc: 0.766794 valid_0's l2: 0.236636  
[4] valid_0's auc: 0.770253 valid_0's l2: 0.232959  
[5] valid_0's auc: 0.767326 valid_0's l2: 0.229684  
[6] valid_0's auc: 0.76964 valid_0's l2: 0.226942  
[7] valid_0's auc: 0.780291 valid_0's l2: 0.223972  
[8] valid_0's auc: 0.783104 valid_0's l2: 0.220928  
[9] valid_0's auc: 0.787869 valid_0's l2: 0.217949  
[10] valid_0's auc: 0.790538 valid_0's l2: 0.21512  
[11] valid_0's auc: 0.793013 valid_0's l2: 0.212605  
[12] valid_0's auc: 0.793956 valid_0's l2: 0.210519  
[13] valid_0's auc: 0.793973 valid_0's l2: 0.208728  
[14] valid_0's auc: 0.790659 valid_0's l2: 0.207585  
[15] valid_0's auc: 0.788901 valid_0's l2: 0.206254  
[16] valid_0's auc: 0.792521 valid_0's l2: 0.20433  
[17] valid_0's auc: 0.793973 valid_0's l2: 0.20294  
[18] valid_0's auc: 0.798246 valid_0's l2: 0.201385  
[19] valid_0's auc: 0.798004 valid_0's l2: 0.20008  
[20] valid_0's auc: 0.798197 valid_0's l2: 0.198988  
Save model...  
Start predicting...  
The rmse of prediction is: 0.446081045061  
(/root/anaconda3/envs/cntk-py35) root@4f8f940b1cb8:~/mylightgbmex# ls  
Dockerfile  model.txt  mylgbm_ex.py  regression.test  regression.train  requirements.txt  
(/root/anaconda3/envs/cntk-py35) root@4f8f940b1cb8:~/mylightgbmex#
```

Tensorflow

<https://hub.docker.com/r/tensorflow/tensorflow/>

https://hub.docker.com/r/tensorflow/tensorflow/

Find on page

stylus

No results

<

>

Options ▾

Search

Explore

Help

Sign up

Sign in

PUBLIC REPOSITORY

tensorflow/tensorflow

☆

Last pushed: 6 days ago

Repo info

Tags

Short Description

Official docker images for deep learning framework TensorFlow (<http://www.tensorflow.org>)

Docker Pull Command

`docker pull tensorflow/tensorflow`

Full Description

Start CPU only container

```
$ docker run -it -p 8888:8888 tensorflow/tensorflow
```

Go to your browser on <http://localhost:8888/>

Start GPU (CUDA) container

Install [nvidia-docker](#) and run

```
$ nvidia-docker run -it -p 8888:8888 tensorflow/tensorflow:latest-gpu
```

Go to your browser on <http://localhost:8888/>

Other versions (like release candidates, nightlies and more)

See the list of [tags](#). Devel docker images include all the necessary dependencies to build from source whereas the other binaries simply have TensorFlow installed.

Owner

tensorflow

Tensorflow Example

~\$ `docker pull tensorflow/tensorflow` # This will get the latest image for CPU only container

~\$ `docker pull tensorflow/tensorflow:latest-gpu` # This will get the latest image for GPU (CUDA) container (Install nvidia-docker)

~\$ `docker images`

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
tensorflow/tensorflow	latest	a61a91cc0d1b	3 weeks ago	1.25GB
microsoft/cntk	2.2-cpu-python3.5	57f6b9f1b27c	2 months ago	6.74GB

Tensorflow Example – Jupyter Notebook

```
docker run -it -p 8888:8888 tensorflow/tensorflow
```

```
PS C:\Users\gshaheen> docker run -it -p 8888:8888 tensorflow/tensorflow
[I 04:36:27.172 NotebookApp] Writing notebook server cookie secret to /root/.local/share/jupyter/runtime/notebook_cookie_secret
[W 04:36:27.218 NotebookApp] WARNING: The notebook server is listening on all IP addresses and not using encryption. This is not recommended.
[I 04:36:27.229 NotebookApp] Serving notebooks from local directory: /notebooks
[I 04:36:27.229 NotebookApp] 0 active kernels
[I 04:36:27.230 NotebookApp] The Jupyter Notebook is running at:
[I 04:36:27.230 NotebookApp] http://[all ip addresses on your system]:8888/?token=fa838ae705a13a098d024df55c0c0d57fb2d55cfe1c7b64d
[I 04:36:27.230 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 04:36:27.231 NotebookApp]
```

Copy/paste this URL into your browser when you connect for the first time,
to login with a token:

<http://localhost:8888/?token=fa838ae705a13a098d024df55c0c0d57fb2d55cfe1c7b64d>

```
^C[I 04:36:33.771 NotebookApp] interrupted
Serving notebooks from local directory: /notebooks
0 active kernels
The Jupyter Notebook is running at:
http://[all ip addresses on your system]:8888/?token=fa838ae705a13a098d024df55c0c0d57fb2d55cfe1c7b64d
Shutdown this notebook server (y/[n])? y
[C 04:36:36.434 NotebookApp] Shutdown confirmed
[I 04:36:36.435 NotebookApp] Shutting down 0 kernels
PS C:\Users\gshaheen>
```

Tensorflow Example – Jupyter Notebook

localhost:8888/tree

☆

≡

Logout

Files

Running

Clusters

Select items to perform actions on them.

Upload

New ▾

↺

<input type="checkbox"/> 0 ▾		Name ▾	Last Modified
<input type="checkbox"/>	📄	1_hello_tensorflow.ipynb	23 days ago
<input type="checkbox"/>	📄	2_getting_started.ipynb	23 days ago
<input type="checkbox"/>	📄	3_mnist_from_scratch.ipynb	23 days ago
<input type="checkbox"/>	📄	BUILD	23 days ago
<input type="checkbox"/>	📄	LICENSE	23 days ago

Tensorflow Example – Container Shell

```
~$ docker run -it tensorflow/tensorflow /bin/bash
```

```
PS C:\Users\gshaheen\Documents\Docker> docker run -it tensorflow/tensorflow /bin/bash
root@89bb3217bb8e:/notebooks# ls
1_hello_tensorflow.ipynb  2_getting_started.ipynb  3_mnist_from_scratch.ipynb  BUILD  LICENSE
root@89bb3217bb8e:/notebooks# python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow
>>> import lightgbm
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named lightgbm
>>> import xgboost
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named xgboost
>>> quit()
root@89bb3217bb8e:/notebooks#
```

Docker



Hit the ground running

No overhead for getting started on a deep learning project

No need to set environment, install packages

Guaranteed to run the same every time

Takeaways

Run *Containers* not codes

Run anywhere, will run the same every time

Get started on deep learning projects in any framework within minutes

Questions?

Thank You

