

AWS AMI for ML

Select either **Deep Learning AMI (Ubuntu)** or **Deep Learning AMI (Amazon Linux)**

The screenshot shows the AWS Management Console interface for the 'Launch Instance Wizard'. The browser address bar indicates the URL: <https://eu-west-1.console.aws.amazon.com/ec2/v2/home?region=eu-west-1#LaunchInstanceWizard:>. The console header shows the user 'daniele_bagni' and the region 'Ireland'.

The wizard steps are: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, 7. Review. The current step is 'Step 1: Choose an Amazon Machine Image (AMI)'.

On the left sidebar, under 'Linux (3)', the 'Ubuntu (51)' option is selected. The main content area displays a list of AMIs:

- Accelerated Machine Learning** (InAccel): \$2.00/hr for software + AWS usage fees. Linux/Unix, Ubuntu 16.04 | 64-bit Amazon Machine Image (AMI) | Updated: 3/9/18.
- Deep Learning AMI (Ubuntu)** (aws): \$0.0208 to \$41.944/hr incl EC2 charges + other AWS usage fees. Linux/Unix, Ubuntu 16.04 | 64-bit Amazon Machine Image (AMI) | Updated: 9/25/18. This option is highlighted with a red box.
- Predictive Analytics Framework** (RRECKTEK): \$1.00 to \$2.85/hr for software + AWS usage fees. Linux/Unix, Ubuntu 16.04.03 LTS | 64-bit Amazon Machine Image (AMI) | Updated: 7/17/18.
- ADAPA Decision Engine** (ZEMENTIS): Starting from \$0.99/hr or from \$6,937/yr (20% savings) for software + AWS usage fees. Linux/Unix, Ubuntu 14.04x64 | 64-bit Amazon Machine Image (AMI) | Updated: 6/11/18.

AWS EC2 instance

Launch Instance ▾ Connect Actions ▾

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name	Monitoring	Launch Time	Security Groups
	i-047b3b81cfec10b63	p2.xlarge	eu-west-1b	running	Initializing	None	ec2-54-194-119-10.eu-...	54.194.119.10	-	Administrator-...	disabled	October 5, 2018 at 2:13:47 ...	launch-wizard-12

Instance: i-047b3b81cfec10b63 Public DNS: ec2-54-194-119-10.eu-west-1.compute.amazonaws.com

Description Status Checks

Instance ID Description Comes with latest binaries of deep learning frameworks pre-installed in separate virtual environments: MXNet, TensorFlow, Caffe, Caffe2, PyTorch, Keras, Chainer, Theano and CNTK. Fully-configured with NVIDIA CUDA, cuDNN and NCCL as well as Intel MKL-DNN

Instance state Status available

Instance type Platform Other Linux

Elastic IPs Image Size 75GB

Availability zone Visibility Public

Security groups Owner 898082745236

Scheduled events

AMI ID Deep Learning AMI (Amazon Linux) Version 12.0 (ami-00051cea)

Platform -

IAM role -

Key pair name Administrator-ML-Amazon-Ireland keys

Subnet ID subnet-1bdd087d

Network interfaces eth0

Source/dest. check True

T2/T3 Unlimited -

WARNING: pathnames (on your local PC or on the AWS)

- > The best solution to avoid editing all the path names in the shell and python scripts is to set a soft link to my pathnames, either you work in your account or on the AWS.
- > For example, assuming you are on an **AWS EC2** and have username “**ubuntu**”:
 - >> create your working directory with name `/home/ubuntu/ML`
 - >> your caffe root directory is named `/home/ubuntu/src/caffe_python_2`
 - >> run the following commands (**`ln -s target_name link_name`**):
 - >> `sudo mkdir /home/danieleb`
 - >> `sudo ln -s /home/ubuntu/ML /home/danieleb/ML`
 - >> `sudo mkdir /home/danieleb/caffe_tools`
 - >> `sudo ln -s /home/ubuntu/src/caffe_python_2 /home/danieleb/caffe_tools/Caffe-SSD-Ristretto`

WARNING: pathnames (valid only on the AWS)

- > Once you have done the settings of previous page, you also have to run the following only in case of AWS:

```
>> sudo ln -s /home/ubuntu/src/caffe_python_2/build/install \
/home/danieleb/caffe_tools/Caffe-SSD-Ristretto/distributesudo
```

```
>> ln -s /home/ubuntu/src/caffe_python_2/build/install/bin/caffe \
/home/danieleb/caffe_tools/Caffe-SSD-Ristretto/distribute/bin/caffe.binsudo
```

```
>> ln -s /home/ubuntu/src/caffe_python_2/build/install/bin/compute_image_mean \
/home/ubuntu/src/caffe_python_2/build/install/bin/compute_image_mean.bin
```

Logging in into AWS Ubuntu AMI

```
https://eu-west-1.console.aws.amazon.com/ec2/v2/home?region=eu-west-1#Instances:sort=instanceId

aws Services Resource Groups

ec2-user@ip-172-31-10-57:~
X11 forwarding request failed on channel 0
Please login as the user "ec2-user" rather than the user "root".

danieleb@CentOS63-x86-64:~$ ssh -i "~/ML/AWS/keys/Administrator-ML-Amazon-Ireland_keys.pem" ec2-user@ec2-52-213-185-236.eu-west-1.compute.amazonaws.com
X11 forwarding request failed on channel 0
=====
 _ | _ | _ )
 _ | ( _ | /
 _ | \ _ | _ |
=====
Deep Learning AMI (Amazon Linux) Version 12.0
=====

Please use one of the following commands to start the required environment with the framework of your choice:
for MXNet(+Keras2) with Python3 (CUDA 9.0 and Intel MKL-DNN) source activate mxnet_p36
for MXNet(+Keras2) with Python2 (CUDA 9.0 and Intel MKL-DNN) source activate mxnet_p27
for TensorFlow(+Keras2) with Python3 (CUDA 9.0 and Intel MKL-DNN) source activate tensorflow_p36
for TensorFlow(+Keras2) with Python2 (CUDA 9.0 and Intel MKL-DNN) source activate tensorflow_p27
for Theano(+Keras2) with Python3 (CUDA 9.0) source activate theano_p36
for Theano(+Keras2) with Python2 (CUDA 9.0) source activate theano_p27
for PyTorch with Python3 (CUDA 9.0 and Intel MKL) source activate pytorch_p36
for PyTorch with Python2 (CUDA 9.0 and Intel MKL) source activate pytorch_p27
for CNTK(+Keras2) with Python3 (CUDA 9.0 and Intel MKL-DNN) source activate cntk_p36
for CNTK(+Keras2) with Python2 (CUDA 9.0 and Intel MKL-DNN) source activate cntk_p27
for Caffe2 with Python2 (CUDA 9.0) source activate caffe2_p27
for Caffe with Python2 (CUDA 8.0) source activate caffe_p27
for Caffe with Python3 (CUDA 8.0) source activate caffe_p35
for Chainer with Python2 (CUDA 9.0 and Intel iDeep) source activate chainer_p27
for Chainer with Python3 (CUDA 9.0 and Intel iDeep) source activate chainer_p36
for base Python2 (CUDA 9.0) source activate python2
for base Python3 (CUDA 9.0) source activate python3

Official Conda User Guide: https://conda.io/docs/user-guide/index.html
AWS Deep Learning AMI Homepage: https://aws.amazon.com/machine-learning/amis/
Developer Guide and Release Notes: https://docs.aws.amazon.com/dlami/latest/devguide/what-is-dlami.html
Support: https://forums.aws.amazon.com/forum.jspa?forumID=263
=====
4 package(s) needed for security, out of 4 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2018.03 is available.
[ec2-user@ip-172-31-10-57 ~]$ source activate caffe_p27
```

replace

The various soft links

```
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ ls -l /home/danieleb/caffe_tools
total 0
lrwxrwxrwx 1 root root 33 12 ago 12.44 Caffe-SSD-Ristretto -> /home/ec2-user/src/caffe_python_2
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ ls -l /home/danieleb/caffe_tools/Caffe-SSD-Ristretto/distribute/bin/caffe.bin
lrwxrwxrwx 1 root root 5 12 ago 14.07 /home/danieleb/caffe_tools/Caffe-SSD-Ristretto/distribute/bin/caffe.bin -> caffe
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ ls -l /home/danieleb/caffe_tools/Caffe-SSD-Ristretto/distribute
lrwxrwxrwx 1 root root 15 12 ago 12.58 /home/danieleb/caffe_tools/Caffe-SSD-Ristretto/distribute -> ./build/install
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ ls -l src/caffe_python_2/
total 124
drwxrwxr-x 14 ec2-user ec2-user 4096 19 lug 20.02 build
-rw-rw-r-- 1 ec2-user ec2-user 1180 19 lug 20.01 caffe.cloc
drwxrwxr-x 5 ec2-user ec2-user 4096 19 lug 20.01 cmake
-rw-rw-r-- 1 ec2-user ec2-user 4197 19 lug 20.01 CMakeLists.txt
-rw-rw-r-- 1 ec2-user ec2-user 1917 19 lug 20.01 CONTRIBUTING.md
-rw-rw-r-- 1 ec2-user ec2-user 620 19 lug 20.01 CONTRIBUTORS.md
drwxrwxr-x 5 ec2-user ec2-user 4096 19 lug 20.01 data
lrwxrwxrwx 1 root root 15 12 ago 12.58 distribute -> ./build/install
drwxrwxr-x 4 ec2-user ec2-user 4096 19 lug 20.01 docker
drwxrwxr-x 6 ec2-user ec2-user 4096 19 lug 20.01 docs
drwxrwxr-x 15 ec2-user ec2-user 4096 19 lug 20.01 examples
drwxrwxr-x 3 ec2-user ec2-user 4096 19 lug 20.01 include
-rw-rw-r-- 1 ec2-user ec2-user 210 19 lug 20.01 INSTALL.md
-rw-rw-r-- 1 ec2-user ec2-user 2092 19 lug 20.01 LICENSE
-rw-rw-r-- 1 ec2-user ec2-user 24041 19 lug 20.01 Makefile
-rw-rw-r-- 1 ec2-user ec2-user 4634 19 lug 20.01 Makefile.config
-rw-rw-r-- 1 ec2-user ec2-user 4631 19 lug 20.01 Makefile.config.example
drwxrwxr-x 5 ec2-user ec2-user 4096 19 lug 20.01 matlab
drwxrwxr-x 7 ec2-user ec2-user 4096 19 lug 20.01 models
drwxrwxr-x 3 ec2-user ec2-user 4096 19 lug 20.01 python
-rw-rw-r-- 1 ec2-user ec2-user 2130 19 lug 20.01 README.md
drwxrwxr-x 3 ec2-user ec2-user 4096 19 lug 20.01 scripts
drwxrwxr-x 4 ec2-user ec2-user 4096 19 lug 20.01 src
drwxrwxr-x 3 ec2-user ec2-user 4096 19 lug 20.01 tools
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$
```

replace

The 2 environment variables just setup

replace

```
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ echo $PYTHONPATH
/home/ec2-user/src/caffe_python_2/python

(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ echo $LD_LIBRARY_PATH
/usr/local/cuda-8.0/lib64:/usr/local/cuda-8.0/extras/CUPTI/lib64:/lib/nccl/cuda-8.0/lib:/usr/lib64/openmpi/lib:/usr/local/lib:/usr/lib:/usr/local/mpi/lib:/lib:/usr/lib64/openmpi/lib:/usr/local/lib:/usr/lib:/usr/local/mpi/lib:/lib:/home/ec2-user/anaconda3/envs/caffe_p27/lib/

(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ export PYTHONPATH=/usr/local/lib/python2.7/dist-packages/:$PYTHONPATH
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ export LD_LIBRARY_PATH=/home/ec2-user/src/caffe_python_2/build/install/lib64/:$LD_LIBRARY_PATH

(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ echo $PYTHONPATH
/usr/local/lib/python2.7/dist-packages:/home/ec2-user/src/caffe_python_2/python

(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$ echo $LD_LIBRARY_PATH
/home/ec2-user/src/caffe_python_2/build/install/lib64:/usr/local/cuda-8.0/lib64:/usr/local/cuda-8.0/extras/CUPTI/lib64:/lib/nccl/cuda-8.0/lib:/usr/lib64/openmpi/lib:/usr/local/lib:/usr/lib:/usr/local/mpi/lib:/lib:/usr/lib64/openmpi/lib:/usr/local/lib:/usr/lib:/usr/local/mpi/lib:/lib:/home/ec2-user/anaconda3/envs/caffe_p27/lib/
(caffe_p27) [ec2-user@ip-172-31-10-57 ~]$
```

Launch the first python script...

```
> cd ~/ML/cats-vs-dogs
> source activate caffe_p27
> export LD_LIBRARY_PATH=/home/ec2-
  user/src/caffe_python_2/build/install/lib64/:$LD_LIBRARY_PATH
> export PYTHONPATH=/usr/local/lib/python2.7/dist-packages/:$PYTHONPATH
> python set_the_CATSvsDOGS_prj.py -i ~/ML/cats-vs-dogs
```


aws_caffe_flow_AlexNet.sh script

```
emacs@Prec5820Tow
File Edit Options Buffers Tools Sh-Script Help

# /bin/sh

CAFFE_ROOT=/home/danieleb/caffe_tools/Caffe-SSD-Ristretto
CAFFE_TOOLS_DIR=$CAFFE_ROOT/distribute
# working dir
WORK_DIR=/home/danieleb/ML/cats-vs-dogs/caffe

NUMIT=12000 # number of iterations
NET=alexnetBNnoLRN
MOD_NUM=2 # model number

# #####
# # create the project directories for input images and hiddenly call SCRIPTS 1 2 3 (DATABASES)

source activate caffe_p27
export PYTHONPATH=/usr/local/lib/python2.7/dist-packages/:$PYTHONPATH
export LD_LIBRARY_PATH=/home/ec2-user/src/caffe_python_2/build/install/lib64/:$LD_LIBRARY_PATH

# WARNING: the below script must be called only once, at the first time
# WARNING: you must have the file train.zip placed below ~/ML/cats-vs-dogs!
python ~/ML/cats-vs-dogs/set_the_CATSvsDOGS_prj.py -i ~/ML/cats-vs-dogs

# #####
# SCRIPT 4 (SOLVER AND TRAINING AND LEARNING CURVE)
echo "TRAINING. Remember that: <Epoch_index = floor((iteration_index * batch_size) / (# data_samples))>"

#$CAFFE_TOOLS_DIR/bin/caffe.bin train --solver ./models/$NET/m$MOD_NUM/solver_$MOD_NUM\_$_NET.prototxt \
# 2>&1 | tee ./models/$NET/m$MOD_NUM/logfile_$MOD_NUM\_$_NET.log

python $WORK_DIR/code/4_training.py -s ./models/$NET/m$MOD_NUM/solver_$MOD_NUM\_$_NET.prototxt -l ./models/$NET/m$MOD_NUM/logfile_$MOD_NUM\_$_NET.log

# print image of CNN architecture
echo "PRINT CNN BLOCK DIAGRAM"
python $CAFFE_TOOLS_DIR/python/draw_net.py $WORK_DIR/models/$NET/m$MOD_NUM/train_val_$MOD_NUM\_$_NET.prototxt $WORK_DIR/models/$NET/m$MOD_NUM/bd_$MOD_NUM\_$_NET.png

# #####
# SCRIPT 5: plot the learning curve
echo "PLOT LEARNING CURVES"
python ./code/5_plot_learning_curve.py $WORK_DIR/models/$NET/m$MOD_NUM/logfile_$MOD_NUM\_$_NET.log $WORK_DIR/models/$NET/m$MOD_NUM/plr_train_val_$MOD_NUM\_$_NET.png

python ./code/plot_training_log.py 6 $WORK_DIR/models/$NET/m$MOD_NUM/plr_trainLoss_$MOD_NUM\_$_NET.png $WORK_DIR/models/$NET/m$MOD_NUM/logfile_$MOD_NUM\_$_NET.log
python ./code/plot_training_log.py 2 $WORK_DIR/models/$NET/m$MOD_NUM/plr_testLoss_$MOD_NUM\_$_NET.png $WORK_DIR/models/$NET/m$MOD_NUM/logfile_$MOD_NUM\_$_NET.log
python ./code/plot_training_log.py 0 $WORK_DIR/models/$NET/m$MOD_NUM/plr_testAccuracy_$MOD_NUM\_$_NET.png $WORK_DIR/models/$NET/m$MOD_NUM/logfile_$MOD_NUM\_$_NET.log

# #####
# SCRIPT 6 (PREDICTION)
echo "COMPUTE PREDICTIONS"
python ./code/6_make_predictions.py -d ./models/$NET/m$MOD_NUM/deploy_$MOD_NUM\_$_NET.prototxt -w ./models/$NET/m$MOD_NUM/snapshot_$MOD_NUM\_$_NET\_$_iter_$NUMIT.caffemodel 2>&1 | tee ./models/$NET/m$MOD_NUM/predictions_$MOD_NUM\_$_NET.txt

-:*** aws_caffe_flow_AlexNet.sh Top L47 (Shell-script[sh])
```

...you should get something like this:

```
ec2-user@ip-172-31-10-57:~/ML/cats-vs-dogs
(caffe_p27) [ec2-user@ip-172-31-10-57 caffe]$ cd ~/ML
(caffe_p27) [ec2-user@ip-172-31-10-57 ML]$ cd cats-vs-dogs/
(caffe_p27) [ec2-user@ip-172-31-10-57 cats-vs-dogs]$ ls -l *
-rw-rw-r-- 1 ec2-user ec2-user 2271  8 set 13.12 set_the_CATSvsDOGS_prj.py

caffe:
total 104
-rw-rw-r-- 1 ec2-user ec2-user 4185 10 set 17.11 aws_caffe_flow_AlexNet.sh
-rwxrwxr-x 1 ec2-user ec2-user 4442 10 set 16.22 caffe_flow_AlexNet.sh
drwxrwxr-x 2 ec2-user ec2-user 4096  8 set 13.03 code
drwxrwxr-x 3 ec2-user ec2-user 4096 10 set 16.13 models

deephi:
drwxrwxr-x 5 ec2-user ec2-user 4096 10 set 13.43 pruning
drwxrwxr-x 6 ec2-user ec2-user 4096 10 set 09.19 quantiz
drwxrwxr-x 5 ec2-user ec2-user 4096 31 ago 08.52 zcu102

input:
total 1552
drwxr-xr-x 5 ec2-user ec2-user 790528 10 set 16.37 jpg
drwxrwxr-x 4 ec2-user ec2-user 4096 10 set 16.34 lmbd
-rw-rw-r-- 1 ec2-user ec2-user 786446  8 set 13.31 mean.binaryproto
(caffe_p27) [ec2-user@ip-172-31-10-57 cats-vs-dogs]$ ls -l input/*
-rw-rw-r-- 1 ec2-user ec2-user 786446  8 set 13.31 input/mean.binaryproto

input/jpg:
total 12
drwxrwxr-x 4 ec2-user ec2-user 4096 10 set 16.33 calib
drwxrwxr-x 4 ec2-user ec2-user 4096 10 set 16.32 test
drwxrwxr-x 4 ec2-user ec2-user 4096 10 set 16.31 val

input/lmbd:
total 8
drwxrwxr-x 2 ec2-user ec2-user 4096 10 set 16.34 train_lmbd
drwxrwxr-x 2 ec2-user ec2-user 4096 10 set 16.35 valid_lmbd
(caffe_p27) [ec2-user@ip-172-31-10-57 cats-vs-dogs]$
```

replace