Search entries or author | Unread | Subscribe

Week 2 Discussion

Dave DeBarr

All Sections

27 27

Reply

Dave DeBarr
Tuesday

Setting up CUDA, CUDNN, and TensorFlow libraries at home: will double check that these commands work tomorrow ...

https://www.cross-entropy.net/ML530/setup.txt

Note: the version of TensorFlow that you want to use dictates the version of CUDA and CUDNN that you need to install:

https://www.tensorflow.org/install/gpu

Reply

Andrew Feldman
Wednesday

Hi Dave, which version of Tensorflow does the DSVM use? I saw that the Python version is 3.7

Reply

Andrew Feldman
Wednesday

I ask because tensorflow-gpu (as I understand it) was rolled into tensorflow for versions 2.x, which makes me think the DSVMs have v1.x. Asking because I set up my own environment with 2.x

Reply

Dave DeBarr
Wednesday

Let's see if replying via email creates a reply in the discussion [fingers crossed] ...

The tensorflow.keras stuff wasn't added until version 2.x.

To check, you can ...
import tensorflow
print(tensorflow.__version__)

That's two '_' characters on both sides of "version".

If you can't access it, I'll check tonight.

My goal for "pip install tensorflow-gpu" is to make sure we notice if we don't have the correct common unified device architecture (cuda) libraries available.

Search entries or author | Unread | Subscribe

Andrew Feldman posted a new comment on the thread Week 2 Discussion for MLEARN 530 A Sp 21: Deep Learning:

I ask because tensorflow-gpu (as I understand it) was rolled into tensorflow for versions 2.x, which makes me think the DSVMs have v1.x. Asking because I set up my own environment with 2.x

Andrew Feldman

Join the conversation using the link below, or comment by replying to this message. When allowed, if you need to include an attachment, please log in to Canvas and reply to the discussion.

CANVAS

Click here to join the conversation | Update your notification settings

Reply

Dave DeBarr
Wednesday

Here are the versions I get for python (3.7.9), tensorflow (2.3.1), and keras (2.4.0) ...

```
(py37_tensorflow) deeplearning@ML-RefVm-871628:~$ python --version
Python 3.7.9
(py37_tensorflow) deeplearning@ML-RefVm-871628:~$ python
Python 3.7.9 (default, Aug 31 2020, 12:42:55)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow
2021-04-15 03:01:58.282301: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dyn
>>> tensorflow.__version__
'2.3.1'
>>> tensorflow.keras.__version__
'2.4.0'
>>> exit()

(py37_tensorflow) deeplearning@ML-RefVm-871628:~$
```

The latest version of tensorflow is 2.4.1, but I think 2.3.1 is probably fine for this class.

Reply

Andrew Feldman
Wednesday

Hi Dave, I am seeing that this week's assignment takes a great bit of time - it has been running for the better part of the day for me.

I see this error message, could this be a sign that my GPU is not being utilized?

2021-04-14 12:42:30.307684: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libnvinfer.so.6'; dlerror: libnvinfer.so.6: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib64
2021-04-14 12:42:30.307783: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libnvinfer_plugin.so.6'; dlerror: libnvinfer_plugin.so.6: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib642021-04-14 12:42:30.307791: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:30] Cannot dlopen some TensorRT libraries. If you would like to use Nvidia GPU with

2021-04-14 12:42:31.099097: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1555] Found device 0 with properties:
pciBusID: 0000:17:00.0 name: GeForce RTX 2080 SUPER computeCapability: 7.5
coreClock: 1.845GHz coreCount: 48 deviceMemorySize: 7.79GiB deviceMemoryBandwidth: 462.00GiB/s
2021-04-14 12:42:31.099291: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcudart.so.10.1'; dlerror: libcudart.so.10.1: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib64
2021-04-14 12:42:31.099363: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcublas.so.10'; dlerror: libcublas.so.10: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib64
2021-04-14 12:42:31.099421: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcufft.so.10'; dlerror: libcufft.so.10: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib64
2021-04-14 12:42:31.099480: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcurand.so.10'; dlerror: libcurand.so.10: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib64
2021-04-14 12:42:31.099539: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcusolver.so.10'; dlerror: libcusolver.so.10: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib64
2021-04-14 12:42:31.099606: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'libcusparse.so.10'; dlerror: libcusparse.so.10: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH: /usr/local/cuda/extras/CUPTI/lib64:/usr/local/cuda/lib64:/usr/local/nvidia/lib:/usr/local/nvidia/lib64
2021-04-14 12:42:31.102281: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libcudnn.so.7
2021-04-14 12:42:31.102299: W tensorflow/core/common_runtime/gpu/gpu_device.cc:1592] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
2021-04-14 12:42:31.102481: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 AVX512F FMA
2021-04-14 12:42:31.127582: I tensorflow/core/platform/profile_utils/cpu_utils.cc:94] CPU Frequency: 3799900000 Hz
2021-04-14 12:42:31.128761: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x5ae1ad0 initialized for platform Host (this does not guarantee that XLA will be used). Devices:
2021-04-14 12:42:31.128794: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): Host, Default Version
2021-04-14 12:42:31.366996: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0x5acee10 initialized for platform CUDA (this does not guarantee that XLA will be used). Devices:
2021-04-14 12:42:31.367050: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): GeForce RTX 2080 SUPER, Compute Capability 7.5
2021-04-14 12:42:31.367239: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1096] Device interconnect StreamExecutor with strength 1 edge matrix:
2021-04-14 12:42:31.367254: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1102]
[Search space summary]
|-Default search space size: 11
> dense_depth (Int)
|-default: None

I ran nvidia-smi to check GPU utilization, and I see

```
Wed Apr 14 12:01:06 2021
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 435.21       Driver Version: 435.21       CUDA Version: 10.1     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  GeForce RTX 208...  Off  | 00000000:17:00.0 Off |                  N/A |
|  0%   39C    P8     3W / 250W |    146MiB /  7979MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
```

Search entries or author | Unread | Subscribe

```
|=============================================================================|
|    0      976      G   /usr/lib/xorg/Xorg                  14MiB |
|    0     1037      G   /usr/bin/gnome-shell                 4MiB |
|    0     6117      C   python                             115MiB |
+-----------------------------------------------------------------------------+
```

I have bolded the python process above, it suggests that my GPU is being utilized. Is 115MiB a good indication that the GPU is being used? I was hoping to see something more like a "% utilization" analogous to CPU utilization but I can't find that.

I am one of the people who let their DSVM hours expire, so I am running on my own machine with NVIDIA RTX 2080 Ti (as you can see above), Python 3.7, Tensorflow 2.x, CUDA 10.x (I set up my environment before you published the guidance :))

Any help on this is appreciated, thanks :)

Reply

Dave DeBarr
Wednesday

I'm thinking it should complete in under 30 minutes on the VM. Will ask Rudy about adding more time to your VM. Please remember to shut it off when not in use.

It's fine to see 0% gpu-util (utilization) every once in a while, but it should be fairly decent non-zero usage.

Are you running the gnome desktop on your machine, for your user interface?

Tensorflow typically allocates all available memory by default, so 115 megabytes is definitely less than the 8 gigabytes of memory for your 2080 card.

How much memory does your machine have? Is there any chance you've run out of memory, and it's swapping pages from memory to disk?

What does your output look like for "free -m"?

Which version of tensorflow are you using?

```
import tensorflow
print(tensorflow.__version__)
```

Reply

Dave DeBarr
Wednesday

Mea culpa (my bad): I ignored your error message. Yeah, you're not configured correctly.

Could not load dynamic library 'libnvinfer.so.6'; dlerror: libnvinfer.so.6: cannot open shared object file: No such file or directory; LD_LIBRARY_PATH:

Reply

Dave DeBarr
Wednesday

How did you install the cuda toolkit?

Reply

Dave DeBarr
Wednesday

I verified that it works:

https://www.cross-entropy.net/ML530/setup-console.txt

Reply

Andrew Feldman
Thursday

Thanks for the feedback. Let me try actually following your setup instructions, since it seems things were not quite right in my setup, and the Hyper-banding did take the better part of a day :)

Reply

Andrew Feldman
Wednesday

As we are starting to investigate a variety of DNN activation functions, I led me to recall malicious adversarial examples as a topic of DNN research https://towardsdatascience.com/breaking-neural-networks-with-adversarial-attacks-f4290a9a45aa . Back when I was considering whether to do a masters, I had reached out to a lab at my Alma Mater that was investigating the robustness of DNNs to malicious adversarial examples. Therefore I read a number of papers such as these

Robustness to adversarial examples for ReLu networks

https://arxiv.org/pdf/1804.09699

Robustness to adversarial examples for general activation functions (this came later, of course)

https://papers.nips.cc/paper/2018/file/d04863f100d59b3eb688a11f95b0ae60-Paper.pdf

It is an interesting concept, that malicious adversarial examples will typically succeed by choosing the smallest-possible image modification that leads the DNN to misclassify. Because small modifications will be noticeable to the DNN but not to the human eye. And so this lab developed a technique that can certify a DNN as being invulnerable to adversarial examples if the sum-of-squares magnitude of the image modification is less than a certain amount.

Reply

Dave DeBarr
Wednesday

For what it's worth: during the second to last class, we'll look at a couple of examples of generative adversarial networks: where we alternatively train a generative model to generate observations and a discriminative model to distinguish "real" observations from the "generated" observations (one batch for the discriminator; one batch for the generator; another batch for the discriminator; another batch for the generator; lather; rinse; repeat). Both will use crossentropy as the loss function, though the generator is only trained on the generated observations. We'll have lots more to say about this later.

Reply

Eric Kuo
Thursday

Hi Dave,

I keep getting a connection timeout error when I try to download the sample code. Any ideas what might be causing the problem?

(py37_tensorflow) deeplearning@ML-RefVm-871628:~$ wget https://www.cross-entropy.net/ML530/fashion-search.py.txt
--2021-04-15 15:55:15-- https://www.cross-entropy.net/ML530/fashion-search.py.txt
Resolving www.cross-entropy.net... 107.180.57.14

--2021-04-15 15:57:27-- (try: 2) https://www.cross-entropy.net/ML530/fashion-search.py.txt
Connecting to www.cross-entropy.net|107.180.57.14|:443... failed: Connection timed out.
Retrying.

--2021-04-15 15:59:39-- (try: 3) https://www.cross-entropy.net/ML530/fashion-search.py.txt
Connecting to www.cross-entropy.net|107.180.57.14|:443... failed: Connection timed out.
Retrying.

--2021-04-15 16:01:53-- (try: 4) https://www.cross-entropy.net/ML530/fashion-search.py.txt
Connecting to www.cross-entropy.net|107.180.57.14|:443... ^C

Eric

Edited by Eric Kuo on Apr 15 at 4:23pm

Reply

Dave DeBarr
Thursday

It looks like shared hosting across the continent is having trouble :([I believe our virtual machines are on the East coast; CoViD has made GPU machines a somewhat scarce resource]

I would try adding the "--timeout=2" option, so it fails quickly:

wget --timeout=2 https://www.cross-entropy.net/ML530/fashion-search.py.txt

Alternatively, you could download from https://canvas.uw.edu/courses/1432990/files/76180372/download?download_frd=1 (or https://www.cross-entropy.net/ML530/fashion-search.py.txt); then upload to your VM using ...

"C:\Program Files\PuTTY\pscp.exe" -P PortNumberGoesHere fashion-search.py.txt deeplearning@HostNameGoesHere:/home/deeplearning/fashion-search.py.txt

Reply (1 like)

Aeden Jameson
Thursday

Hi Dave,

Would you expand on your comments on how to properly compare models at least to the extent of where you might expect us beginners to be at? During class someone had commented "ROC", would also expand on your remarks there as well?

Edited by Aeden Jameson on Apr 15 at 1:21pm

Reply

Dave DeBarr
Thursday

Part 1 ...

On pages 222-223 (based on printed page numbers; numbers found at the top of the pages) of the 2nd edition of The Elements of Statistical Learning (https://web.stanford.edu/~hastie/ElemStatLearn/download.html), they write ...

<begin selective quotes>

Model selection: estimating the performance of different models in order to choose the best one.

Model assessment: having chosen a final model, estimating its prediction error (generalization error) on new data.

If we are in a data-rich situation, the best approach for both problems is to randomly divide the dataset into three parts: a training set, a validation set, and a test set. The training set is used to fit the models; the validation set is used to estimate prediction error for model selection; the test set is used for assessment of the generalization error of the final chosen model. Ideally, the test set should be kept in a "vault," and be brought out only at the end of the data analysis.

Search entries or author | Unread | Subscribe

The methods of this chapter approximate the validation step either analytically (AIC, BIC, MDL, SRM) or by efficient sample re-use (cross-validation and the bootstrap). Besides their use in model selection, we also examine to what extent each method provides a reliable estimate of test error of the final chosen model.

<end selective quotes>

I would strongly encourage you to avoid using a selection measure on the training data, even a penalized version of the measure [such as the Akaike Information Criterion (AIC; equation 7.27), the Bayesian Information Criterion (BIC; equation 7.35); etc].

If you have the resources and you want to repeatedly split the training data into training and validation sets, in order to perform cross validation for model seletion, then that's awesome! Suppose we have 11,000 data points. We could split the data into 9,000 for training, 1,000 for validation, and 1,000 for testing; but it's also possible for us to combine the training and validation together then run 10-fold cross validation [essentially repeating the training/validation process 10 times instead of once].

While it's possible you will run into a tiny data set where you may need to consider using one of the methods discussed in that chapter, I expect this to be a very uncommon circumstance.

Reply

Dave DeBarr
Thursday

Part 2 ...

Uh, can you remind what was said about the Receiver Operating Characteristic (ROC) curve? :)

I think it's fine to compare learning algorithms by comparing the Area Under the ROC (AUC) for some validation data; but keep in mind that, in practice, we often have to choose a classification threshold for the model. For example, let's suppose we can only mail out 10,000 offers. Then we want to maximize the response rate given the constraint that we can only mail out 10,000 offers.

P.S. It's possible for AUC to be used for a Precision-Recall (PR) curve as well.

Reply

Abhijit Majumdar
Thursday

Hello Dave,

For Machine Learning, I use t-tests to understand whether the performance of one model is statistically significant vis-à-vis the other. I run my cross validation through a certain number of epochs and then collect my metric (which I am using to assess the performance of my model via cross validation) before executing my t-test. Although I know this is not a part of our current syllabus, but would you be able to let me know if my approach is correct :-)? The code, which I use for my t-test, is as follows:

```
import scipy.stats as ss

import statsmodels.stats.weightstats as ws

def run_t_test(a, b, alpha=0.05, alternative='two-sided'):
    '''
    This function executes a student's t-test on the given set of observations.
    ----------
    :params: a - The first given set of observations.
    :params: b - The second given set of observations.
    :params: alpha - The significance level to be used. This defaults to 5% (indicating a 95% Confidence Interval).
    :params: alternative - Indicates whether the test is one-sided or two-sided. The default is two-sided.
    ----------
    Returns: a Pandas Series with the calculated t-statistics.
    '''
    diff = a.mean() - b.mean()
```

```
    degfree = means.dof_satt()
    index = ['DegFreedom', 'DifferenceInMeans', 'T-Statistic', 'PValue', 'Low95CI', 'High95CI']
    return pd.Series([degfree, diff, res[0], res[1], confint[0], confint[1]], index = index)
```

Thank you so much !

Reply

Dave DeBarr
Thursday

Why are you calling multiple functions?

Are you using weighted values for the vectors?

If I wanted a simple test of the means for two independent samples, I would vote for Welch's t-test. Here's a simple-ish implementation ...

```
import numpy as np
from scipy.stats import t
def welchs_t_test(vector1, vector2, confidence_level = 0.95):
    percentiles = [ (1.0 - confidence_level) / 2.0,  1.0 - (1.0 - confidence_level) / 2.0 ]
    means = [ np.mean(vector1), np.mean(vector2) ]
    difference_of_means = means[0] - means[1]
    vars = [ np.var(vector1, ddof = 1), np.var(vector2, ddof = 1) ]
    lens = [ len(vector1), len(vector2) ]
    var = vars[0] / lens[0] + vars[1] / lens[1]
    stderr = np.sqrt(var)
    test_statistic = difference_of_means / stderr
    df = (var**2) / (((vars[0] / lens[0])**2) / (lens[0] - 1) + ((vars[1] / lens[1])**2) / (lens[1] - 1))
    bounds = difference_of_means + [ stderr * t.ppf(percentiles[0], df = df), stderr * t.ppf(percentiles[1], df = df) ]
    pvalue = 2 * t.cdf(- np.abs(test_statistic), df = df)
    return [ pvalue, test_statistic, difference_of_means, bounds, df ]

x1 = [0]*105 + [1]*9895
x0 = [0]*169 + [1]*9831
welchs_t_test(x1, x0)
```

Reply

Dave DeBarr
Thursday

Reminder: if we can use the same test set to evaluate both models, I would vote for dependent sample testing instead: so we get a better estimate of uncertainty about the difference in means [this can either increase or decrease the variance, depending on the behavior of the models]

Reply

Abhijit Majumdar
Yesterday

Thank you so much Dave, this helped :-) !

Reply

Benjamin Johnson
Friday

Search entries or author | Unread | Subscribe

Dave DeBarr
Friday

Did you navigate to accept the invitation, using the link from the assignment?

Please navigate to the following URL to accept the invitation for this Kaggle task:

https://www.kaggle.com/t/90efb52a00be40849b3ccde825e6a4c9

Reply

Carl Gaither
Yesterday

Hi Dave,

My VM is used up - I didn't shutdown after the first assignment, but I know to do so now. Can I get more time on my VM?

Thanks,

Carl

Reply

Dave DeBarr
Yesterday

Hi Carl,

I sent email to Rudy. Hopefully, I'll hear from him soon.

Dave

Reply

Carl Gaither
5:48am

Thanks!

Reply