Week 9 Discussion

12 13

Dave DeBarr

All Sections

Search entries or author | Unread | Subscribe

Reply

Dave DeBarr

Jun 1, 2021

Here are links to the notes and console output for the graph convolution example ...

https://www.cross-entropy.net/ML530/ogbn-arxiv-notes.txt

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

Reply (1 like)

Andrew Feldman

Jun 3, 2021

Hello, I am running into an issue with the assignment 9 instructions. I will reproduce below. I am following all of the steps in the instructions one-by-one but the *python ptb-tensors.py.txt* step fails:

(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ python ptb-tensors.py.txt
Traceback (most recent call last):
File "ptb-tensors.py.txt", line 9, in <module>
for line in open(partition + ".dat", "r"):
FileNotFoundError: [Errno 2] No such file or directory: 'trn.dat'

Here is how my working directory (ptb-data) looks:

(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ ls
ptb-sentences.py.txt ptb-tensors.py.txt ptb-train.py.txt trn tst val

Here is how the **trn** directory looks:

(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ ls trn
0000.mrg 0116.mrg 0232.mrg 0348.mrg 0464.mrg 0580.mrg 0696.mrg 0812.mrg 0928.mrg 1044.mrg 1160.mrg 1276.mrg 1392.mrg 1508.mrg 1624.mrg 1740.mrg
0001.mrg 0117.mrg 0233.mrg 0349.mrg 0465.mrg 0581.mrg 0697.mrg 0813.mrg 0929.mrg 1045.mrg 1161.mrg 1277.mrg 1393.mrg 1509.mrg 1625.mrg 1741.mrg
0002.mrg 0118.mrg 0234.mrg 0350.mrg 0466.mrg 0582.mrg 0698.mrg 0814.mrg 0930.mrg 1046.mrg 1162.mrg 1278.mrg 1394.mrg 1510.mrg 1626.mrg 1742.mrg

.....

The directory contains many numbered **.mrg** files. Same story for the **tst** and **val** directories. I guessed that **I** had to merge these .mrg files into a single **trn.dat**, and then copy trn.dat into the ptb-data directory, which **I** tried doing using the commands below:

cd trn

cat *.mrg > trn.dat

cp trn.dat ../

cd ..

I repeated this process to create **tst.dat** and **val.dat** alongside **trn.dat** in the ptb-data directory, so my working directory looks like this:

(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ ls
ptb-sentences.py.txt ptb-tensors.py.txt ptb-train.py.txt trn trn.dat tst tst.dat val val.dat

However *python ptb-tensors.py.txt* still fails, after a long amount of time, with a different error:

python ptb-tensors.py.txt

```
Traceback (most recent call last):
  File "ptb-tensors.py.txt", line 32, in <module>
    npX[nextRow] = [ int(x) for x in X[i].split(" ") ]
  File "ptb-tensors.py.txt", line 32, in <listcomp>
```

npX[nextRow] = [int(x) for x in X[i].split(" ")]
ValueError: invalid literal for int() with base 10: ''

So I am wondering, can you please help me to follow the correct steps in deploying this assignment?

Thanks,

Andy

Reply

Dave DeBarr
Jun 3, 2021

Replying via email; hoping it posts correctly.

Looks like the instructions say to download ptb-sentences.py.txt, but does not say to use it.

You should use the following command before ptb-tensors.py.txt:
python ptb-sentences.py.txt

Will add the call to the ptb-sentences.py.txt to the instructions.

Dave

On Thu, Jun 3, 2021 at 3:25 PM Andrew Feldman <notifications@instructure.com> wrote:

> Andrew Feldman posted a new comment on the thread Week 9 Discussion for MLEARN 530 A Sp 21: Deep Learning:
>
> Hello, I am running into an issue with the assignment 9 instructions. I will reproduce below. I am following all of the steps in the instructions one-by-one but the *python ptb-tensors.py.txt* step fails:
>
> **(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ python ptb-tensors.py.txt**
> **Traceback (most recent call last):**
> **File "ptb-tensors.py.txt", line 9, in <module>**
> **for line in open(partition + ".dat", "r"):**
> **FileNotFoundError: [Errno 2] No such file or directory: 'trn.dat'**

Here is how my working directory (ptb-data) looks:

(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ ls
ptb-sentences.py.txt ptb-tensors.py.txt ptb-train.py.txt trn tst val

Here is how the **trn** directory looks:

(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ ls trn
0000.mrg 0116.mrg 0232.mrg 0348.mrg 0464.mrg 0580.mrg 0696.mrg 0812.mrg 0928.mrg 1044.mrg 1160.mrg 1276.mrg 1392.mrg 1508.mrg 1624.mrg 1740.mrg
0001.mrg 0117.mrg 0233.mrg 0349.mrg 0465.mrg 0581.mrg 0697.mrg 0813.mrg 0929.mrg 1045.mrg 1161.mrg 1277.mrg 1393.mrg 1509.mrg 1625.mrg 1741.mrg
0002.mrg 0118.mrg 0234.mrg 0350.mrg 0466.mrg 0582.mrg 0698.mrg 0814.mrg 0930.mrg 1046.mrg 1162.mrg 1278.mrg 1394.mrg 1510.mrg 1626.mrg 1742.mrg

.....

The directory contains many numbered **.mrg** files. Same story for the **tst** and **val** directories. I guessed that **I** had to merge these .mrg files into a single **trn.dat**, and then copy trn.dat into the ptb-data directory, which **I** tried doing using the commands below:

cd trn

cat *.mrg > trn.dat

cp trn.dat ../

cd ..

I repeated this process to create **tst.dat** and **val.dat** alongside **trn.dat** in the ptb-data directory, so my working directory looks like this:

(py37_tensorflow) deeplearning@ML-RefVm-871628:~/homework/assignment9/ptb-data$ ls
ptb-sentences.py.txt ptb-tensors.py.txt ptb-train.py.txt trn trn.dat tst tst.dat val val.dat

However *python ptb-tensors.py.txt* still fails, after a long amount of time, with a different error:

python ptb-tensors.py.txt

Traceback (most recent call last):
File "ptb-tensors.py.txt", line 32, in <module>
npX[nextRow] = [int(x) for x in X[i].split(" ")]
File "ptb-tensors.py.txt", line 32, in <listcomp>
npX[nextRow] = [int(x) for x in X[i].split(" ")]
ValueError: invalid literal for int() with base 10: "

So I am wondering, can you please help me to follow the correct steps in deploying this assignment?

Thanks,

Andy

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

Jun 4, 2021

A quick note about how the "BERT" ForSequenceClassification models work ...

After the sequence has been processed by the last layer of the transformer stack, the "ForSequenceClassification" models typically use only the embedding for the first position as input to the classification head.

For example, see
https://github.com/huggingface/transformers/blob/1f335aef3bb5382b5cfd7adbe5861ed4979dd98d/src/transformers/models/roberta/modeling_roberta.py#L1166

Reply

Clemente J Fortuna

Saturday

Has anyone succeeded in running ptb-train locally?

Reply

Dave DeBarr

Saturday

Yes :)

What error message(s) are you getting?

Reply

Dave DeBarr

Saturday

Any chance you're running out of memory?

If so, try reducing the number of features and/or the batch size.

Reply

Luis Garduno

Tuesday

I also tried but wasn't able to.

I ran into multiple errors starting with a bug in tensorflow with the latest version of numpy (https://github.com/tensorflow/models/issues/9706) to several others after making sure I was running the same versions of python (3.7), tensorflow (2.3.1) and numpy (1.18.5) as the lab's Azure VM.

The last issue I got was an error with tensorflow:

```
tensorflow.python.framework.errors_impl.InternalError:    Failed to call
ThenRnnForward with model config: [rnn_mode, rnn_input_mode,
rnn_direction_mode]: 2, 0, 0 , [num_layers, input_size, num_units, dir_count,
max_seq_length, batch_size, cell_num_units]: [1, 1024, 1024, 1, 250, 32,
1024]
     [[{{node cond/then/_0/cond/CudnnRNNV3}}]]
     [[functional_1/lstm/PartitionedCall]]
[Op:__inference_predict_function_79782]
```

Function call stack:
predict_function -> predict_function -> predict_function

Reply

Dave DeBarr
Tuesday

Which version of tensorflow are you using, and which version of the cuda toolkit are you using?

Does your cuda toolkit version match your tensorflow version?

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

python -c "import tensorflow; print(tensorflow.__version__)"

nvcc --version

Reply

Luis Garduno
Wednesday

Ah, thanks Dave, that might be it, I have Cuda 11.0 and Tensorflow 2.3.1 requires Cuda 10.1. I'll have to try with that and see if that fixes my issues.

I was able to run the assignment on the Azure VM, it's just my local setup that was giving me problems.

Reply

Abhijit Majumdar

Sunday

Good Morning Dave, may I ask a (perhaps rudimentary) question regarding creating tensors for any vision problems? Let us assume that the image file names are not symmetrical (for example, 00a821b.png, 01a98c2.png etc) and the files are zipped (similar to our exercises in Lessons 4 and 5), is there a way to read the image files into tensors without unzipping the files? If I unzip it I can use glob, but I would prefer not to unzip massive files :-).

I have a solution, but it takes a long time to run, as compared to the solution that we used in class or if I use glob :-). My code snippet is as follows (I am using our cifar10 dataset from Lesson 4 and I am just converting our train set for the sake of simplicity).

```
with ZipFile('ml530-2021-sp-cifar10.zip', 'r') as archive:
    trnX = np.zeros((45000, 32, 32, 3), dtype='float32')
    trn_idx = 0
    for j in archive.infolist():
        for i in range(trnX.shape[0]):
            if ('trn' in j.filename) and (j.filename.endswith('.png')):
                with archive.open(j.filename) as file:
                    img = Image.open(file)
                    trnX[i] = np.asarray(img)
            trn_idx += 1
```

Is there a more efficient way to do this? Thank you so much again !!

Reply

Dave DeBarr

Sunday

Maybe a list comprehension with .namelist() helps? For example ...

trn_png_list = [name for name in archive.namelist() if (("trn" in name) and name.endswith(".png"))]

Edited by Dave DeBarr on Jun 6 at 12:02pm

Reply

Abhijit Majumdar

Sunday

Thank you Dave :-) !

Reply