

# Week 2

SUBFORUMS
All
Assignment: Zombie Detector

← Week 2

L Ex 10 training is not converging

**Lou** Assignment: Zombie Detector · 5 months ago

Hi, in ex. 10 when runing the training loss is stagnating

```
batch 0 of 100, loss=1.4635149
batch 10 of 100, loss=1.3846332
batch 20 of 100, loss=1.3801727
batch 30 of 100, loss=1.4035373
batch 40 of 100, loss=1.3999285
batch 50 of 100, loss=1.4360347
batch 60 of 100, loss=1.4331912
batch 70 of 100, loss=1.4305105
batch 80 of 100, loss=1.3867929
batch 90 of 100, loss=1.4258771
Done fine-tuning!
```

If I debug the code below total\_loss is 0

```
10
11     Args:
12         image_list: A list of [1, height, width, 3] Tensor of type tf.float32.
```



```

13 Note that the height and width can vary across images, as they are
14 reshaped within this function to size 28x28x640.
15 groundtruth_boxes_list: A list of Tensors of shape [N_i, 4] with type
16 tf.float32 representing groundtruth boxes for each image in the batch.
17 groundtruth_classes_list: A list of Tensors of shape [N_i, num_classes]
18 with type tf.float32 representing groundtruth boxes for each image in
19 the batch.
20
21 Returns:
22     A scalar tensor representing the total loss for the input batch.
23
24 """
25
26 model.provide_groundtruth(
27     groundtruth_boxes_list=groundtruth_boxes_list,
28     groundtruth_classes_list=groundtruth_classes_list
29 )
30
31 with tf.GradientTape() as tape:
32
33     # ### START CODE HERE (Replace instances of `None` with your code) ###
34
35     preprocessed_image_list = []
36     true_shape_list = []
37
38     # Preprocess the images
39     for img in image_list:
40         processed_img, true_shape = model.preprocess(img)
41         print(true_shape)
42         preprocessed_image_list.append(processed_img)
43         true_shape_list.append(true_shape)
44
45     preprocessed_image_tensor = tf.concat(preprocessed_image_list, axis=0)
46     true_shape_tensor = tf.concat(true_shape_list, axis=0)
47
48     print(f"preprocessed_image_tensor shape: {preprocessed_image_tensor.shape}")
49     print(f"true_shape_tensor shape: {true_shape_tensor.shape}")
50

```

Would you have spotted a potential root cause?

↑ 1 Upvote

 Reply

Follow this discussion

**Earliest**

**Top**

### Most Recent

PC

**PIERRE CONTENCIN** · a month ago

Well I had the same problem and it was the checkpoint which was responsible your code for the `step_fn` is ok. As I couldn't figure out properly where the mistake was in my code I have taken the code from the tutorial on Colab with the duckie images on this and copied paste it (with the portability) at cell where I load the checkpoint and it went fine I still have to figure out why the `tmp_model` checkpoints that I tried by myself didn't work properly but the notebook could be improved by some more thorough tests to run to check the quality of the checkpoints. Because it went fine for me

0 Upvotes

 Reply



Nitin Kandpal · a month ago

you dont know but saved me and deserve a beer from me. Thanks

0 Upvotes

 Reply

C

**Christian** · 3 months ago

I recommend to review the tutorial on Colab with the duckie images. It's very similar.

0 Upvotes

TW

Thomas Woo · 4 months ago · Edited

if anyone is getting this problem, make sure you properly named your checkpoint  
args

issue i had: (note that i had an extra **underscore** character in the arg name from a bad copy/paste)

```
1 tmp_model_checkpoint = tf.train.Checkpoint(  
2     _feature_extractor = detection_model._feature_extractor,  
3     _box_predictor_ = tmp_box_predictor_checkpoint  
4 )
```

corrected:

```
1 tmp_model_checkpoint = tf.train.Checkpoint(  
2     _feature_extractor = detection_model._feature_extractor,  
3     _box_predictor = tmp_box_predictor_checkpoint  
4 )
```

0 Upvotes Reply

MB **Minn Bo Bo** · 5 months ago

Try to put model.provide\_groundtruth inside gradient tape just after the prediction and run again?

0 Upvotes Reply

L **Lou** · 5 months ago

Thanks ! What do you mean by re-running training: adding a loop in train\_step\_fn itself?

Would you like to share your code for train\_step\_fn?

0 Upvotes Reply

**Pavitra Shah** · 5 months ago

Keep re-running the training loop. I did that and after few loop it came down to 0.0005 for me.

0 Upvotes Hide 6 Replies

LV **Leo Vinzenz** · 5 months ago

I did that by increasing the batch size but it is a long way not a few loops for me:

0 Upvotes

LV **Leo Vinzenz** · 5 months ago

Actually, I found the problem earlier in the code when I didn't restore the checkpoint properly. I revisited all the steps before and found some errors leading to that problem. Take a look at exercise 6 and 7. Check if you did the following in Exercise 6. Note: I missed [...].expect\_partial() in the beginning, but it didn't throw an error

# Restore the checkpoint to the checkpoint path

```
checkpoint.restore(checkpoint_path).expect_partial()
```

1 Upvote

PM **Prakadeeswaran Manivannan** · 5 months ago

Hey Lio Vinzez even after doing that my loss is not converging any idea why?

0 Upvotes

**Juv Chan** · 5 months ago

What is the final loss you get before able to pass this assignment? I dont believe just a few loops.



0 Upvotes



SR **Shrayes Raman** · 4 months ago

Did you figure out solution?

0 Upvotes



**nitin bhardwaj** · 4 months ago

Please check if you are passing in Step 6 checkpoint path with ".index" extension. For me that was the error.

0 Upvotes

SD

Reply

**B** *I* U | [Link](#)

Reply

L

**Lou** · 5 months ago

In fact the true\_shape\_list already seems dubious, it seems to be stuffed with zeros. Any hint?

```
1 true_shape_list : [<tf.Tensor 'Preprocessor/stack_1:0'
2 shape=(1, 3) dtype=int32>, <tf.Tensor 'Preprocessor_1/stack_1:0'
3 shape=(1, 3) dtype=int32>, <tf.Tensor
4 'Preprocessor_2/stack_1:0' shape=(1, 3) dtype=int32>,
5 <tf.Tensor 'Preprocessor_3/stack_1:0' shape=(1, 3)
6 dtype=int32>]
7
```

0 Upvotes

Reply

SD

Reply

**B** *I* U | [Link](#)

Reply























































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































































