

Branch: master ▾

Find file

Copy path

[appliedai](#) / [SelfDrivingCar](#) / [train \(1\).py](#) / [Jump to ▾](#)

sunneysood Add files via upload

0470b4d 5 minutes ago

[1 contributor](#)

Raw Blame History



61 lines (46 sloc) 2.09 KB

● You're using code navigation to jump to definitions or references.

[Learn more](#) or [give us feedback](#)

```
1 import os
2 #import tensorflow as tf
3 from tensorflow.core.protobuf import saver_pb2
4 import driving_data
5 import model
6
7 #https://stackoverflow.com/questions/37383812/tensorflow-module-object-has-no-attribute-placeholder
8 import tensorflow.compat.v1 as tf
9 tf.disable_v2_behavior()
10
11 LOGDIR = './save'
12
13 sess = tf.InteractiveSession()
14
15 L2NormConst = 0.001
16
17 train_vars = tf.trainable_variables()
18
19
20 loss = tf.reduce_mean(tf.square(tf.subtract(model.y_, model.y))) + tf.add_n([tf.nn.l2_loss(v) for v in train_vars]) * L2NormConst
21 train_step = tf.train.AdamOptimizer(1e-4).minimize(loss)
22 sess.run(tf.initialize_all_variables())
23
24 # create a summary to monitor cost tensor
25 tf.summary.scalar("loss", loss)
26 # merge all summaries into a single op
27 merged_summary_op = tf.summary.merge_all()
28
29 saver = tf.train.Saver(write_version = saver_pb2.SaverDef.V1)
30
31 # op to write logs to Tensorboard
32 logs_path = './logs'
33 summary_writer = tf.summary.FileWriter(logs_path, graph=tf.get_default_graph())
34
35 epochs = 30
36 batch_size = 100
37
38 # train over the dataset about 30 times
39 for epoch in range(epochs):
40     for i in range(int(driving_data.num_images/batch_size)):
41         xs, ys = driving_data.LoadTrainBatch(batch_size)
42         train_step.run(feed_dict={model.x: xs, model.y_: ys, model.keep_prob: 0.5})
43         if i % 10 == 0:
44             xs, ys = driving_data.LoadValBatch(batch_size)
45             loss_value = loss.eval(feed_dict={model.x:xs, model.y_: ys, model.keep_prob: 0.5})
46             print("Epoch: %d, Step: %d, Loss: %g" % (epoch, epoch * batch_size + i, loss_value))
47
48     # write logs at every iteration
49     summary = merged_summary_op.eval(feed_dict={model.x:xs, model.y_: ys, model.keep_prob: 1.0})
50     summary_writer.add_summary(summary, epoch * driving_data.num_images/batch_size + i)
```

```
51
52     if i % batch_size == 0:
53         if not os.path.exists(LOGDIR):
54             os.makedirs(LOGDIR)
55         checkpoint_path = os.path.join(LOGDIR, "model.ckpt")
56         filename = saver.save(sess, checkpoint_path)
57         print("Model saved in file: %s" % filename)
58
59     print("Run the command line:\n" \
60           "--> tensorboard --logdir=./logs " \
61           "\nThen open http://0.0.0.0:6006/ into your web browser")
```