

```

1  # based on code from https://www.tensorflow.org/tutorials
2
3  import tensorflow as tf
4  import numpy as np
5
6  # specify path to training data and testing data
7
8  train_x_location = "mnist_x_train.csv"
9  train_y_location = "mnist_y_train.csv"
10 test_x_location = "mnist_x_test.csv"
11 test_y_location = "mnist_y_test.csv"
12
13 print("Reading training data")
14 x_train_2d = np.loadtxt(train_x_location, dtype="uint8", delimiter=",")
15 x_train_3d = x_train_2d.reshape(-1,28,28,1)
16 x_train = x_train_3d
17 y_train = np.loadtxt(train_y_location, dtype="uint8", delimiter=",")
18
19 print("Preprocessing x of training data")
20 x_train = x_train / 255.0
21
22 # define the training model
23 model = tf.keras.models.Sequential([
24     tf.keras.layers.Conv2D(
25         7,
26         (3,3),
27         padding='same',
28         activation=tf.nn.relu,
29         input_shape=(28,28,1)),
30     tf.keras.layers.MaxPool2D(2, 2),
31     tf.keras.layers.Flatten(),
32     tf.keras.layers.Dense(512, activation=tf.nn.relu),
33     tf.keras.layers.Dense(10, activation=tf.nn.softmax)
34 ])
35 model.compile(optimizer='adam',
36               loss='sparse_categorical_crossentropy',
37               metrics=['accuracy'])
38
39 print("train")
40 model.fit(x_train, y_train, epochs=5)
41
42 print("Reading testing data")
43 x_test_2d = np.loadtxt(test_x_location, dtype="uint8", delimiter=",")
44 x_test_3d = x_test_2d.reshape(-1,28,28,1)
45 x_test = x_test_3d
46 y_test = np.loadtxt(test_y_location, dtype="uint8", delimiter=",")
47
48 print("Preprocessing testing data")
49 x_test = x_test / 255.0
50
51 print("evaluate")
52 model.evaluate(x_test, y_test)

```