Term assignment 1

(Deep learning Methods and Applications)

2016025678 이강민

1. Code Description

```
learning_rate = 0.001
training_epochs = 20
batch_size = 128
```

- line 6: training epoch을 20으로 설정하였습니다.

```
    W1 = tf.get_variable(name="W1", shape=[3, 3, 3, 64], initializer=tf.contrib.layers.xavier_initializer())
    b1 = tf.get_variable(name="b1", shape=[64], initializer=tf.contrib.layers.xavier_initializer())
    c1 = tf.nn.conv2d(x_image, W1, strides=[1, 1, 1, 1], padding='SAME')
    l1 = tf.nn.relu(tf.nn.bias_add(c1, b1))
    l1_pool = tf.nn.max_pool(l1, ksize=[1, 3, 3, 1], strides=[1, 2, 2, 1], padding='SAME')

    W2 = tf.get_variable(name="W2", shape=[3, 3, 64, 128], initializer=tf.contrib.layers.xavier_initializer())
    b2 = tf.get_variable(name="b2", shape=[128], initializer = tf.contrib.layers.xavier_initializer())
    c2 = tf.nn.conv2d(l1_pool, W2, strides=[1,1,1,1], padding='SAME')
    l2 = tf.nn.relu(tf.nn.bias_add(c2, b2))
    l2_pool = tf.nn.max_pool(l2, ksize=[1,3,3,1], strides= [1,2,2,1], padding='SAME')

    W3 = tf.get_variable(name="W3", shape=[3, 3, 128, 256], initializer=tf.contrib.layers.xavier_initializer())
    b3 = tf.get_variable(name="b3", shape=[256], initializer = tf.contrib.layers.xavier_initializer())
    c3 = tf.nn.conv2d(l2_pool, W3, strides=[1,1,1,1], padding='SAME')
    l3 = tf.nn.relu(tf.nn.bias_add(c3,b3))
    l3_pool = tf.nn.max_pool(l3,ksize=[1,3,3,1], strides= [1,2,2,1], padding='SAME')

    W4 = tf.get_variable(name="W4", shape=[3, 3, 256, 512], initializer=tf.contrib.layers.xavier_initializer())
    b4 = tf.get_variable(name="b4", shape=[512], initializer = tf.contrib.layers.xavier_initializer())
    c4 = tf.nn.conv2d(l3_pool, W4, strides=[1,1,1,1], padding='SAME')
    l4 = tf.nn.relu(tf.nn.bias_add(c4,b4))
    l4_pool = tf.nn.max_pool(l4,ksize=[1,3,3,1], strides= [1,2,2,1], padding='SAME')

    W5 = tf.get_variable(name="W5", shape=[3,3, 512, 1024], initializer=tf.contrib.layers.xavier_initializer())
    b5 = tf.get_variable(name="b5", shape=[1024], initializer = tf.contrib.layers.xavier_initializer())
    c5 = tf.nn.conv2d(l4_pool, W5, strides=[1,1,1,1], padding='SAME')
    l5 = tf.nn.relu(tf.nn.bias_add(c5,b5))
    l5_pool = tf.nn.max_pool(l5,ksize=[1,3,3,1], strides= [1,2,2,1], padding='SAME')

    l5_flat = tf.reshape(l5_pool, [-1,1*1*1024])
```

-Convolution Neural Network 부분입니다.

-line 20~48: 총 5개의 layer로 구성되어있습니다. 필터의 개수는 layer가 깊어질 때마다 2배씩 늘려주었습니다. 각 layer마다 max_pooling을 해주었습니다.

-line 50: convolution layer의 마지막 output을 full connected에 넣어주기 위해 flating을 해줍니다.

```
52    W_fc1 = tf.get_variable(name="W_fc1", shape=[1*1*1024, 512], initializer=tf.contrib.layers.xavier_initializer())
53    b_fc1 = tf.get_variable(name="b_fc1", shape=[512], initializer=tf.contrib.layers.xavier_initializer())
54    l1_fc = tf.nn.relu(tf.nn.bias_add(tf.matmul(l5_flat, W_fc1),b_fc1))
55
56    W_fc2 = tf.get_variable(name="W_fc2", shape=[512,256], initializer=tf.contrib.layers.xavier_initializer())
57    b_fc2 = tf.get_variable(name="b_fc2", shape=[256], initializer=tf.contrib.layers.xavier_initializer())
58    l2_fc = tf.nn.relu(tf.nn.bias_add(tf.matmul(l1_fc,W_fc2), b_fc2))
59
60    W_fc3 = tf.get_variable(name="W_fc3", shape=[256,10], initializer=tf.contrib.layers.xavier_initializer())
61    b_fc3 = tf.get_variable(name="b_fc3", shape=[10], initializer=tf.contrib.layers.xavier_initializer())
62
63    logits = tf.nn.bias_add(tf.matmul(l2_fc, W_fc3), b_fc3)
64    hypothesis = tf.nn.softmax(logits)
65
66    return hypothesis, logits
```

-line 52~61: full connected layer를 3개의 layer로 구성하였습니다. 각 layer의 활성화 함수는 relu 함수를 사용하였습니다. 마지막 output을 출력하는 함수는 softmax함수를 사용하였습니다.

```
95 with tf.Session() as sess:
96     sess.run(tf.global_variables_initializer())
97     print("학습시작")
98
99     for epoch in range(training_epochs):
100        print("Epoch", epoch+1)
101        start = 0
102        shuffled_idx = np.arange(0, len(x_train))
103        np.random.shuffle(shuffled_idx)
104
105        for i in range(total_batch):
106            batch = batch_data(shuffled_idx, batch_size, x_train, y_train_one_hot.eval(), i*batch_size)
107            sess.run(train_step, feed_dict={x: batch[0], y: batch[1]})
108
109    saver = tf.train.Saver()
110    saver.save(sess, ckpt_path)
111    saver.restore(sess, ckpt_path)
112
113    y_prediction = np.argmax(y_pred.eval(feed_dict={x: x_dev}), 1)
114    y_true = np.argmax(y_dev_one_hot.eval(), 1)
115    dev_f1 = f1_score(y_true, y_prediction, average="weighted") # f1 스코어 측정
116    print("dev 데이터 f1 score: %f" % dev_f1)
```

-총 20번을 학습시키고 모델을 저장시킵니다. 저장된 모델을 불러와 dev data를 분류하고 dev data에 대한 f1 스코어를 측정합니다.

2.Result

```
학습시작
Epoch 1
2021-05-20 03:52:22.985270: I tensorflow/stream_executor/dsc
Epoch 2
Epoch 3
Epoch 4
Epoch 5
Epoch 6
Epoch 7
Epoch 8
Epoch 9
Epoch 10
Epoch 11
Epoch 12
Epoch 13
Epoch 14
Epoch 15
Epoch 16
Epoch 17
Epoch 18
Epoch 19
Epoch 20
WARNING:tensorflow:From /usr/local/lib/python3.7/dist-packag
Instructions for updating:
Use standard file APIs to check for files with this prefix.
dev 데이터 f1 score: 0.713192
```

```
학습시작
Epoch 1
2021-05-20 04:18:55.701835: I tensorflow/stream_executor/ds
Epoch 2
Epoch 3
Epoch 4
Epoch 5
Epoch 6
Epoch 7
Epoch 8
Epoch 9
Epoch 10
Epoch 11
Epoch 12
Epoch 13
Epoch 14
Epoch 15
Epoch 16
Epoch 17
Epoch 18
Epoch 19
Epoch 20
WARNING:tensorflow:From /usr/local/lib/python3.7/dist-packa
Instructions for updating:
Use standard file APIs to check for files with this prefix.
dev 데이터 f1 score: 0.712531
```

```
학습시작
Epoch 1
2021-05-20 05:51:07.537655: I tensorflow/stream_executor/ds
Epoch 2
Epoch 3
Epoch 4
Epoch 5
Epoch 6
Epoch 7
Epoch 8
Epoch 9
Epoch 10
Epoch 11
Epoch 12
Epoch 13
Epoch 14
Epoch 15
Epoch 16
Epoch 17
Epoch 18
Epoch 19
Epoch 20
WARNING:tensorflow:From /usr/local/lib/python3.7/dist-packa
Instructions for updating:
Use standard file APIs to check for files with this prefix.
dev 데이터 f1 score: 0.727559
```

```
학습시작
Epoch 1
2021-05-20 04:26:21.137575: I tensorflow/stream_executor/dso
Epoch 2
Epoch 3
Epoch 4
Epoch 5
Epoch 6
Epoch 7
Epoch 8
Epoch 9
Epoch 10
Epoch 11
Epoch 12
Epoch 13
Epoch 14
Epoch 15
Epoch 16
Epoch 17
Epoch 18
Epoch 19
Epoch 20
WARNING:tensorflow:From /usr/local/lib/python3.7/dist-packag
Instructions for updating:
Use standard file APIs to check for files with this prefix.
dev 데이터 f1 score: 0.710383
```

- dev 데이터에 대한 f1 score입니다.