

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

ОТЧЕТ

**Лабораторная работа №4**  
по курсу «Проектирование интеллектуальных систем»

Тема: «Сохранение модели и Tensorboard»

ИСПОЛНИТЕЛЬ:

группа ИУ5-22

Петрова Ирина

ФИО

подпись

"22" 04 2020 г.

ПРЕПОДАВАТЕЛЬ:

ФИО

подпись

" " 2020 г.

Москва - 2020

---

## Задание

Модифицировать программный код лабораторной №3 с добавлением сохранения модели и сохранения сводных статистик для визуализации Tensorboard.

Написать дополнительный код, который покажет демонстрацию восстановления модели из файла с расширением .ckpt.

## Выполнение

Сохранение модели и сохранение сводных статистик:

```
In [13]: with tf.Session() as sess:
    train_writer = tf.summary.FileWriter(LOG_DIR + '/train', graph = tf.get_default_graph())
    test_writer = tf.summary.FileWriter(LOG_DIR + '/test', graph = tf.get_default_graph())

    sess.run(tf.global_variables_initializer())
    start_time = time.time()

    for i in range(50000):
        batch = mnist.train.next_batch(50)
        if i % 100 == 0:
            summary, train_accuracy = sess.run([merged, accuracy], feed_dict = {x:batch[0], y:batch[1], keep_prob:1.0})
            print ("time {}, step {}, training accuracy {}".format(time.time() - start_time, i, train_accuracy))
            # Add to summaries
            train_writer.add_summary(summary, i)
            sess.run(train_step, feed_dict = {x:batch[0], y:batch[1], keep_prob:0.5})

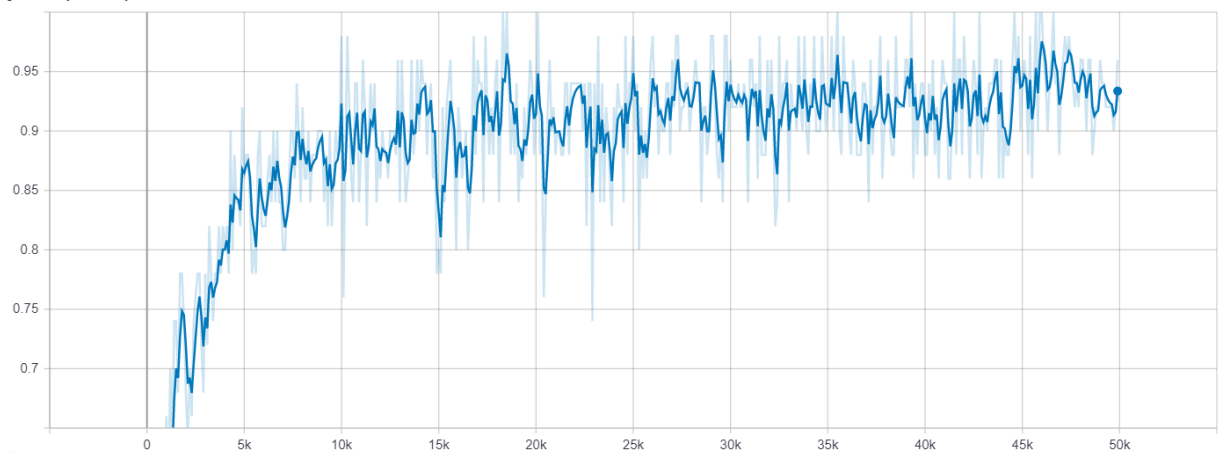
    saver.save(sess, os.path.join(DIR, "model_ckpt"), global_step = i)

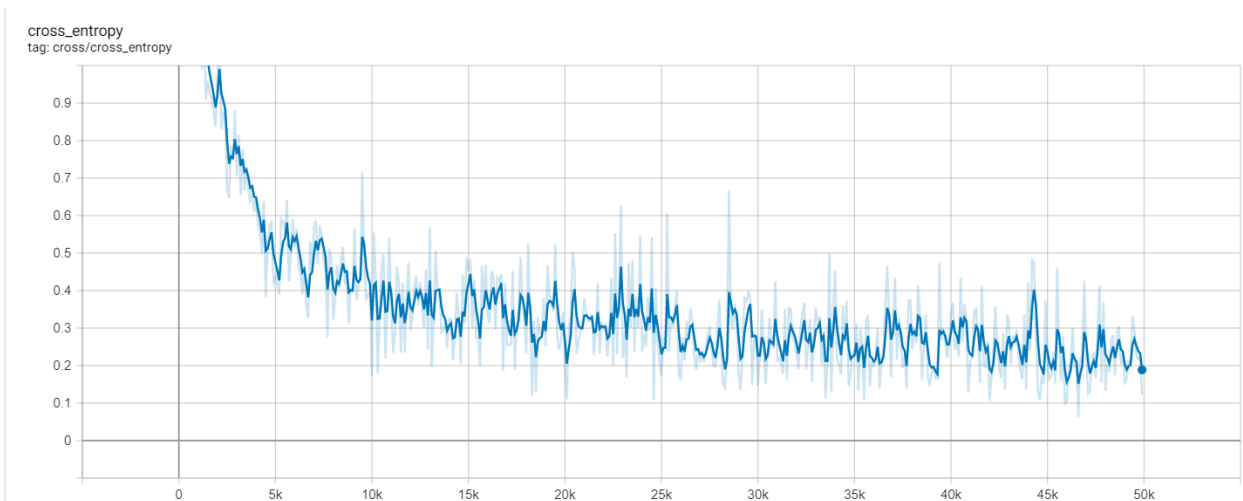
    X = mnist.test.images.reshape(10, 1000, 784)
    Y = mnist.test.labels.reshape(10, 1000, 10)

    #summary, merged
    test_accuracy = np.mean([sess.run([accuracy], feed_dict = {x:X[i], y:Y[i], keep_prob:1.0}) for i in range(10)])
    test_writer.add_summary(summary, i)
    print ("test accuracy: {}".format(test_accuracy))
```

```
time 3928.761617898941, step 48200, training accuracy 0.9399999976158142
time 3935.2627563476562, step 48300, training accuracy 0.8999999761581421
time 3941.6423106193542, step 48400, training accuracy 0.9599999785423279
time 3948.1620948314667, step 48500, training accuracy 0.9599999785423279
time 3954.6418483257294, step 48600, training accuracy 0.8799999952316284
time 3961.130725622177, step 48700, training accuracy 0.8999999761581421
time 3967.6221702098846, step 48800, training accuracy 0.9200000166893005
time 3974.112881422043, step 48900, training accuracy 0.9200000166893005
time 3980.6270151138306, step 49000, training accuracy 0.9599999785423279
time 3987.2698204517365, step 49100, training accuracy 0.9399999976158142
time 3993.9589717388153, step 49200, training accuracy 0.9399999976158142
time 4000.655799627304, step 49300, training accuracy 0.9200000166893005
time 4007.278669834137, step 49400, training accuracy 0.9200000166893005
time 4014.113363981247, step 49500, training accuracy 0.9200000166893005
time 4020.84037566185, step 49600, training accuracy 0.9200000166893005
time 4027.457731962204, step 49700, training accuracy 0.8999999761581421
time 4034.004760503769, step 49800, training accuracy 0.9200000166893005
time 4040.7838048934937, step 49900, training accuracy 0.9599999785423279
test accuracy: 0.9359999895095825
```

accuracy  
tag: accuracy/accuracy





Восстановление модели из файла с расширением .ckpt:

```
In [18]: 1 tf.reset_default_graph()
2 with tf.Session() as sess:
3     sess.run(tf.global_variables_initializer())
4     start_time = time.time()
5     saver = tf.train.import_meta_graph(os.path.join(DIR, "model_ckpt-9.meta"))
6     saver.restore(sess, os.path.join(DIR, "model_ckpt-49999"))
7     x = tf.get_collection('train_var')[0]
8     y_ = tf.get_collection('train_var')[1]
9     keep_prob = tf.get_collection('train_var')[2]
10    accuracy = tf.get_collection('train_var')[3]
11
12
13    test_accuracy = np.mean([sess.run([accuracy], feed_dict = {x:X[i], y_:Y[i], keep_prob:1.0}) for i in range(10)])
14    print("test accuracy: {}".format(test_accuracy))
```

INFO:tensorflow:Restoring parameters from data\model\_ckpt-49999  
test accuracy: 0.9359999895095825

## Контрольные вопросы

1. Как включить TensorBoard?  
В терминале выполнить команду `tensorboard --logdir=path/to/log-directory`.  
TensorBoard работает по адресу `http://localhost:6006`.
2. Как сбросить граф?  
Выполнить команду `tf.reset_default_graph()`.
3. Зачем нужны коллекции?  
Коллекция – это объект, похожий на словарь, внутри которого хранятся элементы узлов графа.
4. Перечислите команды для добавления переменных в сводную статистику.  
`tf.summary.scalar('тег', значение)`  
`tf.summary.histogram('histogram', переменная)`  
`tf.summary.merge_all()`

## Список литературы

1. Google. Tensorflow. 2018. Apr. url - [https://www.tensorflow.org/api\\_docs/python/tf/train/Saver](https://www.tensorflow.org/api_docs/python/tf/train/Saver).
2. Google. TensorBoard. 2018. Apr. url - [https://www.tensorflow.org/programmers\\_guide/summaries\\_and\\_tensorboard](https://www.tensorflow.org/programmers_guide/summaries_and_tensorboard).