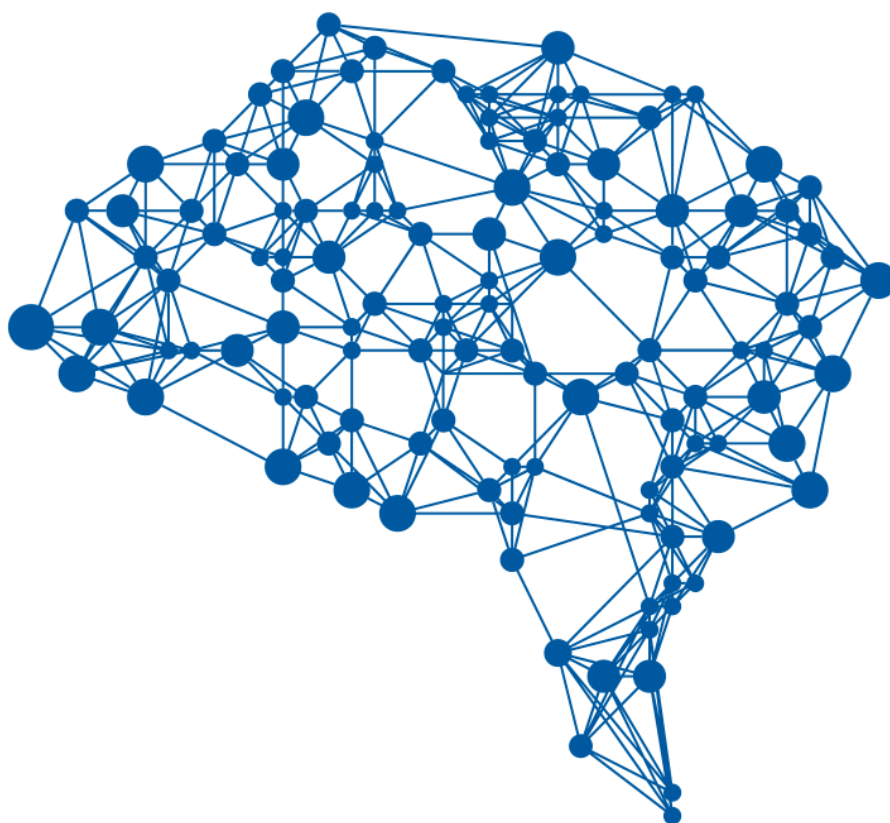


# Deep Learning Homework 4



安捷 1601210097  
2017 年 4 月 22 日

## 1 算法实现简介

在这次作业中，我使用 tensorflow github 中的 cifar10 cnn 代码，进行了一定的修改和调参，希望可以使得算法达到最好的性能。

## 2 数值实验结果

我做了四组数值实验，下面分别将结果展示如下：

### 2.1 2conv-32-64,ite=50000

2017-04-21 12:39:10.283589: step 49950, loss = 0.76 (1050.7 examples/sec; 0.122 sec/batch)  
 2017-04-21 12:39:11.515913: step 49960, loss = 0.75 (1038.7 examples/sec; 0.123 sec/batch)  
 2017-04-21 12:39:12.732593: step 49970, loss = 0.58 (1052.0 examples/sec; 0.122 sec/batch)  
 2017-04-21 12:39:13.975698: step 49980, loss = 0.59 (1029.7 examples/sec; 0.124 sec/batch)  
 2017-04-21 12:39:15.223052: step 49990, loss = 0.68 (1026.2 examples/sec; 0.125 sec/batch)  
 2017-04-21 12:57:15.110323: precision @ 1 = 0.854

### 2.2 2conv-32-64,ite=438860

2017-04-21 10:49:53.340448: step 438820, loss = 0.12 (1148.0 examples/sec; 0.111 sec/batch)  
 2017-04-21 10:49:54.463862: step 438830, loss = 0.11 (1139.4 examples/sec; 0.112 sec/batch)  
 2017-04-21 10:49:55.602834: step 438840, loss = 0.11 (1123.8 examples/sec; 0.114 sec/batch)  
 2017-04-21 10:49:56.711436: step 438850, loss = 0.11 (1154.6 examples/sec; 0.111 sec/batch)  
 2017-04-21 10:49:57.837942: step 438860, loss = 0.11 (1136.3 examples/sec; 0.113 sec/batch)  
 2017-04-21 10:50:14.644544: precision @ 1 = 0.870

### 2.3 2conv-32-32-64,ite=50000

2017-04-21 20:23:15.227485: step 49950, loss = 0.77 (1342.8 examples/sec; 0.095 sec/batch)  
 2017-04-21 20:23:16.157790: step 49960, loss = 0.50 (1375.9 examples/sec; 0.093 sec/batch)  
 2017-04-21 20:23:17.112762: step 49970, loss = 0.63 (1340.3 examples/sec; 0.095 sec/batch)  
 2017-04-21 20:23:18.056657: step 49980, loss = 0.62 (1356.1 examples/sec; 0.094 sec/batch)  
 2017-04-21 20:23:19.001197: step 49990, loss = 0.66 (1355.1 examples/sec; 0.094 sec/batch)  
 2017-04-21 21:06:45.512474: precision @ 1 = 0.834

## 2.4 2conv-32-32-64,ite=200000

2017-04-22 02:35:04.625261: step 199950, loss = 0.32 (1304.0 examples/sec; 0.098 sec/batch) 2017-04-22 02:35:05.628415: step 199960, loss = 0.40 (1276.0 examples/sec; 0.100 sec/batch) 2017-04-22 02:35:06.630871: step 199970, loss = 0.27 (1276.9 examples/sec; 0.100 sec/batch) 2017-04-22 02:35:07.636670: step 199980, loss = 0.30 (1272.6 examples/sec; 0.101 sec/batch) 2017-04-22 02:35:08.615497: step 199990, loss = 0.27 (1307.7 examples/sec; 0.098 sec/batch)

2017-04-22 11:42:12.524210: precision @ 1 = 0.853

## 3 代码运行环境及测试平台信息

---

Python Version: 3.6.0  
 Tensorflow Version: tensorflow-gpu-1.0.1  
 CUDA Version: 8.0  
 OS: Arch Linux  
 Kernel: x86\_64 Linux 4.10.4-1-ARCH  
 CPU: Intel Core i7-6700K @ 8x 4.2GHz  
 GPU: GeForce GTX 1060 6GB  
 RAM: 16003MiB

---

表 1: 代码运行环境及测试环境表

在没有 NVIDIA GPU 及 CUDA 支持的环境下代码依然可以运行，只是速度较慢

## 4 总结

针对 cifar10 数据集, cnn 的合适规模大约是两个卷积层, 两层卷积层在 500000 次左右迭代的效果最好, 过大的规模会降低测试集准确度。