

week5 & 6 Experiment

1. Week5

- 运行环境：

 **NVIDIA GeForce RTX 2080 Ti**
ID: q8NxK6

● 运行中 [停止并释放](#) [更多](#)

● 长时间运行推荐使用SSH。该链接非长期有效链接，偶尔会因网络问题而变更，请留意邮件通知。

SSH 链接: [ssh -p 28359 root@hz-t3.matpool.com](ssh-p-28359-root@hz-t3.matpool.com) [复制密码](#) [重置密码](#)

JupyterLab 链接: <https://hz.matpool.com:27773?token=Wa3VglVxGA>

硬件信息

GPU: NVIDIA GeForce RTX 2080 Ti
每秒浮点运算次数: 13.13 TFLOPS
显卡内存: 11 GB

租用配置

镜像: Pytorch 1.6.0
镜像描述: 预装: Python 3.7, CUDA 10.2,...
挂载: /mnt

计费: ¥0.74 +

折扣价: ¥ 3.00/小时
原价: ¥ 5.99/小时
余额还够租用: ~ 3小时

- TranseE运行：
 - 数据集：“WN18RR”

2. 代码调整

2.1 Transe

```

1 def set_interact_args():
2     parser = argparse.ArgumentParser()
3     parser.add_argument('--margin', default=5.0,
4         type=float, required=False, help='Margin loss中
5         margin值')
6     parser.add_argument('--nbatches', default=100,
7         type=int, required=False, help='Batch size')
8     parser.add_argument('--dim', default=100,
9         type=int, required=False, help='Embedding size')
10    parser.add_argument('--p_norm', default=1,
11        type=int, required=False, help='能量函数为1范数形式')
12    parser.add_argument('--train_times',
13        default=1, type=int, required=False, help='epoch-
14        训练轮次')
15    parser.add_argument('--alpha', default=1,
16        type=float, required=False, help='学习率')
17    return parser.parse_args()

```

2.1.1 原始实验:

```

1 #脚本命令
2 Date=`date +%y%m%d`
3 echo "1.sh back begin at `date +%H:%M:%S`" >>
4 out.log
5 nohup python -u train_transe_FB15K237.py --
6 margin=5 --nbatches=100 --dim=100 --p_norm=1 --
7 train_times=1000 --alpha=1 >
8 logs/`./result/transe/transe(5,100,100,1,1000,1).l
9 og`
10 echo "1.sh back end at `date +%H:%M:%S`" >>
11 out.log

```

- 效果呈现:

```

2007 Epoch 997 | loss: 0.207091: 100% | 997/1000 [10:49<00:01, 1.56it/s]
2008 Epoch 998 | loss: 0.213130: 100% | 998/1000 [10:49<00:01, 1.56it/s]
2009 Epoch 998 | loss: 0.213130: 100% | 999/1000 [10:49<00:00, 1.56it/s]
2010 Epoch 999 | loss: 0.210327: 100% | 999/1000 [10:50<00:00, 1.56it/s]
2011 Epoch 999 | loss: 0.210327: 100% | 1000/1000 [10:50<00:00, 1.56it/s]
2012 Epoch 999 | loss: 0.210327: 100% | 1000/1000 [10:50<00:00, 1.54it/s]
2013

```

```

2086 100% | 3120/3134 [00:07<00:00, 452.21it/s]
2087 100% | 3134/3134 [00:07<00:00, 406.18it/s]
2088 no type constraint results:
2089 metric:      MRR      MR      hit@10      hit@3      hit@1
2090 l(raw):      0.138370      5158.203613      0.419272      0.226548      0.000319
2091 r(raw):      0.159494      3496.497559      0.477026      0.247926      0.006063
2092 averaged(raw):      0.148932      4327.350586      0.448149      0.237237      0.003191
2093
2094 l(filter):    0.193350      5134.768555      0.451181      0.367581      0.001276
2095 r(filter):    0.210217      3491.129883      0.493618      0.384812      0.010530
2096 averaged(filter): 0.201783      4312.949219      0.472399      0.376197      0.005903
2097 0.472399
2098 0.4723994731903076

```

2.1.2 改变margin为4

```

1 Date=`date +%y%m%d`
2 echo "2.sh back begin at `date +%H:%M:%S`" >>
  out.log
3 nohup python -u train_transe_FB15K237.py --
  margin=4 --nbatches=100 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=1 >
  logs/`./result/transe/transe(4,100,100,1,1000,1).1
  og`
4 echo "2.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2006 Epoch 997 | loss: 0.065870: 100% | 997/1000 [10:54<00:01, 1.52it/s]
2007 Epoch 997 | loss: 0.065870: 100% | 998/1000 [10:54<00:01, 1.52it/s]
2008 Epoch 998 | loss: 0.069522: 100% | 998/1000 [10:55<00:01, 1.52it/s]
2009 Epoch 998 | loss: 0.069522: 100% | 999/1000 [10:55<00:00, 1.53it/s]
2010 Epoch 999 | loss: 0.068871: 100% | 999/1000 [10:56<00:00, 1.53it/s]
2011 Epoch 999 | loss: 0.068871: 100% | 1000/1000 [10:56<00:00, 1.54it/s]
2012 Epoch 999 | loss: 0.068871: 100% | 1000/1000 [10:56<00:00, 1.52it/s]

```

```

2088 100%|██████████| 3134/3134 [00:07<00:00, 395.70it/s]
2089 no type constraint results:
2090 metric:      MRR      MR      hit@10      hit@3      hit@1
2091 l(raw):      0.133139    5983.143555    0.407467    0.220485    0.000000
2092 r(raw):      0.149238    3621.848145    0.458519    0.233567    0.004467
2093 averaged(raw): 0.141189    4802.496094    0.432993    0.227026    0.002234
2094
2095 l(filter):    0.189913    5959.667969    0.434269    0.366305    0.001276
2096 r(filter):    0.199787    3616.461670    0.471602    0.375877    0.005105
2097 averaged(filter): 0.194850    4788.064941    0.452936    0.371091    0.003191
2098 0.452936
2099 0.4529355466365814

```

- 虽然loss降得很低，但是测试结果却发生了一定下降，所以该参数条件下造成了过拟合

2.1.3 改变p_norm为2

```

2004 Epoch 996 | loss: 400.414801: 100%|██████████| 996/1000 [10:56<00:02, 1.58it/s]
2005 Epoch 996 | loss: 400.414801: 100%|██████████| 997/1000 [10:56<00:01, 1.55it/s]
2006 Epoch 997 | loss: 400.444719: 100%|██████████| 997/1000 [10:56<00:01, 1.55it/s]
2007 Epoch 997 | loss: 400.444719: 100%|██████████| 998/1000 [10:56<00:01, 1.54it/s]
2008 Epoch 998 | loss: 400.467891: 100%|██████████| 998/1000 [10:57<00:01, 1.54it/s]
2009 Epoch 998 | loss: 400.467891: 100%|██████████| 999/1000 [10:57<00:00, 1.53it/s]
2010 Epoch 999 | loss: 400.370608: 100%|██████████| 999/1000 [10:58<00:00, 1.53it/s]
2011 Epoch 999 | loss: 400.370608: 100%|██████████| 1000/1000 [10:58<00:00, 1.51it/s]
2012 Epoch 999 | loss: 400.370608: 100%|██████████| 1000/1000 [10:58<00:00, 1.52it/s]

```

```

2095 99%|██████████| 3118/3134 [00:08<00:00, 468.29it/s]
2096 100%|██████████| 3134/3134 [00:08<00:00, 367.20it/s]
2097 no type constraint results:
2098 metric:      MRR      MR      hit@10      hit@3      hit@1
2099 l(raw):      0.028054    6113.328613    0.057435    0.029036    0.012125
2100 r(raw):      0.064897    4102.169922    0.127632    0.066369    0.034142
2101 averaged(raw): 0.046476    5107.749023    0.092534    0.047703    0.023133
2102
2103 l(filter):    0.042108    6095.167969    0.073389    0.043714    0.024888
2104 r(filter):    0.069423    4098.048340    0.130504    0.069241    0.039566
2105 averaged(filter): 0.055766    5096.608398    0.101946    0.056477    0.032227
2106 0.101946
2107 0.1019463986158371

```

- 用 l_2 范数发现效果显著下降，这说明 l_2 范数明显不适合此次实验

2.1.4 改变维数为200

```

1 Date=`date +%y%m%d`
2 echo "5.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transe_FB15K237.py --
  margin=5 --nbatches=200 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=1 >
  logs/`./result/transe/transe(5,200,100,1,1000,1).1
  og`
5 echo "5.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2005 Epoch 996 | loss: 0.051217: 100% | ██████████ | 997/1000 [13:39<00:02, 1.22it/s]
2006 Epoch 997 | loss: 0.048420: 100% | ██████████ | 997/1000 [13:39<00:02, 1.22it/s]
2007 Epoch 997 | loss: 0.048420: 100% | ██████████ | 998/1000 [13:39<00:01, 1.22it/s]
2008 Epoch 998 | loss: 0.053169: 100% | ██████████ | 998/1000 [13:40<00:01, 1.22it/s]
2009 Epoch 998 | loss: 0.053169: 100% | ██████████ | 999/1000 [13:40<00:00, 1.22it/s]
2010 Epoch 999 | loss: 0.053974: 100% | ██████████ | 999/1000 [13:41<00:00, 1.22it/s]
2011 Epoch 999 | loss: 0.053974: 100% | ██████████ | 1000/1000 [13:41<00:00, 1.22it/s]
2012 Epoch 999 | loss: 0.053974: 100% | ██████████ | 1000/1000 [13:41<00:00, 1.22it/s]
2013

```

```

2101 97% | ██████████ | 3036/3134 [00:09<00:00, 339.45it/s]
2102 98% | ██████████ | 3072/3134 [00:09<00:00, 343.01it/s]
2103 99% | ██████████ | 3116/3134 [00:09<00:00, 368.96it/s]
2104 100% | ██████████ | 3134/3134 [00:09<00:00, 326.01it/s]
2105 no type constraint results:
2106 metric:      MRR      MR      hit@10      hit@3      hit@1
2107 l(raw):      0.131014    6682.884277  0.408424    0.216018    0.000000
2108 r(raw):      0.147385    3212.349121  0.454052    0.228462    0.004148
2109 averaged(raw): 0.139199    4947.616699  0.431238    0.222240    0.002074
2110
2111 l(filter):    0.189673    6659.300781  0.434588    0.365029    0.001595
2112 r(filter):    0.200265    3206.929443  0.468092    0.376197    0.004786
2113 averaged(filter): 0.194969    4933.115234  0.451340    0.370613    0.003191
2114 0.451340
2115 0.4513401389122009

```

- 200dim效果并不好，且增加计算开销，所以200 dim不适合

2.1.5 改变n_batches为200

```

1 Date=`date +%y%m%d`
2 echo "5.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transe_FB15K237.py --
  margin=5 --nbatches=200 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=1 >
  logs/`./result/transe/transe(5,200,100,1,1000,1).l
  og`
5 echo "5.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2005 Epoch 996 | loss: 0.441036: 100%|██████████| 997/1000 [17:54<00:03, 1.09s/it]
2006 Epoch 997 | loss: 0.441166: 100%|██████████| 997/1000 [17:55<00:03, 1.09s/it]
2007 Epoch 997 | loss: 0.441166: 100%|██████████| 998/1000 [17:55<00:02, 1.08s/it]
2008 Epoch 998 | loss: 0.441047: 100%|██████████| 998/1000 [17:56<00:02, 1.08s/it]
2009 Epoch 998 | loss: 0.441047: 100%|██████████| 999/1000 [17:56<00:01, 1.08s/it]
2010 Epoch 999 | loss: 0.437477: 100%|██████████| 999/1000 [17:57<00:01, 1.08s/it]
2011 Epoch 999 | loss: 0.437477: 100%|██████████| 1000/1000 [17:57<00:00, 1.09s/it]
2012 Epoch 999 | loss: 0.437477: 100%|██████████| 1000/1000 [17:57<00:00, 1.08s/it]
2013

```

```

2095 100%|██████████| 3122/3134 [00:08<00:00, 393.27it/s]
2096 100%|██████████| 3134/3134 [00:08<00:00, 361.24it/s]
2097 no type constraint results:
2098 metric:      MRR      MR      hit@10    hit@3    hit@1
2099 l(raw):      0.138690   5034.560547  0.421506  0.228143  0.000319
2100 r(raw):      0.159422   3353.724365  0.475431  0.248883  0.007658
2101 averaged(raw): 0.149056   4194.142578  0.448468  0.238513  0.003989
2102
2103 l(filter):    0.195667   5011.101562  0.451819  0.372048  0.002553
2104 r(filter):    0.209843   3348.364746  0.488832  0.380664  0.012125
2105 averaged(filter): 0.202755   4179.733398  0.470325  0.376356  0.007339
2106 0.470325
2107 0.4703254699707031

```

- 200 batch效果有所下降

2.1.6 学习率调为0.5

```

1 Date=`date +%y%m%d`
2 echo "6.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transe_FB15K237.py --
  margin=5 --nbatches=100 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=0.5 >
  logs/`./result/transe/transe(5,100,100,1,1000,0.5)
  .log`
5 echo "6.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2003 Epoch 995 | loss: 0.208036: 100%|██████████| 996/1000 [10:12<00:02, 1.64it/s]
2004 Epoch 996 | loss: 0.205459: 100%|██████████| 996/1000 [10:13<00:02, 1.64it/s]
2005 Epoch 996 | loss: 0.205459: 100%|██████████| 997/1000 [10:13<00:01, 1.62it/s]
2006 Epoch 997 | loss: 0.201967: 100%|██████████| 997/1000 [10:14<00:01, 1.62it/s]
2007 Epoch 997 | loss: 0.201967: 100%|██████████| 998/1000 [10:14<00:01, 1.60it/s]
2008 Epoch 998 | loss: 0.209225: 100%|██████████| 998/1000 [10:14<00:01, 1.60it/s]
2009 Epoch 998 | loss: 0.209225: 100%|██████████| 999/1000 [10:14<00:00, 1.61it/s]
2010 Epoch 999 | loss: 0.207769: 100%|██████████| 999/1000 [10:15<00:00, 1.61it/s]
2011 Epoch 999 | loss: 0.207769: 100%|██████████| 1000/1000 [10:15<00:00, 1.63it/s]
2012 Epoch 999 | loss: 0.207769: 100%|██████████| 1000/1000 [10:15<00:00, 1.63it/s]

```

```

2087 | 99%|██████████| 3103/3134 [00:07<00:00, 394.64it/s]
2088 |100%|██████████| 3134/3134 [00:07<00:00, 398.11it/s]
2089 no type constraint results:
2090 metric:      MRR      MR      hit@10      hit@3      hit@1
2091 l(raw):      0.139823    5140.723145    0.421187    0.230696    0.000000
2092 r(raw):      0.158415    3528.097412    0.479579    0.250479    0.006063
2093 averaged(raw): 0.149119    4334.410156    0.450383    0.240587    0.003031
2094
2095 l(filter):    0.196255    5117.284180    0.454690    0.370134    0.002872
2096 r(filter):    0.210339    3522.729736    0.494576    0.385769    0.011168
2097 averaged(filter): 0.203297    4320.006836    0.474633    0.377952    0.007020
2098 0.474633
2099 0.47463303804397583

```

- 学习率降低后，效果有一定上升

2.1.7 epoch调维500


```

1 Date=`date +%y%m%d`
2 echo "7.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transe_FB15K237.py --
  margin=5 --nbatches=100 --dim=100 --p_norm=1 --
  train_times=500 --alpha=1 >
  logs/`./result/transe/transe(5,100,100,1,500,1).lo
  g`
5 echo "7.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

1005 Epoch 496 | loss: 0.265535: 99%| ██████████ | 497/500 [05:06<00:01, 1.67it/s]
1006 Epoch 497 | loss: 0.271473: 99%| ██████████ | 497/500 [05:07<00:01, 1.67it/s]
1007 Epoch 497 | loss: 0.271473: 100%| ██████████ | 498/500 [05:07<00:01, 1.67it/s]
1008 Epoch 498 | loss: 0.268631: 100%| ██████████ | 498/500 [05:08<00:01, 1.67it/s]
1009 Epoch 498 | loss: 0.268631: 100%| ██████████ | 499/500 [05:08<00:00, 1.67it/s]
1010 Epoch 499 | loss: 0.251173: 100%| ██████████ | 499/500 [05:08<00:00, 1.67it/s]
1011 Epoch 499 | loss: 0.251173: 100%| ██████████ | 500/500 [05:08<00:00, 1.67it/s]
1012 Epoch 499 | loss: 0.251173: 100%| ██████████ | 500/500 [05:08<00:00, 1.62it/s]

```

```

1086 | 99%| ██████████ | 3118/3134 [00:07<00:00, 446.40it/s]
1087 | 100%| ██████████ | 3134/3134 [00:07<00:00, 405.24it/s]
1088 no type constraint results:
1089 metric:      MRR      MR      hit@10    hit@3    hit@1
1090 l(raw):      0.136213   4858.082520  0.421825   0.218571  0.000000
1091 r(raw):      0.155674   3397.925293  0.473197   0.238034  0.007339
1092 averaged(raw): 0.145943   4128.003906  0.447511   0.228302  0.003669
1093
1094 l(filter):    0.192871   4834.606445  0.454052   0.364071  0.001914
1095 r(filter):    0.207637   3392.539551  0.488832   0.378430  0.012125
1096 averaged(filter): 0.200254   4113.573242  0.471442   0.371251  0.007020
1097 0.471442
1098 0.47144225239753723

```

- 发现训练500 epoch的loss降低了一半，且测试结果与原始条件相差甚微，所以预估训练的最佳epoch在500~1000epochs

2.2 Transh

2.2.1 原始条件


```

1 Date=`date +%y%m%d`
2 echo "8.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transh_FB15K237.py --
  margin=4 --nbatches=100 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=0.5 >
  logs/`./result/transh/transh(4,100,100,1,1000,0.5)
  .log`
5 echo "8.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2005 Epoch 996 | loss: 0.059751: 100%| ██████████ | 997/1000 [15:52<00:02, 1.10it/s]
2006 Epoch 997 | loss: 0.059463: 100%| ██████████ | 997/1000 [15:53<00:02, 1.10it/s]
2007 Epoch 997 | loss: 0.059463: 100%| ██████████ | 998/1000 [15:53<00:01, 1.13it/s]
2008 Epoch 998 | loss: 0.062854: 100%| ██████████ | 998/1000 [15:53<00:01, 1.13it/s]
2009 Epoch 998 | loss: 0.062854: 100%| ██████████ | 999/1000 [15:53<00:00, 1.11it/s]
2010 Epoch 999 | loss: 0.063306: 100%| ██████████ | 999/1000 [15:54<00:00, 1.11it/s]
2011 Epoch 999 | loss: 0.063306: 100%| ██████████ | 1000/1000 [15:54<00:00, 1.10it/s]
2012 Epoch 999 | loss: 0.063306: 100%| ██████████ | 1000/1000 [15:54<00:00, 1.05it/s]

```

```

2119 100%| ██████████ | 3122/3134 [00:11<00:00, 312.95it/s]
2120 100%| ██████████ | 3134/3134 [00:11<00:00, 266.22it/s]
2121 no type constraint results:
2122 metric:      MRR      MR      hit@10      hit@3      hit@1
2123 l(raw):      0.134871    6491.286621  0.409700    0.228143    0.000000
2124 r(raw):      0.149951    3728.079102  0.455967    0.240587    0.005105
2125 averaged(raw): 0.142411    5109.682617  0.432833    0.234365    0.002553
2126
2127 l(filter):    0.192379    6467.757324  0.434269    0.374920    0.001595
2128 r(filter):    0.203356    3722.689453  0.467773    0.380664    0.007020
2129 averaged(filter): 0.197868    5095.223633  0.451021    0.377792    0.004308
2130 0.451021
2131 0.45102107524871826

```

2.2.2 margin改为5

```

1 Date=`date +%y%m%d`
2 echo "9.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transh_FB15K237.py --
  margin=5 --nbatches=100 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=0.5 >
  logs/`./result/transh/transh(5,100,100,1,1000,0.5)
  .log`
5 echo "9.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2003 Epoch 995 | loss: 0.187397: 100%| ██████████ | 996/1000 [16:19<00:03, 1.12it/s]
2004 Epoch 996 | loss: 0.178820: 100%| ██████████ | 996/1000 [16:20<00:03, 1.12it/s]
2005 Epoch 996 | loss: 0.178820: 100%| ██████████ | 997/1000 [16:20<00:02, 1.13it/s]
2006 Epoch 997 | loss: 0.178085: 100%| ██████████ | 997/1000 [16:21<00:02, 1.13it/s]
2007 Epoch 997 | loss: 0.178085: 100%| ██████████ | 998/1000 [16:21<00:01, 1.09it/s]
2008 Epoch 998 | loss: 0.183935: 100%| ██████████ | 998/1000 [16:22<00:01, 1.09it/s]
2009 Epoch 998 | loss: 0.183935: 100%| ██████████ | 999/1000 [16:22<00:00, 1.09it/s]
2010 Epoch 999 | loss: 0.183669: 100%| ██████████ | 999/1000 [16:23<00:00, 1.09it/s]
2011 Epoch 999 | loss: 0.183669: 100%| ██████████ | 1000/1000 [16:23<00:00, 1.11it/s]
2012 Epoch 999 | loss: 0.183669: 100%| ██████████ | 1000/1000 [16:23<00:00, 1.02it/s]
2013

```

```

2121 99%| ██████████ | 3113/3134 [00:11<00:00, 261.95it/s]
2122 100%| ██████████ | 3134/3134 [00:11<00:00, 263.24it/s]
2123 no type constraint results:
2124 metric:      MRR      MR      hit@10    hit@3    hit@1
2125 l(raw):      0.138006   5210.297852  0.422463  0.228462  0.000000
2126 r(raw):      0.159668   3527.291260  0.477664  0.249840  0.007339
2127 averaged(raw): 0.148837   4368.794434  0.450064  0.239151  0.003669
2128
2129 l(filter):    0.195326   5186.865234  0.451819  0.371729  0.002234
2130 r(filter):    0.212261   3521.917725  0.492342  0.386088  0.012125
2131 averaged(filter): 0.203794   4354.391602  0.472080  0.378909  0.007179
2132 0.472080
2133 0.47208040952682495

```

- 效果强于原始条件，有较大提升

2.2.3 n_batches改为200

```

1 Date=`date +%y%m%d`
2 echo "10.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transh_FB15K237.py --
  margin=4 --nbatches=200 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=0.5 >
  logs/`./result/transh/transh(4,200,100,1,1000,0.5)
  .log`
5 echo "10.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2007 Epoch 997 | loss: 0.124270: 100%| ██████████ | 998/1000 [27:44<00:04, 2.47s/it]
2008 Epoch 998 | loss: 0.120987: 100%| ██████████ | 998/1000 [27:46<00:04, 2.47s/it]
2009 Epoch 998 | loss: 0.120987: 100%| ██████████ | 999/1000 [27:46<00:02, 2.35s/it]
2010 Epoch 999 | loss: 0.121911: 100%| ██████████ | 999/1000 [27:48<00:02, 2.35s/it]
2011 Epoch 999 | loss: 0.121911: 100%| ██████████ | 1000/1000 [27:48<00:00, 2.24s/it]
2012 Epoch 999 | loss: 0.121911: 100%| ██████████ | 1000/1000 [27:48<00:00, 1.67s/it]

```

```

2130 100%| ██████████ | 3134/3134 [00:13<00:00, 228.13it/s]
2131 no type constraint results:
2132 metric:      MRR      MR      hit@10    hit@3    hit@1
2133 l(raw):      0.133892  6120.731445  0.406509  0.224633  0.000000
2134 r(raw):      0.150489  3729.130615  0.447990  0.244097  0.005424
2135 averaged(raw): 0.142190  4924.931152  0.427250  0.234365  0.002712
2136
2137 l(filter):    0.190570  6097.273926  0.431078  0.370453  0.001914
2138 r(filter):    0.202768  3723.750488  0.461391  0.381940  0.007020
2139 averaged(filter): 0.196669  4910.512207  0.446235  0.376197  0.004467
2140 0.446235
2141 0.4462348222732544

```

- 无显著提升，但至少没有过拟合

2.2.4 dim改为200

```

1 Date=`date +%y%m%d`
2 echo "11.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transh_FB15K237.py --
  margin=4 --nbatches=100 --dim=200 --p_norm=1 --
  train_times=1000 --alpha=0.5 >
  logs/`./result/transh/transh(4,100,200,1,1000,0.5)
  .log`
5 echo "11.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2005 Epoch 996 | loss: 0.030468: 100% | ██████████ | 997/1000 [20:40<00:03, 1.11s/it]
2006 Epoch 997 | loss: 0.029040: 100% | ██████████ | 997/1000 [20:41<00:03, 1.11s/it]
2007 Epoch 997 | loss: 0.029040: 100% | ██████████ | 998/1000 [20:41<00:02, 1.10s/it]
2008 Epoch 998 | loss: 0.031466: 100% | ██████████ | 998/1000 [20:42<00:02, 1.10s/it]
2009 Epoch 998 | loss: 0.031466: 100% | ██████████ | 999/1000 [20:42<00:01, 1.09s/it]
2010 Epoch 999 | loss: 0.031822: 100% | ██████████ | 999/1000 [20:43<00:01, 1.09s/it]
2011 Epoch 999 | loss: 0.031822: 100% | ██████████ | 1000/1000 [20:43<00:00, 1.09s/it]
2012 Epoch 999 | loss: 0.031822: 100% | ██████████ | 1000/1000 [20:43<00:00, 1.24s/it]

```

```

2190 100% | ██████████ | 3133/3134 [00:24<00:00, 91.52it/s]
2191 100% | ██████████ | 3134/3134 [00:24<00:00, 125.39it/s]
2192 no type constraint results:
2193 metric:      MRR      MR      hit@10      hit@3      hit@1
2194 l(raw):      0.125150    7965.114258    0.391512    0.206126    0.000000
2195 r(raw):      0.139344    3804.763184    0.432355    0.223676    0.001595
2196 averaged(raw): 0.132247    5884.938477    0.411934    0.214901    0.000798
2197
2198 l(filter):    0.182157    7941.565918    0.409381    0.355456    0.000957
2199 r(filter):    0.192034    3799.331543    0.441927    0.368220    0.001914
2200 averaged(filter): 0.187096    5870.448730    0.425654    0.361838    0.001436
2201 0.425654
2202 0.4256541132926941

```

- 发生了严重过拟合

2.2.5 p_norm改为2

```

1 Date=`date +%y%m%d`
2 echo "12.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transh_FB15K237.py --
  margin=4 --nbatches=100 --dim=100 --p_norm=2 --
  train_times=1000 --alpha=0.5 >
  logs/`./result/transh/transh(4,100,100,2,1000,0.5)
  .log`
5 echo "12.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2005 Epoch 996 | loss: 290.810003: 100% | 997/1000 [16:26<00:02, 1.15it/s]
2006 Epoch 997 | loss: 290.803325: 100% | 997/1000 [16:27<00:02, 1.15it/s]
2007 Epoch 997 | loss: 290.803325: 100% | 998/1000 [16:27<00:01, 1.15it/s]
2008 Epoch 998 | loss: 291.046695: 100% | 998/1000 [16:27<00:01, 1.15it/s]
2009 Epoch 998 | loss: 291.046695: 100% | 999/1000 [16:27<00:00, 1.13it/s]
2010 Epoch 999 | loss: 290.985742: 100% | 999/1000 [16:28<00:00, 1.13it/s]
2011 Epoch 999 | loss: 290.985742: 100% | 1000/1000 [16:28<00:00, 1.14it/s]
2012 Epoch 999 | loss: 290.985742: 100% | 1000/1000 [16:28<00:00, 1.01it/s]

```

```

2125 100% | 3134/3134 [00:12<00:00, 258.03it/s]
2126 no type constraint results:
2127 metric:      MRR      MR      hit@10      hit@3      hit@1
2128 l(raw):      0.012927    7268.114746    0.030632    0.012125    0.002553
2129 r(raw):      0.043794    6866.465820    0.082642    0.046586    0.021378
2130 averaged(raw): 0.028360    7067.290039    0.056637    0.029355    0.011966
2131
2132 l(filter):    0.021719    7247.605469    0.042119    0.022655    0.009572
2133 r(filter):    0.046703    6861.593262    0.083599    0.048500    0.025207
2134 averaged(filter): 0.034211    7054.599609    0.062859    0.035578    0.017390
2135 0.062859
2136 0.06285896897315979

```

- 整个模型都坏掉，实在不适合

2.2.6 epoch改为500

```

1 Date=`date +%y%m%d`
2 echo "13.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transh_FB15K237.py --
  margin=4 --nbatches=100 --dim=100 --p_norm=1 --
  train_times=500 --alpha=0.5 >
  logs/`./result/transh/transh(4,100,100,1,500,0.5).
  log`
5 echo "13.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

1006 Epoch 497 | loss: 0.080395: 99%| ██████████ | 497/500 [08:37<00:02, 1.06it/s]
1007 Epoch 497 | loss: 0.080395: 100%| ██████████ | 498/500 [08:37<00:01, 1.05it/s]
1008 Epoch 498 | loss: 0.081110: 100%| ██████████ | 498/500 [08:37<00:01, 1.05it/s]
1009 Epoch 498 | loss: 0.081110: 100%| ██████████ | 499/500 [08:37<00:00, 1.08it/s]
1010 Epoch 499 | loss: 0.073310: 100%| ██████████ | 499/500 [08:38<00:00, 1.08it/s]
1011 Epoch 499 | loss: 0.073310: 100%| ██████████ | 500/500 [08:38<00:00, 1.07it/s]
1012 Epoch 499 | loss: 0.073310: 100%| ██████████ | 500/500 [08:38<00:00, 1.04s/it]
1013

```

```

1116 | 99%| ██████████ | 3116/3134 [00:11<00:00, 300.85it/s]
1117 | 100%| ██████████ | 3134/3134 [00:11<00:00, 276.56it/s]
1118 no type constraint results:
1119 metric:      MRR      MR      hit@10      hit@3      hit@1
1120 l(raw):      0.131306    6369.646484    0.404595    0.216656    0.000000
1121 r(raw):      0.151146    4088.569580    0.447990    0.240906    0.006382
1122 averaged(raw): 0.141226    5229.107910    0.426292    0.228781    0.003191
1123
1124 l(filter):    0.189459    6346.155762    0.431398    0.366624    0.002553
1125 r(filter):    0.203111    4083.173340    0.460753    0.376516    0.011487
1126 averaged(filter): 0.196285    5214.664551    0.446075    0.371570    0.007020
1127 0.446075
1128 0.44607532024383545

```

- 效果一般，感觉有点欠拟合

2.2.7 学习率改为1

```

1 Date=`date +%y%m%d`
2 echo "14.sh back begin at `date +%H:%M:%S`" >>
  out.log
3
4 nohup python -u train_transh_FB15K237.py --
  margin=4 --nbatches=100 --dim=100 --p_norm=1 --
  train_times=1000 --alpha=1 >
  logs/`./result/transh/transh(4,100,100,1,1000,1).1
  og`
5 echo "14.sh back end at `date +%H:%M:%S`" >>
  out.log

```

```

2005 Epoch 996 | loss: 0.062057: 100%| 997/1000 [16:34<00:02, 1.11it/s]
2006 Epoch 997 | loss: 0.060758: 100%| 997/1000 [16:35<00:02, 1.11it/s]
2007 Epoch 997 | loss: 0.060758: 100%| 998/1000 [16:35<00:01, 1.12it/s]
2008 Epoch 998 | loss: 0.064736: 100%| 998/1000 [16:35<00:01, 1.12it/s]
2009 Epoch 998 | loss: 0.064736: 100%| 999/1000 [16:35<00:00, 1.13it/s]
2010 Epoch 999 | loss: 0.065083: 100%| 999/1000 [16:36<00:00, 1.13it/s]
2011 Epoch 999 | loss: 0.065083: 100%| 1000/1000 [16:36<00:00, 1.15it/s]
2012 Epoch 999 | loss: 0.065083: 100%| 1000/1000 [16:36<00:00, 1.00it/s]
2013

```

```

2113 100%| 3126/3134 [00:10<00:00, 272.02it/s]
2114 100%| 3134/3134 [00:10<00:00, 287.48it/s]
2115 no type constraint results:
2116 metric:      MRR      MR      hit@10      hit@3      hit@1
2117 l(raw):      0.135098    6139.355469    0.408743    0.228781    0.000000
2118 r(raw):      0.147974    3716.475830    0.449585    0.230696    0.004467
2119 averaged(raw): 0.141536    4927.915527    0.429164    0.229738    0.002234
2120
2121 l(filter):    0.191477    6115.862305    0.432036    0.371729    0.001276
2122 r(filter):    0.199862    3711.073486    0.462029    0.373325    0.005105
2123 averaged(filter): 0.195670    4913.467773    0.447033    0.372527    0.003191
2124 0.447033
2125 0.4470325708389282
2126

```

- 过拟合

2. week6


NVIDIA Tesla K80
 ID: wza7GV
 ● 已释放

硬件信息

GPU: NVIDIA Tesla K80
 每秒浮点运算次数: 1.37 TFLOPS
 显卡内存: 12 GB

租用配置

镜像: Keras 2.2 & Tensorflow 1.13.1 GP...
 镜像描述: 预装: Python 3.5, CUDA 10.0,...
 挂载: /mnt

总计: ¥5.12

折扣价: ¥ 1.00/小时
 原价: ¥ 1.99/小时
 租用开始于: 2021-12-06 15:03

2.1 分别使用EN_FR_15K_V2的split1和EN_DE_15K_V2的split2来运行MTransE, 记录使用的命令和结果

2.1.1 EN_FR_15K_V2的split1

```
1 python main_from_args.py
   ./args/mtranse_args_15K.json EN_FR_15K_V2
   721_5fold/1/
```

```
epoch 177, avg. mapping loss: 0.2076, cost time: 1.6305s
epoch 178, avg. triple loss: 0.2771, cost time: 2.4821s
epoch 178, avg. mapping loss: 0.1956, cost time: 1.6316s
epoch 179, avg. triple loss: 0.2765, cost time: 2.4742s
epoch 179, avg. mapping loss: 0.2093, cost time: 1.6333s
epoch 180, avg. triple loss: 0.2759, cost time: 2.4848s
epoch 180, avg. mapping loss: 0.1992, cost time: 1.5824s
quick results: hits@[1, 5, 10, 50] = [24. 41.6 49.2 67. ]%, time = 0.726 s

== should early stop ==

Training ends. Total time = 785.819 s.
accurate results: hits@[1, 5, 10, 50] = [22.952 41.914 50.6 69.667]%, mr = 217.770, mrr = 0.321593, time = 9.100 s
accurate results with csls: csls=10, hits@[1, 5, 10, 50] = [32.648 55.105 64.562 82.933]%, mr = 67.767, mrr = 0.433668, time = 12.716 s
Results saved!
.../output/results/MTransE/EN_FR_15K_V2/721_5fold/1/20211206105908/kg1_ent_ids saved.
.../output/results/MTransE/EN_FR_15K_V2/721_5fold/1/20211206105908/kg2_ent_ids saved.
.../output/results/MTransE/EN_FR_15K_V2/721_5fold/1/20211206105908/kg1_rel_ids saved.
.../output/results/MTransE/EN_FR_15K_V2/721_5fold/1/20211206105908/kg2_rel_ids saved.
.../output/results/MTransE/EN_FR_15K_V2/721_5fold/1/20211206105908/kg1_attr_ids saved.
.../output/results/MTransE/EN_FR_15K_V2/721_5fold/1/20211206105908/kg2_attr_ids saved.
Embeddings saved!
Total run time = 824.684 s.
(openeae2) root@58a6f813cf09:/mnt/OpenEA-master/run#
```

> EN_DE_15K_V2\721_5fold\2\202112061...	1	http://dbpedia.org/resource/E245396	0
EN_FR_15K_V2\721_5fold\1\202112061...	2	http://dbpedia.org/resource/E292763	2
alignment_results_12	3	http://dbpedia.org/resource/E172224	4
ent_embeddings.npy	4	http://dbpedia.org/resource/E206541	6
kg1_attr_ids	5	http://dbpedia.org/resource/E388971	8
kg1_ent_embeddings_txt	6	http://dbpedia.org/resource/E204061	10
kg1_ent_ids	7	http://dbpedia.org/resource/E380309	12
kg1_rel_embeddings_txt	8	http://dbpedia.org/resource/E269014	14
kg1_rel_ids	9	http://dbpedia.org/resource/E145920	16
kg2_attr_ids	10	http://dbpedia.org/resource/E612528	18
kg2_ent_embeddings_txt	11	http://dbpedia.org/resource/E061882	20
kg2_ent_ids	12	http://dbpedia.org/resource/E917119	22
kg2_rel_embeddings_txt	13	http://dbpedia.org/resource/E891999	24
kg2_rel_ids			

MTRANSE		EN_FR_15K_V2 > 721_5fold > 1 > 20211206105908 > kg2_ent_ids		
> EN_DE_15K_V2		1	http://fr.dbpedia.org/resource/E294655	1
EN_FR_15K_V2\721_5fold\1\202112061...		2	http://fr.dbpedia.org/resource/E791059	3
alignment_results_12		3	http://fr.dbpedia.org/resource/E319827	5
ent_embeddings.npy		4	http://fr.dbpedia.org/resource/E924195	7
kg1_attr_ids		5	http://fr.dbpedia.org/resource/E481752	9
kg1_ent_embeddings_txt		6	http://fr.dbpedia.org/resource/E098676	11
kg1_ent_ids		7	http://fr.dbpedia.org/resource/E859836	13
kg1_rel_embeddings_txt		8	http://fr.dbpedia.org/resource/E313692	15
kg1_rel_ids		9	http://fr.dbpedia.org/resource/E455185	17
kg2_attr_ids		10	http://fr.dbpedia.org/resource/E125020	19
kg2_ent_embeddings_txt		11	http://fr.dbpedia.org/resource/E988247	21
kg2_ent_ids		12	http://fr.dbpedia.org/resource/E851859	23
kg2_rel_embeddings_txt		13	http://fr.dbpedia.org/resource/E812144	25
kg2_rel_ids				
mapping_mat.npy				

2.1.2 EN_DE_15K_V2的split2

```
1 | python main_from_args.py
   ./args/mtranse_args_15K.json EN_DE_15K_V2
   721_5fold/2/]
```

```
epoch 170, avg. triple loss: 0.0710, cost time: 2.4267s
epoch 170, avg. mapping loss: 0.0911, cost time: 1.5808s
quick results: hits@[1, 5, 10, 50] = [19.533 34.133 42.733 64.067]%, time = 0.746 s

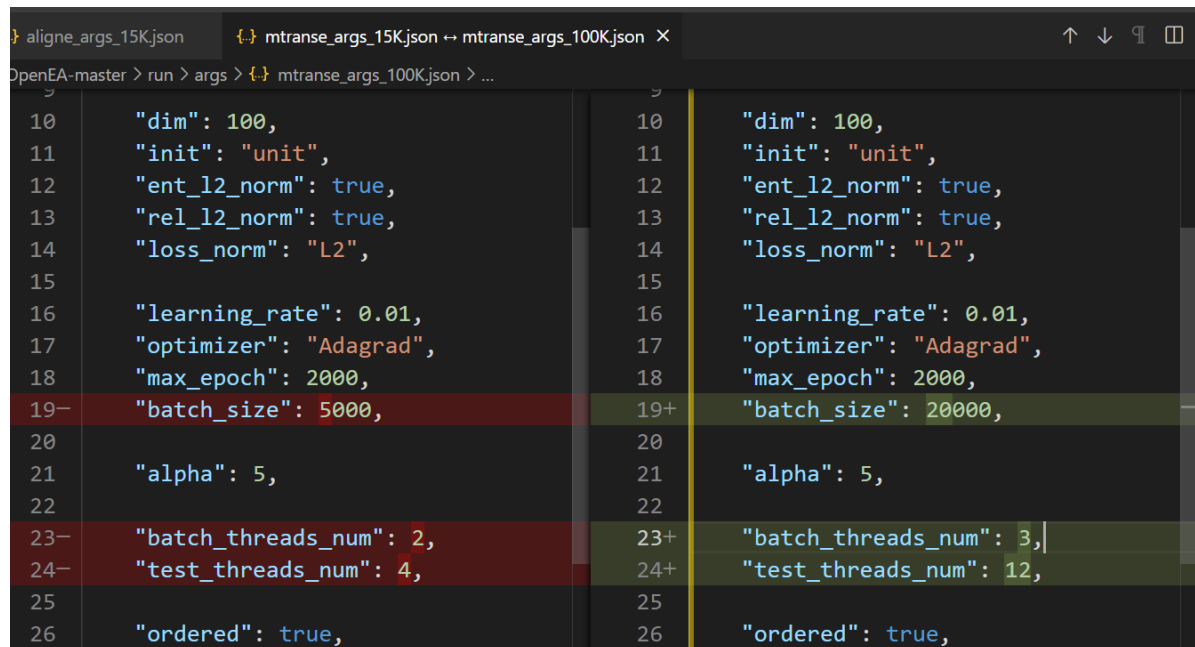
== should early stop ==

Training ends. Total time = 700.301 s.
accurate results: hits@[1, 5, 10, 50] = [20.705 37.124 44.99 67.162]%, mr = 179.597, mrr = 0.290044, time = 9.313 s
accurate results with csfs: csfs=10, hits@[1, 5, 10, 50] = [26.41 45.99 55.219 77.143]%, mr = 98.254, mrr = 0.360881, time = 12.964 s
Results saved!
.../output/results/MTransE/EN_DE_15K_V2/721_5fold/2/20211206111639/kg1_ent_ids saved.
.../output/results/MTransE/EN_DE_15K_V2/721_5fold/2/20211206111639/kg2_ent_ids saved.
.../output/results/MTransE/EN_DE_15K_V2/721_5fold/2/20211206111639/kg1_rel_ids saved.
.../output/results/MTransE/EN_DE_15K_V2/721_5fold/2/20211206111639/kg2_rel_ids saved.
.../output/results/MTransE/EN_DE_15K_V2/721_5fold/2/20211206111639/kg1_attr_ids saved.
.../output/results/MTransE/EN_DE_15K_V2/721_5fold/2/20211206111639/kg2_attr_ids saved.
Embeddings saved!
Total run time = 742.507 s.
(openea2) root@58a6f813cf09:/mnt/OpenEA-master/run#
```

MTRANSE		EN_DE_15K_V2 > 721_5fold > 2 > 20211206111639 > kg1_ent_ids	
✓ EN_DE_15K_V2\721_5fold\2\202112061...		1	http://dbpedia.org/resource/E156265 0
alignment_results_12		2	http://dbpedia.org/resource/E940714 2
ent_embeds.npy		3	http://dbpedia.org/resource/E737899 4
kg1_attr_ids		4	http://dbpedia.org/resource/E183425 6
kg1_ent_embeds_txt		5	http://dbpedia.org/resource/E832030 8
⚙ kg1_ent_ids		6	http://dbpedia.org/resource/E036271 10
kg1_rel_embeds_txt		7	http://dbpedia.org/resource/E507224 12
kg1_rel_ids		8	http://dbpedia.org/resource/E004640 14
kg2_attr_ids		9	http://dbpedia.org/resource/E021511 16
kg2_ent_embeds_txt		10	http://dbpedia.org/resource/E095770 18
kg2_ent_ids		11	http://dbpedia.org/resource/E030112 20
kg2_rel_embeds_txt		12	http://dbpedia.org/resource/E805323 22
kg2_rel_ids		13	http://dbpedia.org/resource/E014986 24
mapping_mat.npy		14	http://dbpedia.org/resource/E506020 26
rel_embeds.npy		15	http://dbpedia.org/resource/E397777 28
> EN_FR_15K_V2		16	http://dbpedia.org/resource/E537013 30
		17	http://dbpedia.org/resource/E635776 32

TRANSE		EN_DE_15K_V2 > 721_5fold > 2 > 20211206111639 > kg2_ent_ids	
EN_DE_15K_V2\721_5fold\2\202112061...		1	http://de.dbpedia.org/resource/E209564 1
alignment_results_12		2	http://de.dbpedia.org/resource/E326381 3
ent_embeds.npy		3	http://de.dbpedia.org/resource/E931673 5
kg1_attr_ids		4	http://de.dbpedia.org/resource/E424305 7
kg1_ent_embeds_txt		5	http://de.dbpedia.org/resource/E052472 9
⚙ kg1_ent_ids		6	http://de.dbpedia.org/resource/E402418 11
kg1_rel_embeds_txt		7	http://de.dbpedia.org/resource/E994111 13
kg1_rel_ids		8	http://de.dbpedia.org/resource/E058102 15
kg2_attr_ids		9	http://de.dbpedia.org/resource/E897209 17
kg2_ent_embeds_txt		10	http://de.dbpedia.org/resource/E062375 19
⚙ kg2_ent_ids		11	http://de.dbpedia.org/resource/E263036 21
kg2_rel_embeds_txt		12	http://de.dbpedia.org/resource/E012750 23
kg2_rel_ids		13	http://de.dbpedia.org/resource/E088518 25
mapping_mat.npy		14	http://de.dbpedia.org/resource/E973424 27
rel_embeds.npy		15	http://de.dbpedia.org/resource/E126902 29
EN_FR_15K_V2\721_5fold\1\202112061...		16	http://de.dbpedia.org/resource/E599620 31
alignment_results_12			

2.2 mtranse_args_15K.json和mtranse_args_100K.json有何区别，为什么要设置这种区别，而不是直接写一个mtranse_args.json?



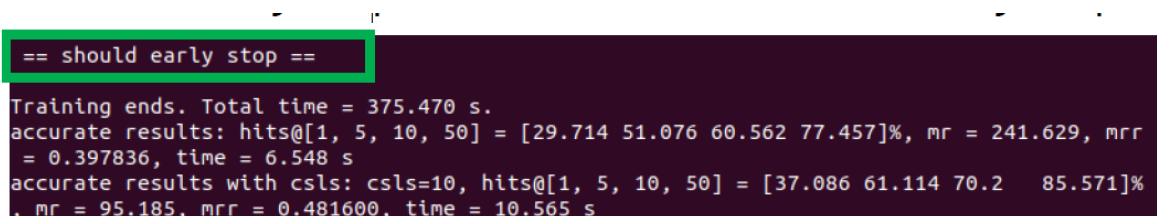
```
aligne_args_15K.json  mtranse_args_15K.json ↔ mtranse_args_100K.json ×
OpenEA-master > run > args > mtranse_args_100K.json > ...

10  "dim": 100,
11  "init": "unit",
12  "ent_l2_norm": true,
13  "rel_l2_norm": true,
14  "loss_norm": "L2",
15
16  "learning_rate": 0.01,
17  "optimizer": "Adagrad",
18  "max_epoch": 2000,
19- "batch_size": 5000,
20
21  "alpha": 5,
22
23- "batch_threads_num": 2,
24- "test_threads_num": 4,
25
26  "ordered": true,

10  "dim": 100,
11  "init": "unit",
12  "ent_l2_norm": true,
13  "rel_l2_norm": true,
14  "loss_norm": "L2",
15
16  "learning_rate": 0.01,
17  "optimizer": "Adagrad",
18  "max_epoch": 2000,
19+ "batch_size": 20000,
20
21  "alpha": 5,
22
23+ "batch_threads_num": 3,
24+ "test_threads_num": 12,
25
26  "ordered": true,
```

- 我们发现这两个文件对应训练的规模不同，mtranse_args_15K.json训练的batch_size比较小，所以对应的阈值也设置比较小；mtranse_args_100K.json训练的batch_size比较大，所以对应的阈值也设置比较大

2.3 什么是earlystop? 这个实例中为什么需要earlystop?



```
== should early stop ==
Training ends. Total time = 375.470 s.
accurate results: hits@[1, 5, 10, 50] = [29.714 51.076 60.562 77.457]%, mr = 241.629, mrr
= 0.397836, time = 6.548 s
accurate results with csls: csls=10, hits@[1, 5, 10, 50] = [37.086 61.114 70.2 85.571]%,
mr = 95.185, mrr = 0.481600, time = 10.565 s
```

- earlystop指的是在跑完所有epoch前停止训练；
- 在实例中，由于为防止训练过拟合，当我们发现测试的准确率发生明显下降，我们应该停止迭代

