项目汇报

汇报人： 周昱辰

2024年6月20日

一 本周总结

Published as a conference paper at ICLR 2021

GRAPHCODEBERT: PRE-TRAINING CODE REPRESENTATIONS WITH DATA FLOW

Daya Guo[1]*, Shuo Ren[2]*, Shuai Lu[3]*, Zhangyin Feng[4]*, Duyu Tang[5], Shujie Liu[5], Long Zhou[5], Nan Duan[5], Alexey Svyatkovskiy[6], Shengyu Fu[6], Michele Tufano[6], Shao Kun Deng[6], Colin Clement[6], Dawn Drain[6], Neel Sundaresan[6], Jian Yin[1], Daxin Jiang[7], and Ming Zhou[5]
[1]School of Computer Science and Engineering, Sun Yat-sen University.
[2]Beihang University, [3]Peking University, [4]Harbin Institute of Technology,
[5]Microsoft Research Asia, [6]Microsoft Devdiv, [7]Microsoft STCA

采用了DFG（数据流向图），DFG在一定程的上能够避免AST中一些不必要的联系从而提升了模型的有效性。

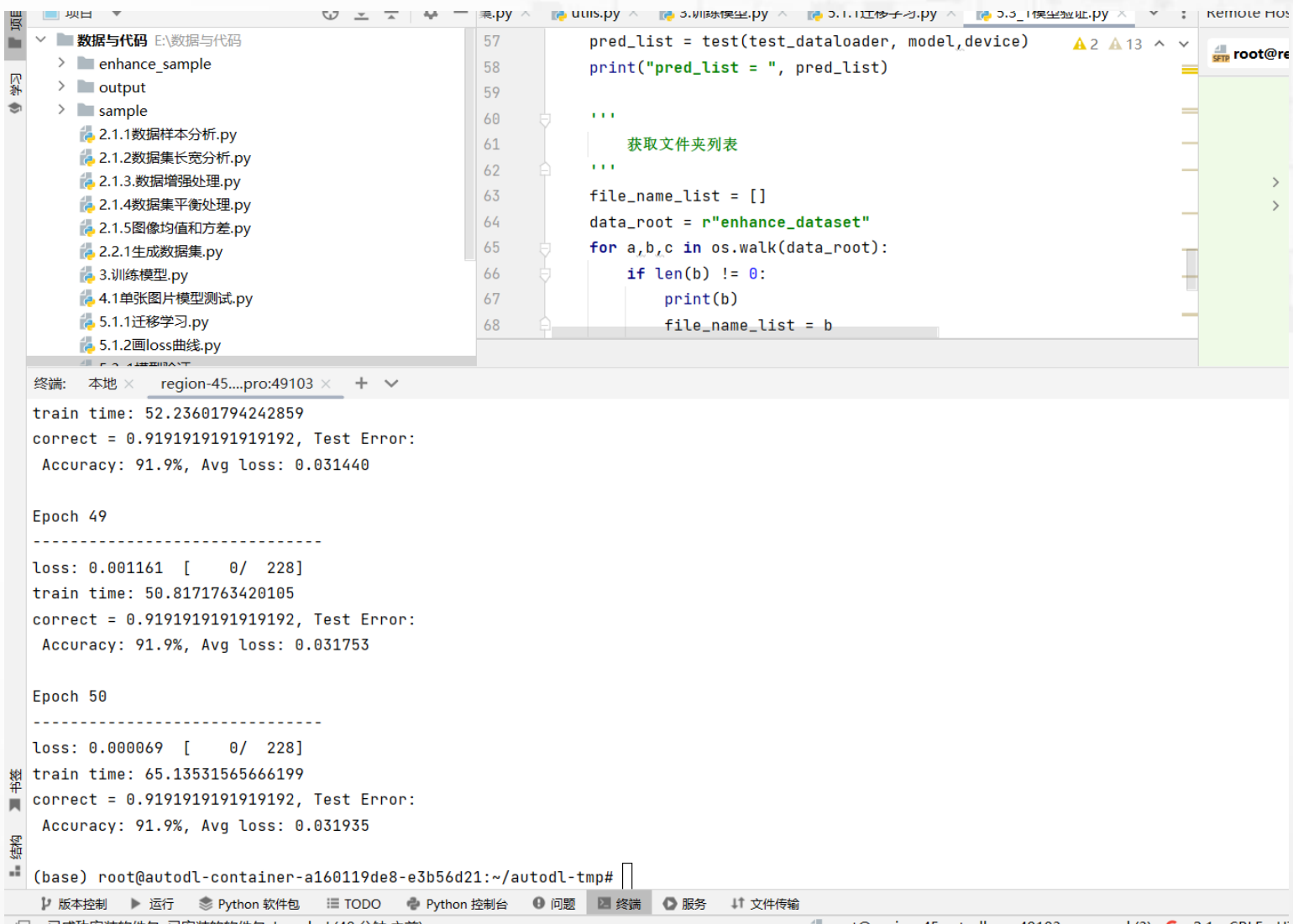
GraphCodeBert也是基于Transformer开发的

GraphCodeBert是通过三项预测任务来训练模型的：MLM（掩语言模型）、EP（数据流向图的边预测）和NA（数据流向图节点和代码token之间的关系预测）。

用于四项下游任务，包括了：代码搜索、克隆检测、代码翻译和代码改错

一 本周总结

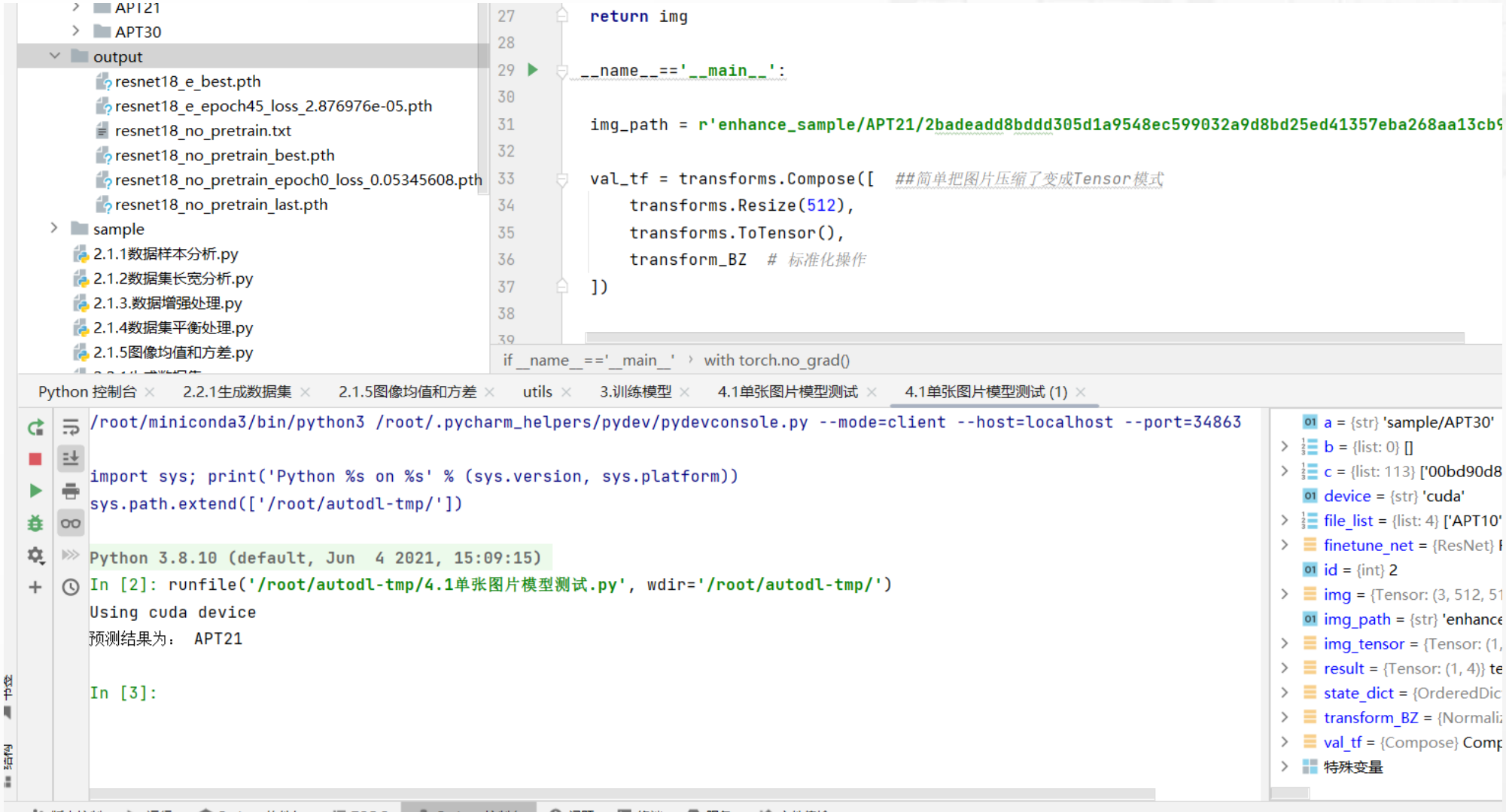
用图模型ResNet18进行图像多分类



一 本周总结

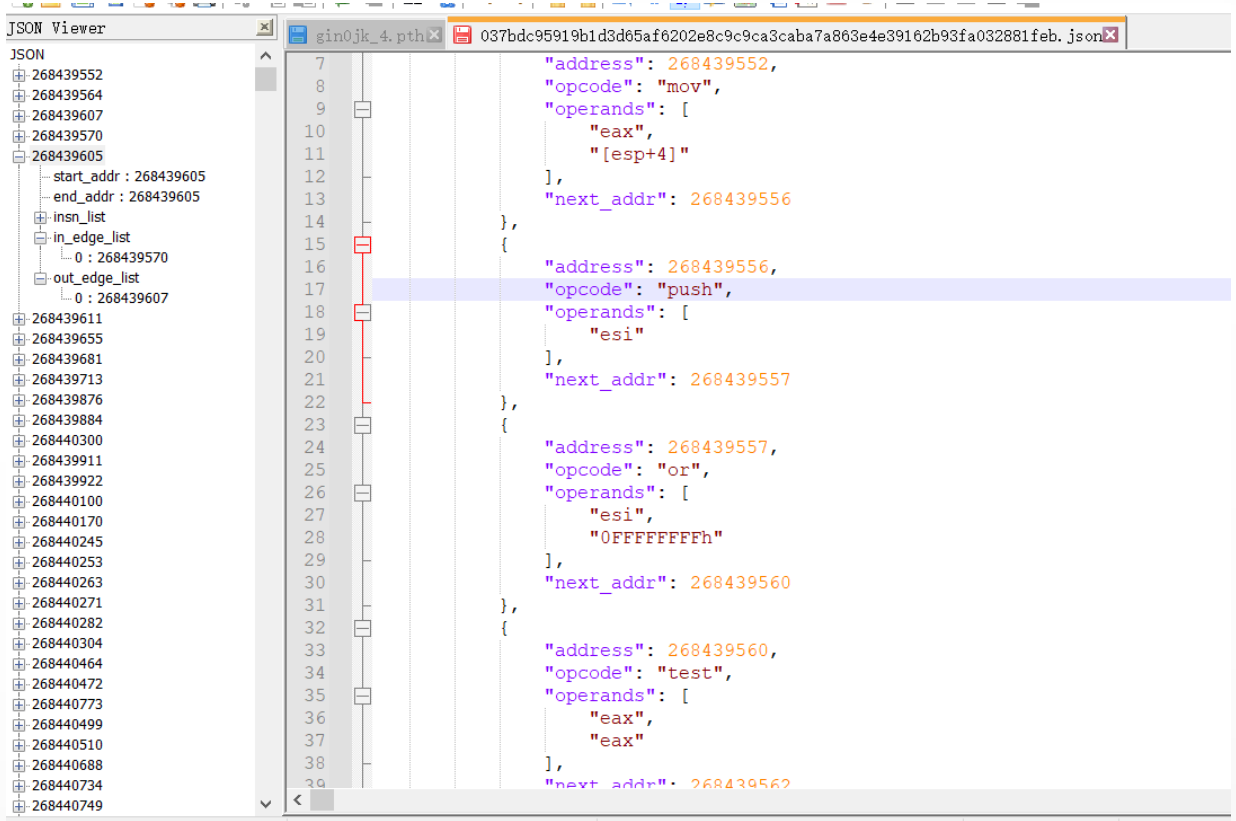
用TEXT-CNN进行多分类

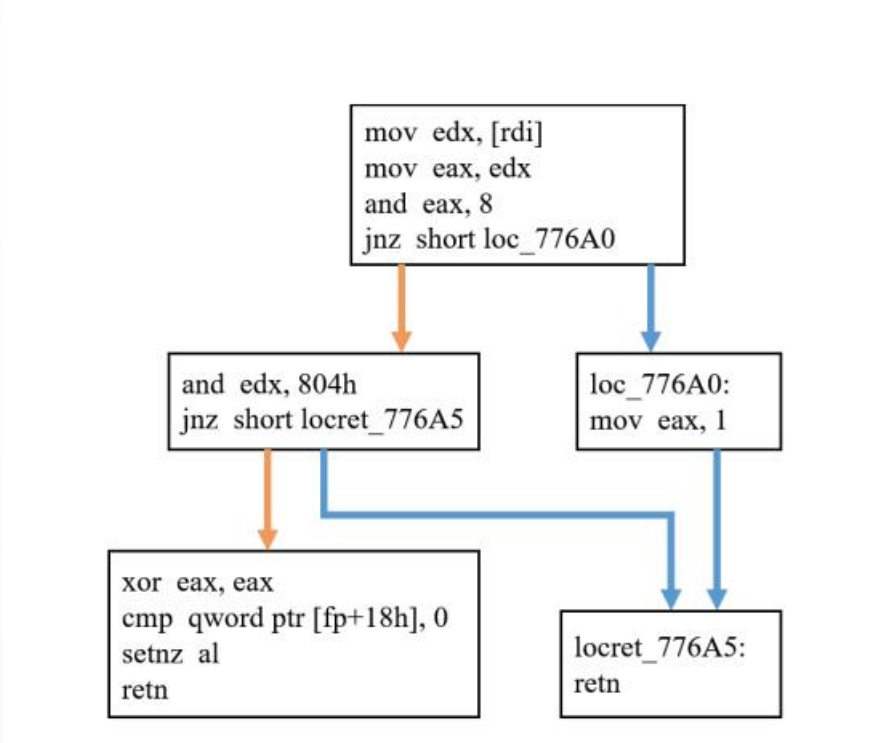
单张测试



项目的任务

汇编文件构建控制流图
通过学习控制流图的语义和结构特征对恶意软件分类





一 项目的任务

数据集比较小，用的数据集是APT-malware约1600条

log_train | best_val_acc.txt | train.py | measure.py | dataset.py

```
6354  Step 4526, Epoch 69/69, iter 41/64, Loss 0.6583, Acc 0.7500.
6355  Step 4527, Epoch 69/69, iter 42/64, Loss 0.5524, Acc 0.8500.
6356  Step 4528, Epoch 69/69, iter 43/64, Loss 1.0228, Acc 0.8000.
6357  Step 4529, Epoch 69/69, iter 44/64, Loss 1.0792, Acc 0.7000.
6358  Step 4530, Epoch 69/69, iter 45/64, Loss 0.5293, Acc 0.8500.
6359  Step 4531, Epoch 69/69, iter 46/64, Loss 0.6804, Acc 0.8000.
6360  Step 4532, Epoch 69/69, iter 47/64, Loss 0.8887, Acc 0.7500.
6361  Step 4533, Epoch 69/69, iter 48/64, Loss 0.7871, Acc 0.7500.
6362  Step 4534, Epoch 69/69, iter 49/64, Loss 0.7524, Acc 0.7500.
6363  Step 4535, Epoch 69/69, iter 50/64, Loss 0.6268, Acc 0.8000.
6364  Step 4536, Epoch 69/69, iter 51/64, Loss 0.7015, Acc 0.7500.
6365  Step 4537, Epoch 69/69, iter 52/64, Loss 0.8049, Acc 0.7000.
6366  Step 4538, Epoch 69/69, iter 53/64, Loss 0.9094, Acc 0.7500.
6367  Step 4539, Epoch 69/69, iter 54/64, Loss 0.5217, Acc 0.9000.
6368  Step 4540, Epoch 69/69, iter 55/64, Loss 0.7526, Acc 0.7000.
6369  Step 4541, Epoch 69/69, iter 56/64, Loss 0.5184, Acc 0.8000.
6370  Step 4542, Epoch 69/69, iter 57/64, Loss 0.7555, Acc 0.7500.
6371  Step 4543, Epoch 69/69, iter 58/64, Loss 0.8474, Acc 0.8000.
6372  Step 4544, Epoch 69/69, iter 59/64, Loss 0.6699, Acc 0.8500.
6373  Step 4545, Epoch 69/69, iter 60/64, Loss 0.3584, Acc 0.8000.
6374  Step 4546, Epoch 69/69, iter 61/64, Loss 0.7690, Acc 0.8500.
6375  Step 4547, Epoch 69/69, iter 62/64, Loss 0.8142, Acc 0.8000.
6376  Step 4548, Epoch 69/69, iter 63/64, Loss 1.0478, Acc 0.6500.
6377  Step 4549, Epoch 69/69, iter 64/64, Loss 0.8948, Acc 0.7778.
6378  Epoch: 69
6379  Train Loss: 0.7052, Train Acc: 0.7929
6380  Val Loss: 0.7791, Val Acc: 0.7785
6381  Best val acc: 0.7846
6382
```

corpus_vocab.sh | dataset.py | train.py | train.sh | log_train | gnn\best_val_acc.txt | model.py

```
7333  Step 7119, Epoch 69/69, iter 81/101, Loss 0.0356, Acc 1.0000.
7334  Step 7120, Epoch 69/69, iter 82/101, Loss 0.1503, Acc 1.0000.
7335  Step 7121, Epoch 69/69, iter 83/101, Loss 0.3331, Acc 0.9000.
7336  Step 7122, Epoch 69/69, iter 84/101, Loss 0.0415, Acc 1.0000.
7337  Step 7123, Epoch 69/69, iter 85/101, Loss 0.0527, Acc 1.0000.
7338  Step 7124, Epoch 69/69, iter 86/101, Loss 0.5059, Acc 0.8500.
7339  Step 7125, Epoch 69/69, iter 87/101, Loss 0.0856, Acc 1.0000.
7340  Step 7126, Epoch 69/69, iter 88/101, Loss 0.1454, Acc 1.0000.
7341  Step 7127, Epoch 69/69, iter 89/101, Loss 0.1050, Acc 1.0000.
7342  Step 7128, Epoch 69/69, iter 90/101, Loss 0.0568, Acc 1.0000.
7343  Step 7129, Epoch 69/69, iter 91/101, Loss 0.0437, Acc 1.0000.
7344  Step 7130, Epoch 69/69, iter 92/101, Loss 0.1693, Acc 0.9500.
7345  Step 7131, Epoch 69/69, iter 93/101, Loss 0.1209, Acc 1.0000.
7346  Step 7132, Epoch 69/69, iter 94/101, Loss 0.1263, Acc 1.0000.
7347  Step 7133, Epoch 69/69, iter 95/101, Loss 0.3209, Acc 0.9000.
7348  Step 7134, Epoch 69/69, iter 96/101, Loss 0.1856, Acc 0.9500.
7349  Step 7135, Epoch 69/69, iter 97/101, Loss 0.2125, Acc 0.9500.
7350  Step 7136, Epoch 69/69, iter 98/101, Loss 0.0823, Acc 1.0000.
7351  Step 7137, Epoch 69/69, iter 99/101, Loss 0.1412, Acc 0.9500.
7352  Step 7138, Epoch 69/69, iter 100/101, Loss 0.1380, Acc 0.9500.
7353  Step 7139, Epoch 69/69, iter 101/101, Loss 2.0129, Acc 0.3333.
7354  Epoch: 69
7355  Train Loss: 0.1521, Train Acc: 0.9644
7356  Val Loss: 0.3883, Val Acc: 0.9289
7357  Best val acc: 0.9308
7358
```

一 项目的任务

尝试了不同图模型，调整参数和数据集

corpus_vocab.sh | dataset.py | train.py | train.sh

```
1  dgcnn_2: 0.7846153846153846
2  gin0jk_2: 0.864321608040201
3  gin0jk_2: 0.9308300395256917
4
```

对恶意软件分类贡献最大的子图

```
CFGExplainer result: [malware = Ldpinch |  graph = Trojan-PSW.Win32.LdPinch.bu| #nodes = 1582]

node :1
node sub_13149F50(320118643) info:
 start: 320118643
 prev: []
 instructions:
  320118643, ['test', 'esi,3']
  320118649, ['jz', 'shortloc_13149F8E']
 next: [320118651, 320118670]

node :2
node sub_13149F50(320118651) info:
 start: 320118651
 prev: []
 instructions:
  320118651, ['xor', 'al,[esi]']
  320118653, ['inc', 'esi']
  320118654, ['mov', 'ebx,0FFh']
  320118659, ['and', 'ebx,eax']
  320118661, ['shr', 'eax,8']
  320118664, ['xor', 'eax,[edi+ebx*4]']
  320118667, ['dec', 'ecx']
  320118668, ['jnz', 'shortloc_13149F73']
 next: [320118643, 320118670]

node :3
node sub_13149F50(320118684) info:
 start: 320118684
 prev: []
 instructions:
  320118684, ['xor', 'eax,[esi]']
  320118686, ['add', 'esi,4']
  320118689, ['mov', 'ebx,0FFh']
```