**原文作者**：Chen Liang,Ziqi Liu,Bin Liu,Jun Zhou,Xiaolong Li (均来自蚂蚁金服)

**原文标题**：Who Stole the Postage? Fraud Detection in Return-Freight Insurance Claims

**原文会议**：SIGKDD2018 (CCF A类)

**原文链接**：

https://github.com/chenlianMT/Who-Stole-the-Postage-/blob/master/return-freight-insurance.pdf

**主要内容**：通过图的学习进行运费险诈骗识别

**论文笔记作者**：Carrie

第一章 Introduction

第一章首先介绍了“运费险诈骗问题”发生的相关背景；并声明再本文将把相关涉及运费险诈骗的相关账户的识别通过有监督的二分类方法划分为为frandulent and regular两类。本文的目标是：以更高的准确率，比目前已有的基于规则发现fraudulent accounts的方法发现更多fraudulent accounts。

然后，介绍了Fraud Detection中面临的三大主要困难：

1. Concept Drith: 具体含义可在文中1.2节找到；重要的是在此作者提出了使用图进行研究的原因：Relations between collaborating fraudsters are naturally illustrated through a device-sharing graph and modeled with graph learning algorithms.。
2. Label uncertainty: 训练数据的标签不确定性问题。已有的基于规则的fraud detection system会对每个账号进行风险等级判别得到"hith risk"、"low risk"和"no observable risk"的标签；作者认为，能够确定"hith risk"账号的性质，但却无法确定"no observable risk"账号是否恶意，因此存在" Label uncertainty "的困难。针对此的解决，作者在5.1 Data PreParation 中进行了阐述；
3. Excessive human effort: 具体信息可在论文1.2节中查看。

第二章 Related Work

第二章首先大致阐述了已有的Insurance fraud detection是基于有监督学习(NN, SVM Regression, Bayesian, DT)、无监督学习(Cluster analysis, outlier detection, spike detection)和二者结合的 三类方法进行的。

然后作者提出本文用了基于图的方法: Our two approaches fall under supervised learning and hybrids of both, respectively. Our approaches differ, as they represent data with graphs, which are one of the most natural representations of data and allow for complex analysis without simplification of data.

第三章 Graph Construction

本文涉及到了三类图: device-sharing graph, transaction graph 和 friendship graph，第三章阐述了关于图的选择与构建的相关内容。

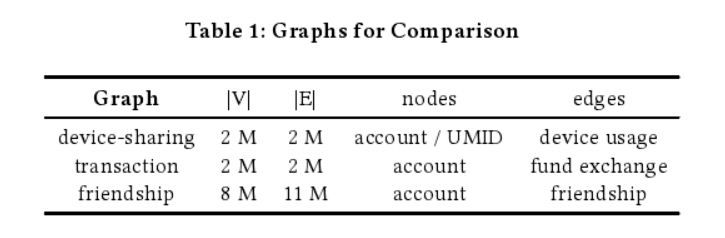
1. 作者阐述了在当前研究的问题中，Good Graph的两则要求：① 相距较近的节点应当具有相似的标签；② fraudulent accounts 与 regular accounts 所构成图的结构应当是有明显不同的。
2. 三类图的构建

|  |  |  |  |
| --- | --- | --- | --- |
| Graph | vertex | edge | Function |
| Device-Sharing Graph | A device(UMID) or an account | Edges only exist between a device vertex and a UMID vertex | Indicating log-in activities in the history |
| Transaction Graph | Accounts | An edge indicates the existence of established transactions between accounts. | Show fund exchange relations between accounts |
| Friendship Graph | 基于Alipay的好友关系构建图 | | |

构建图后的处理：移除掉孤立的点，具体见3.2节。

1. 图的比较

通过作图发现，正常账户和账户账户团体在Device-sharing Graph和Friendship Graph两类图中的形态是不同的，但在Transaction Graph中却没有明显的差别。在此作者再次提出对图的要求：能够区分正常账户与诈骗账户，尤其是在图中相距较近的节点需要具有相同的标签。根据Table1统计的结果，作者选用了Device-sharing Graph来进行分类任务。



第四章 Graph Learning Approach

作者分别采用两种方式对已构建的图进行学习表示。一种是基于Node2Vec方法对图中节点进行向量化，然后将向量作为分类器的输入(We concatenate graph embeddings and account features and feed the new feature vectors to downstream classification tasks using gradient boosted decision tree ) 。另一种是基于图神经网络GNN选择了GeniePath方法进行表示，最后再通过sigmoid层或softmax层完成分类任务。

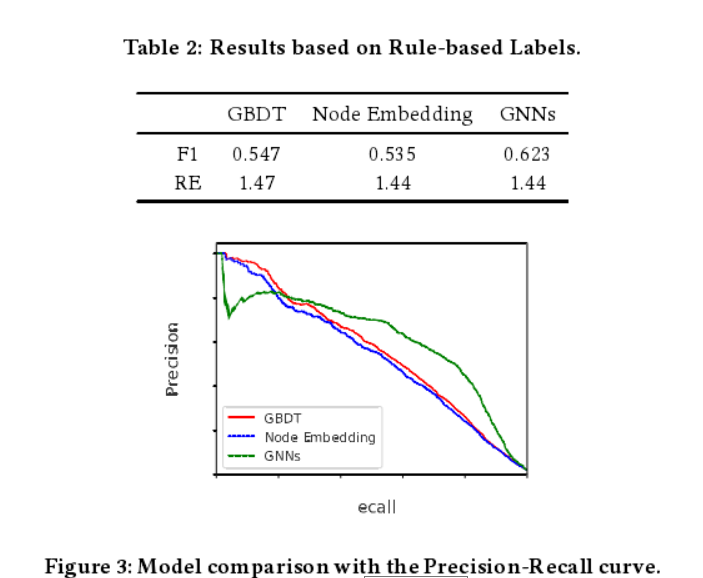
第五章 Experiment

**数据**：账号及其50个特征项

**标签：**基于规则判断出的“risk level”, 认为“high risk”的账户是确定的fraudulent accounts， “no observable risk”是regular accounts。 要特别说明的是，由于Label uncertainty，作者采用了Downsampling方法(具体见文中公式)来减少 “no observable risk”可能带来的误差影响。

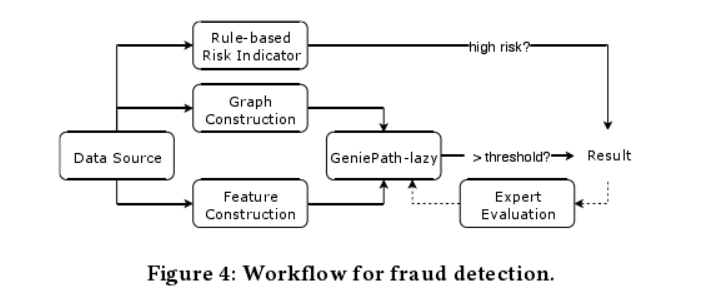
**训练：**作者用三种方法进行了实验：前两种是基于第四章介绍的节点向量化表示方法，第三种是根据账户特征直接进行训练。

**结果**：作者以F1之和RE值作为评估，发现GNN效果最优，GBDT直接对account features训练次之，Node2Vec最次。



第六章 Application

作者指出论文中的方法投入实际应用，并给出了Work Flow。



个人思考与总结

1. 以前较少读到基于图进行研究的文章，因此这篇文章让我觉得非常新颖，“图”的思想为以后其他问题的解决也能够提供新的思路；
2. Node Embedding: Node Embedding是继Word2Vec后提出的基于图进行embedding的方法，它能够将节点或边投影到低维向量空间中，再用于后续的任务。文中作者采用了Node2Vec，它是再最基本的Node Embedding(DeepWalk)上定义了bias random walk。关于Node Embedding作者在文中并未细致说明，这里附几个自己查找资料所得的相关链接：

[1]NE论文小览 <https://www.sohu.com/a/154650431_642762>

[2] 网络表示总述 <https://zhuanlan.zhihu.com/p/42022918>

[3] 网络学习概述 <https://zhuanlan.zhihu.com/p/32504175>

[4] Node2VEc Paper:

https://cs.stanford.edu/people/jure/pubs/node2vec-kdd16.pdf

1. GNNs: 文中作者使用了图神经网络，这也是比较新的应用。文中给出的相关说明较少，附一个自己查资料所得相关链接

[1] 图网络让神经网络也能做因果推理 <https://zhuanlan.zhihu.com/p/38071963>

(4) 作者对Label uncertainty提出了一种解决的办法，这在以后的研究与实验中是值得借鉴的；

(5) 从写作风格来看，十分简练，行文流畅，“干货”十足。

(6) 自己的想法：作者没有提到用“推理”，既然已经建立了图结构，如果基于图做一些推理不知能否会对结果有所提升。

另外一个问题：通过图表示方法得到节点向量后，还使用了accoutnt features吗？ 尤其是Node2Vec方法里面，是把 account features 和 embedding 一起作为GBDT 的输入吗？