Enhancing Graph Neural Network-based Fraud Detectors against Camouflaged Fraudsters

作者：Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, Philip S. Yu

Department of Computer Science, University of Illinois at Chicago School of Computer Science, Beijing University of Posts and Telecommunications Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University

Abstract

提出一种模型 $CAmouflages-REsistantGNN$（$CARE-GNN$），用于基于GNN的欺骗检测，特别是对抗使用伪装的欺骗者。

- 标签感知去找到信息丰富的邻居节点
- 利用强化学习去选择最佳邻居数量
- 将不同关系的选定邻居聚合在一起

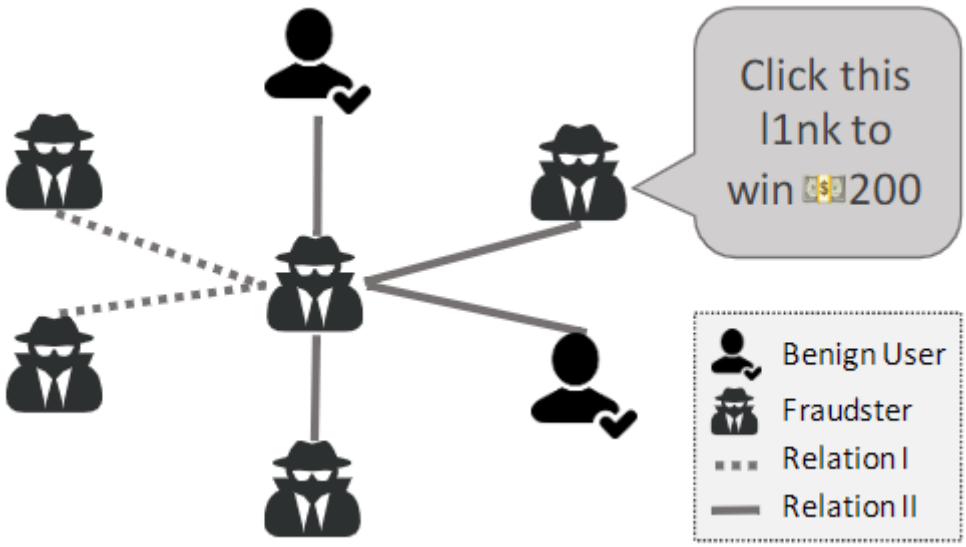
创新点/贡献/优势：

- 适应性：CARE-GNN给定任意的多重关系图，自适应地选择最佳邻居进行聚合
- 高效性：CARE-GNN具有较高的计算效率，无需attention和深入的强化学习
- 灵活性：可以将许多其他神经模块和外部知识插入到CARE-GNN

Introduction

特征伪装：加入特殊字符

关系伪装：欺骗者连接较多的良性用户



对于特征伪装，提出了一种标签感知的相似度度量，以基于节点特征找到最相似的邻居 。

对于关系伪装，设计了一个相似性感知的邻居选择器来选择中心节点的相似邻居 关系中，此外，我们利用强化学习（RL）以及GNN训练过程来自适应地找到最佳邻居选择阈值。

我们利用RL学习的邻居过滤阈值来制定区域感知邻居聚合器，该聚合器结合了来自不同关系的邻域信息并获得 最终的中心节点表示形式

Model

总体的结构如下：

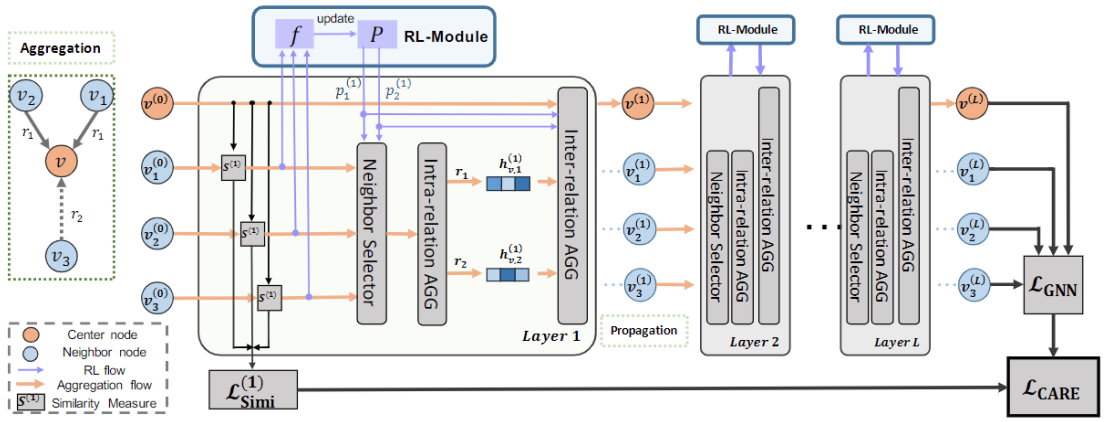


Figure 2: The aggregation process of proposed CARE-GNN at the training phase.

对于每一层都有邻居选择，关系感知邻居聚合器。

- 邻居选择包括：标签感知相似度度量、相似度感知邻居选择器。
- 关系感知邻居聚合器：内部关系聚合、相互关系聚合

标签感知相似度度量：

$$\mathcal{D}^{(l)}(v,v') = \left\| \sigma\left(MLP^{(l)}(\mathbf{h}_v^{(l-1)})\right) - \sigma\left(MLP^{(l)}(\mathbf{h}_{v'}^{(l-1)})\right) \right\|_1, \quad (2)$$

对于两个邻接节点，将上一层的特征经过MLP后在经过一个激活函数，将两个点做差

$$S^{(l)}(v,v') = 1 - \mathcal{D}^{(l)}(v,v'), \quad (3)$$

定义相似度为S

$$\mathcal{L}_{\text{Simi}}^{(l)} = \sum_{v\in\mathcal{V}} -\log\left(y_v \cdot \sigma\left(MLP^{(l)}(\mathbf{h}_v^{(l)})\right)\right). \quad (4)$$

定义损失函数，来调整MLP上的w

相似度感知邻居选择器：

对于每种关系的联系，去其S值，也就是相似度最高的top-p，使用强化学习来学习最佳的阈值 p_r^l 来筛选邻居节点

内部关系聚合：

$$\mathbf{h}_{v,r}^{(l)} = \text{ReLU}\left(\text{AGG}_r^{(l)}\left(\left\{\mathbf{h}_{v'}^{(l-1)} : (v,v') \in \mathcal{E}_r^{(l)}\right\}\right)\right), \quad (8)$$

AGG为任意聚合函数

相互关系聚合：

$$\mathbf{h}_v^{(l)} = \text{ReLU}\left(\text{AGG}_{all}^{(l)}\left(\mathbf{h}_v^{(l-1)} \oplus \{p_r^{(l)} \cdot \mathbf{h}_{v,r}^{(l)}\}|_{r=1}^{R}\right)\right), \quad (9)$$

聚合来自不同关系的邻居信息。 先前的方法采用注意力机制，以在从不同关系聚合信息时学习关系权重。 但是，假设我们在每个关系下选择了最相似的邻居，则注意系数在不同关系之间应相似。 因此，为了节省计算成本，同时保留相关重要性信息，我们直接将强化学习流程获得的最佳过滤阈值 $p_r^{(l)}$ 作为权重

定义损失函数

$$\mathcal{L}_{\text{GNN}} = \sum_{v \in \mathcal{V}} -\log\left(y_v \cdot \sigma(MLP(\mathbf{z}_v))\right). \quad (10)$$

$$\mathcal{L}_{\text{CARE}} = \mathcal{L}_{\text{GNN}} + \lambda_1 \mathcal{L}_{\text{Simi}}^{(1)} + \lambda_2 ||\Theta||_2, \quad (11)$$

Experiments

数据集

Table 2: Dataset and graph statistics.

	#Nodes (Fraud%)	Relation	#Edges	Avg. Feature Similarity	Avg. Label Similarity
Yelp	45,954 (14.5%)	*R-U-R*	49,315	0.83	0.90
		R-T-R	573,616	0.79	0.05
		R-S-R	3,402,743	0.77	0.05
		ALL	3,846,979	0.77	0.07
Amazon	11,944 (9.5%)	*U-P-U*	175,608	0.61	0.19
		U-S-U	3,566,479	0.64	0.04
		U-V-U	1,036,737	0.71	0.03
		ALL	4,398,392	0.65	0.05

Yelp数据集包含由Yelp过滤和推荐的酒店和餐厅评论。 Amazon数据集包括“乐器”类别下的产品评论

R-U-R: it connects reviews posted by the same user

R-S-R: it connects reviews under the same product with the same star rating (1-5 stars)

R-T-R: it connects two reviews under the same product posted in the same month

U-P-U: it connects users reviewing at least one same product

U-S-V: it connects users having at least one same star rating within one week

U-V-U: it connects users with top 5% mutual review text similarities (measured by TF-IDF) among all users.

可以发现标签还是很重要的

实验结果

Table 3: Fraud detection performance (%) on two datasets under different percentage of training data.

	Metric	Train%	GCN	GAT	RGCN	Graph-SAGE	Genie-Path	Player-2Vec	Semi-GNN	Graph-Consis	CARE-*Att*	CARE-*Weight*	CARE-*Mean*	CARE-GNN
Yelp	AUC	5%	54.98	56.23	50.21	53.82	56.33	51.03	53.73	61.58	66.08	71.10	69.83	**71.26**
		10%	50.94	55.45	55.12	54.20	56.29	50.15	51.68	62.07	70.21	71.02	71.85	**73.31**
		20%	53.15	57.69	55.05	56.12	57.32	51.56	51.55	62.31	73.26	74.32	73.32	**74.45**
		40%	52.47	56.24	53.38	54.00	55.91	53.65	51.58	62.07	74.98	74.42	74.77	**75.70**
	Recall	5%	53.12	54.68	50.38	54.25	52.33	50.00	52.28	62.60	63.52	66.64	**68.09**	67.53
		10%	51.10	52.34	51.75	52.23	54.35	50.00	52.57	62.08	67.38	68.35	**68.92**	67.77
		20%	53.87	53.20	50.92	52.69	54.84	50.00	52.16	62.35	68.34	69.07	**69.48**	68.60
		40%	50.81	54.52	50.43	52.86	50.94	50.00	50.59	62.08	71.13	70.22	69.25	**71.92**
Amazon	AUC	5%	74.44	73.89	75.12	70.71	71.56	76.86	70.25	85.46	89.49	89.36	89.35	**89.54**
		10%	75.25	74.55	74.13	73.97	72.23	75.73	76.21	85.29	**89.58**	89.37	89.43	89.44
		20%	75.13	72.10	75.58	73.97	71.89	74.55	73.98	85.50	89.58	**89.68**	89.34	89.45
		40%	74.34	75.16	74.68	75.27	72.65	56.94	70.35	85.50	89.70	89.69	89.52	**89.73**
	Recall	5%	65.54	63.22	64.23	69.09	65.56	50.00	63.29	85.49	88.22	88.31	88.02	**88.34**
		10%	67.81	65.84	67.22	69.36	66.63	50.00	63.32	85.38	87.87	**88.36**	88.12	88.29
		20%	66.15	67.13	65.08	70.30	65.08	50.00	61.28	85.59	88.40	**88.60**	88.00	88.27
		40%	67.45	65.51	67.68	70.16	65.41	50.00	62.89	85.53	88.41	88.45	88.22	**88.48**

（CARE-Att, CARE-Weight, and CARE-Mean,and they differ from each other in Attention, Weight, and Mean inter-relation aggregator respectively）

Graph-Consis也有较好的表现，因为其也有对于邻居节点的筛选的过程

多种聚合方式的比较

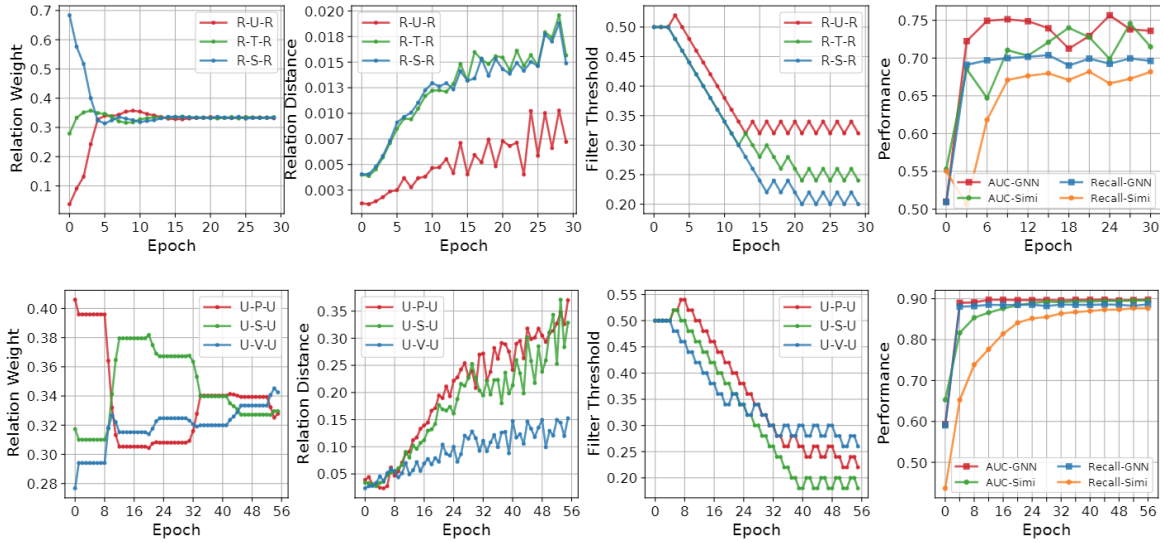


Figure 3: The training process and testing performance of CARE-*Weight* on Yelp (upper) and Amazon (lower) dataset.

第4列图 显示了对于两种测试集的测试性能。 对于Yelp数据集，GNN具有比相似性度量更好的AUC和Recall，这表明利用结构信息有利于模型对欺诈和良性实体进行分类。对于亚马逊来说，GNN的性能和相似性度量可以相互媲美。 这是因为输入功能提供了足够的信息来区分欺诈者

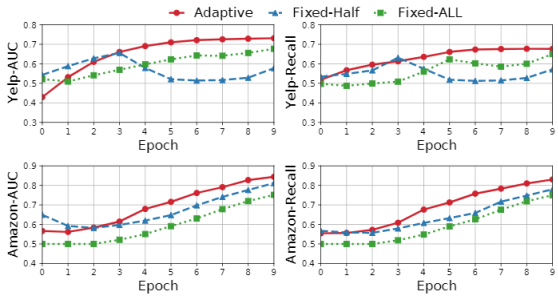


Figure 4: The testing AUC and Recall for CARE-GNN with different neighbor filtering methods during training.

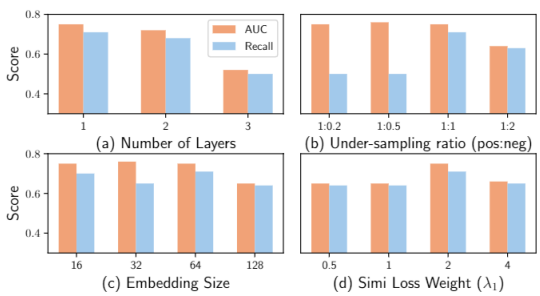


Figure 5: Parameter Sensitivity. For each parameter configuration, only the best results among 30 epochs are recorded.

多层的模型，适用于稀疏图

Conclusion

本文研究了欺诈者的伪装行为及其对基于GNN的欺诈检测器的对抗作用。 为了增强针对欺诈者的特征伪装和关系伪装的基于GNN的欺诈检测器，我们提出了一种使用强化学习的标签感知相似度度量和相似感知邻居选择器。 连同两个神经模块，我们进一步提出了一个关系感知聚合器，以最大化计算的实用性。