Alleviating the Inconsistency Problem of Applying Graph Neural Network to Fraud Detection

SIGIR'20 , Virtual Event, China

Zhiwei Liu, Yingtong Dou, Philip S. Yu (University of Illinois at Chicago)

Yutong Deng (Beijing University of Posts and Telecommunications)

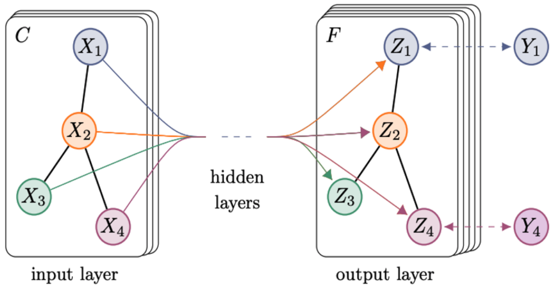
Hao Peng (Beihang University)

Code: https://github.com/safe-graph/DGFraud.git

Background

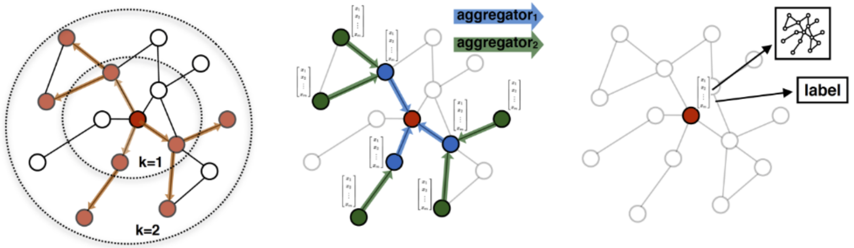
Graph Neural Network (GNN)

GCN[1]



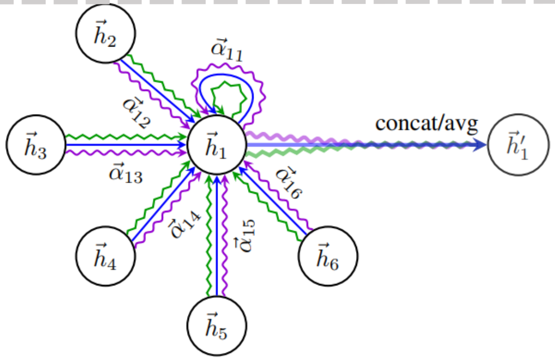
- Directly aggregate neighbors using Laplacian adjacency matrix

GraphSAGE[2]



- Sample and aggregate neighbors

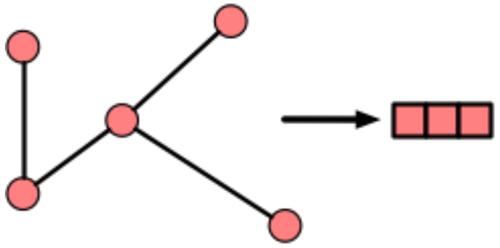
GAT[3]



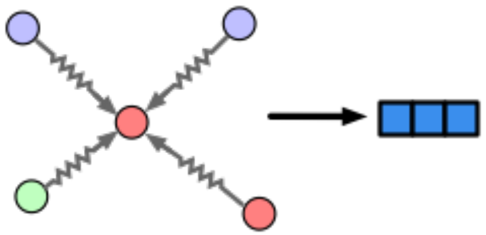
- Attentively aggregate neighbors

[1] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.
[2] W. Hamilton, Hamilton, William L. Ying, Rex Leskovec, Jure. Inductive Representation Learning on Large Graphs , NIPS 2017
[3] Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.

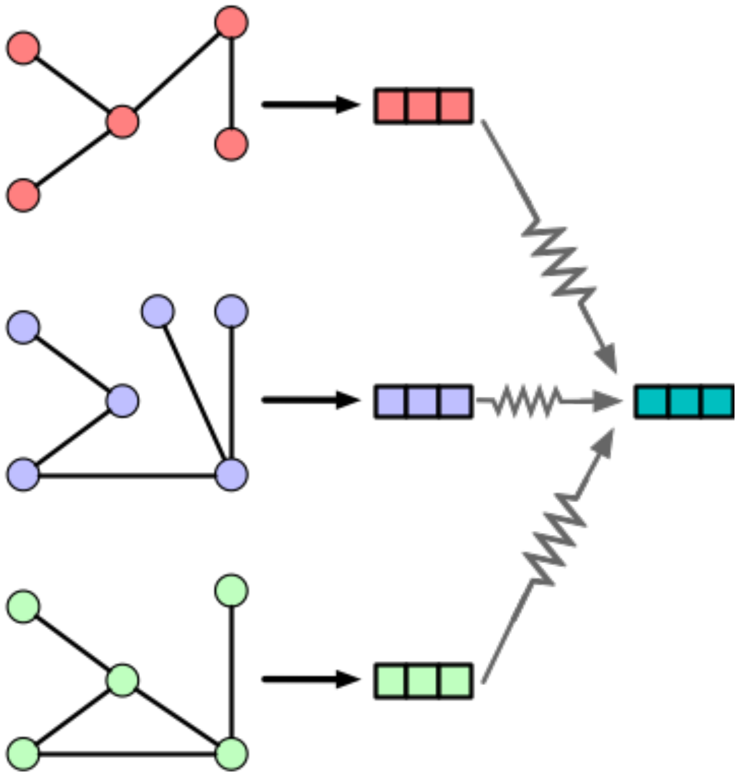
GNN-based Fraud Detectors



FdGars[1] (GCN-based)



GAS[2] (GAT-based)



Player2Vec[3] (Hybrid)

[1] Wang, J., Wen, R., Wu, C., Huang, Y. and Xion, J., 2019, May. Fdgars: Fraudster detection via graph convolutional networks in online app review system. WWW 2019.
[2] Li, A., Qin, Z., Liu, R., Yang, Y. and Li, D., 2019, November. Spam review detection with graph convolutional networks. CIKM 2019.
[3] Zhang, Y., Fan, Y., Ye, Y., Zhao, L. and Shi, C., 2019, November. Key Player Identification in Underground Forums over Attributed Heterogeneous Information Network Embedding Framework. CIKM 2019

Motivation

Motivation

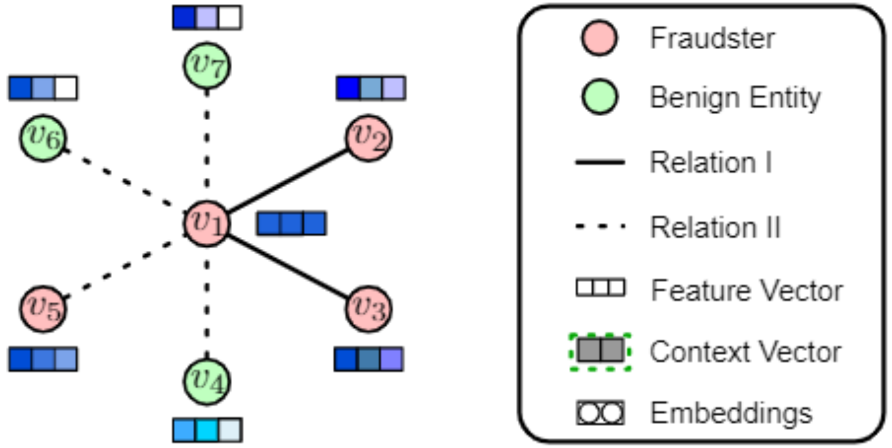
- Assumption of GNN: neighbors share similar features, context, and labels (smoothness[1])
- This assumption can no longer hold in fraud detection task, i.e., inconsistency problem

[1] Hou, Yifan, et al. "Measuring and improving the use of graph information in graph neural networks." *International Conference on Learning Representations*. 2019.

Motivation

- **Context inconsistency:** v_1 staying with three begin nodes (4,6,7)
- **Feature inconsistency:** v_1 having features of great difference to others
- **Relation inconsistency:** under relation I (solid), v_1 only connecting to other fraudsters; while under relation II(dash), v_1 connecting to three benign nodes.

Direct aggregation results in the loss of information! We should design a new GNN model to handle these inconsistencies.

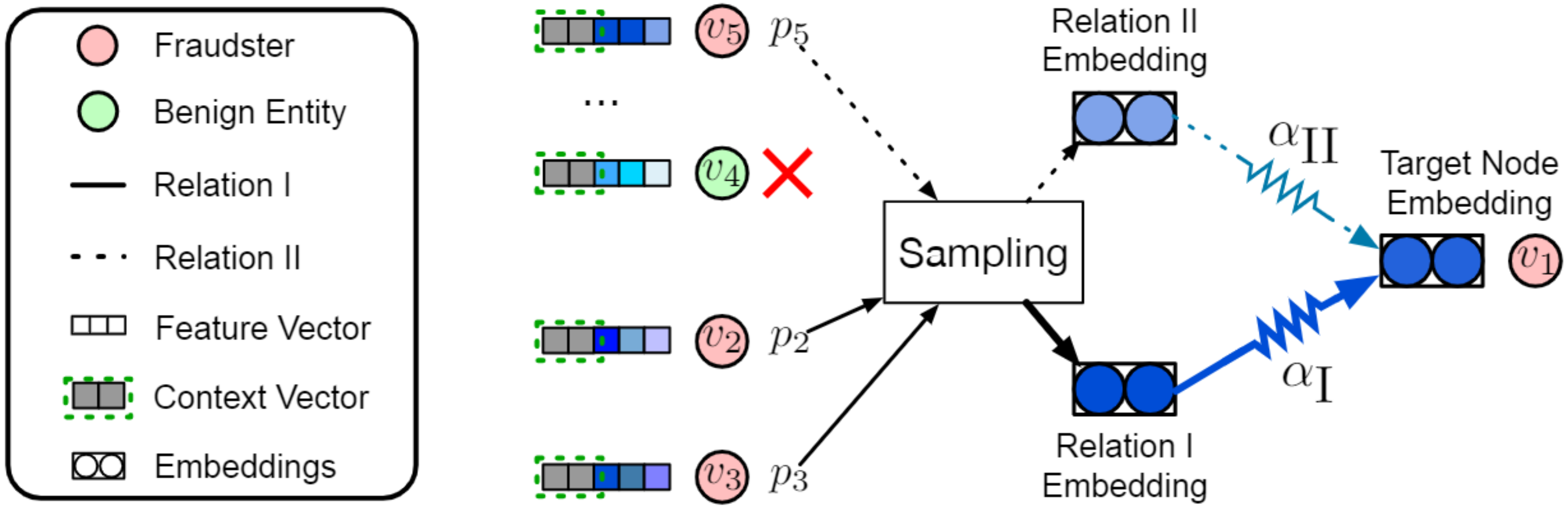


Left: Inconsistency Problem

Proposed Model (GraphConsis)

GraphConsis

- Context Embedding: trainable context embedding
- Embedding consistency measurement
  - Ignoring inconsistent neighbors
  - Generating consistent sampling probabilities
- Relation Attention: specially dealing with various relations

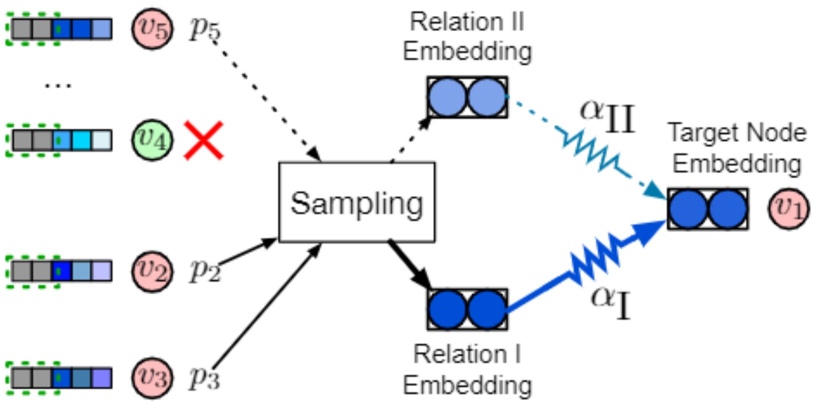


GraphConsis

- General GNN structure

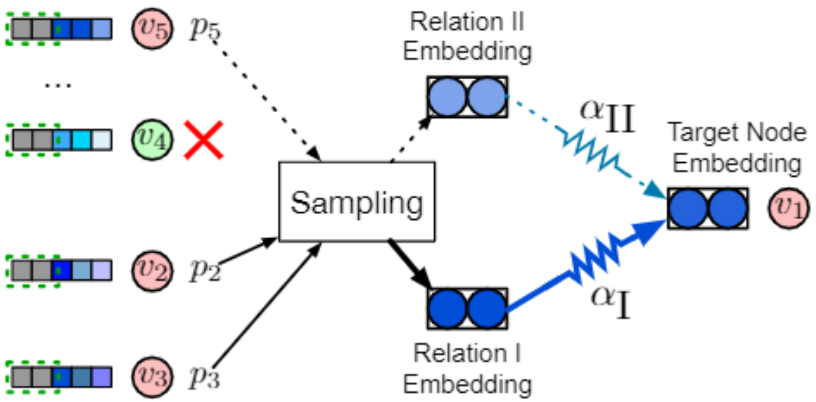
Sampled neighbors

$$\mathbf{h}_v^{(l)} = \mathbf{h}_v^{(l-1)} \oplus \mathrm{AGG}^{(l)}\left(\left\{\mathbf{h}_{v'}^{(l-1)} : v' \in \boxed{\mathcal{N}_v}\right\}\right)$$



Right: Proposed **GraphConsis** Model

GraphConsis



Right: Proposed **GraphConsis** Model

- Context Embedding

$$\mathbf{h}_v^{(1)} = \{\mathbf{x}_v \| \mathbf{c}_v\} \oplus \mathrm{AGG}^{(1)}\left(\left\{\mathbf{x}_{v'} \| \mathbf{c}_{v'} : v' \in \mathcal{N}_v\right\}\right)$$

- Sampling Probability

Consistency score

$$p^{(l)}(u;v) = s^{(l)}(u,v) / \sum_{u \in \tilde{\mathcal{N}}_v} \boxed{s^{(l)}(u,v)} \longrightarrow s^{(l)}(u,v) = \exp\left(-\|\mathbf{h}_u^{(l)} - \mathbf{h}_v^{(l)}\|_2^2\right)$$

- Relation Attention

$$\mathrm{AGG}^{(l)}\left(\left\{\mathbf{h}_q^{(l-1)}\right\}\Big|_{q=1}^{Q}\right) = \sum_{q=1}^{Q} \alpha_q^{(l)} \mathbf{h}_q^{(l)}$$

Q: # of samples; α_q :a scalar denoting the attention weights of q-th sample

$$\alpha_q^{(l)} = \exp\left(\sigma\left(\{\mathbf{h}_q^{(l)} \| \mathbf{t}_{r_q}\}\mathbf{a}^\top\right)\right) / \sum_{q=1}^{Q} \exp\left(\sigma\left(\{\mathbf{h}_q^{(l)} \| \mathbf{t}_{r_q}\}\mathbf{a}^\top\right)\right)$$

$\boldsymbol{t}_{r_q}$: the relation vector of relation r (of q sample); $\boldsymbol{a}^T$: the weight of attention layer

Experiment

Inconsistency problem

Context Characteristic Score (Label smoothness)

Indication of whether node u,v have the same label

$$\gamma_r^{(c)} = \sum_{(u,v)\in E_r} \left(1 - \boxed{\mathbb{I}(u \sim v)}\right) / |E_r|$$

Feature Characteristic Score (feature smoothness)

$$\gamma_r^{(f)} = \sum_{(u,v)\in E_r} \exp\left(-\|\mathbf{x}_u - \mathbf{x}_v\|_2^2\right) / |E_r| \cdot d,$$

Yelp data
29431 users,
182 products, and
45954 reviews

Table 1: The statistics of different graphs.

	Graph	#Nodes	#Edges	$\gamma^{(f)}$	$\gamma^{(c)}$
Others	**Cora**	2,708	5,278	0.72	0.81
	PPI	14,755	225,270	0.48	0.98
	Reddit	232,965	11,606,919	0.70	0.63
Ours	***R-U-R***	45,954	98,630	0.83	0.90
	R-T-R	45,954	1,147,232	0.79	0.05
	R-S-R	45,954	6,805,486	0.77	0.05
	Yelp-ALL	45,954	7,693,958	0.77	0.07

Overall Comparison

Table 2: Experiment results under different training %.

Method	40%		60%		80%	
	F1	AUC	F1	AUC	F1	AUC
LR	0.4647	**0.6140**	0.4640	0.6239	0.4644	0.6746
GraphSAGE	0.4956	0.5081	0.5127	0.5165	0.5158	0.5169
FdGars	0.4603	0.5505	0.4600	0.5468	0.4603	0.5470
Player2Vec	0.4608	0.5426	0.4608	0.5697	0.4608	0.5403
GraphConsis	**0.5656**	0.5911	**0.5888**	**0.6613**	**0.5776**	**0.7428**

- Observations
  - LR is better than other GNNs
  - GraphConsis performs better than other baselines
  - Increasing training data improves GraphConsis a lot

Implementations

- Code: https://github.com/safe-graph/DGFraud.git
- We also reproduced some GNN-based fraud detector

DGFraud

build passing | license Apache-2.0 | downloads 0 | release v1.0-alpha

A Deep Graph-based Toolbox for Fraud Detection

Introduction

DGFraud is a Graph Neural Network (GNN) based toolbox for fraud detection. It integrates the implementation & comparison of state-of-the-art GNN-based fraud detection models. It also includes several utility functions such as graph preprocessing, graph sampling, and performance evaluation. The introduction of implemented models can be found here.

We welcome contributions on adding new fraud detectors and extending the features of the toolbox. Some of the planned features are listed in TODO list.

If you use the toolbox in your project, please cite the paper below and the algorithms you used :

Implemented Models

Model	Paper	Venue	Reference
SemiGNN	A Semi-supervised Graph Attentive Network for Financial Fraud Detection	ICDM 2019	BibTex
Player2Vec	Key Player Identification in Underground Forums over Attributed Heterogeneous Information Network Embedding Framework	CIKM 2019	BibTex
GAS	Spam Review Detection with Graph Convolutional Networks	CIKM 2019	BibTex
FdGars	FdGars: Fraudster Detection via Graph Convolutional Networks in Online App Review System	WWW 2019	BibTex
GeniePath	GeniePath: Graph Neural Networks with Adaptive Receptive Paths	AAAI 2019	BibTex
GEM	Heterogeneous Graph Neural Networks for Malicious Account Detection	CIKM 2018	BibTex

Discussion

Conclusion and Future Work

- Conclusion
  - Investigate three inconsistencies (context, feature, and relation)
  - Design three mechanisms in GraphConsis
- Future work
  - General inconsistencies?
  - Adaptive sampling?
  - Other consistency metrics?

Thanks!