

Graph Neural Network-based Fraud Detection *from Research to Application*

Yingtong Dou
University of Illinois at Chicago

Email: ydou5@uic.edu

Homepage: <http://ytongdou.com>

Project Page: <https://github.com/safe-graph>

Wechat: ytongdou

Twitter: [@dozee_sim](https://twitter.com/dozee_sim)

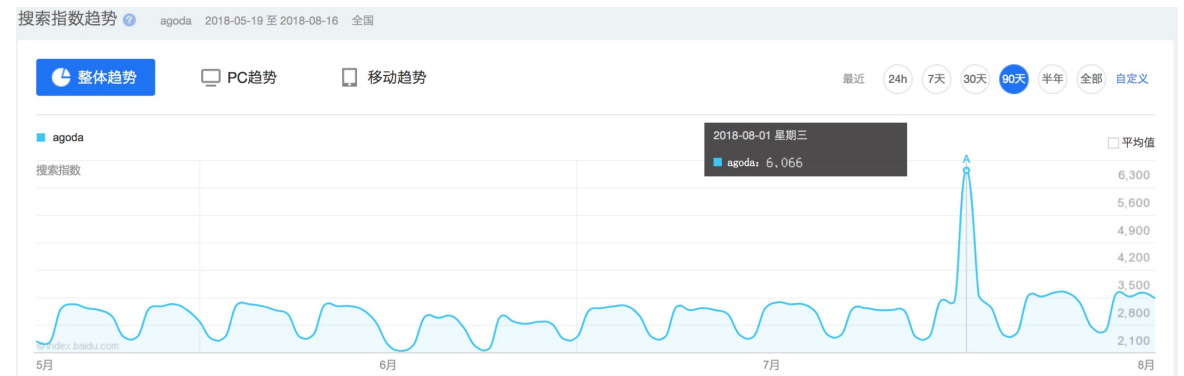


Outline

- **Background:** Graph Neural Network & Fraud Detection.
- **Research:** A History (w/ Highlights) of GNN-based Fraud Detection Research.
- **Application:** The Guideline for Applying GNNs to Fraud Detection.
- **Resources:** Dataset, Toolbox, Paper List, etc..
- **Q&A**

What is Fraud?

- **Fraud definition according to U.S. Law:**
 - a misrepresentation of a fact, made from one person to another, with knowledge of its falsity and for the purpose of inducing the other to act.
- **Fraudster vs. Hacker**
 - Most fraudsters are **NOT** hackers.
 - Only few hackers are fraudsters.
- **Fraud vs. Anomaly**
 - Not all frauds are anomalies.
 - Not all anomalies are frauds.



Fraud Types in 2021

Social Network

- Fake Reviews
- Social Bots
- Misinformation
- Disinformation
- Fake Accounts
- Social Sybils
- Link Advertising

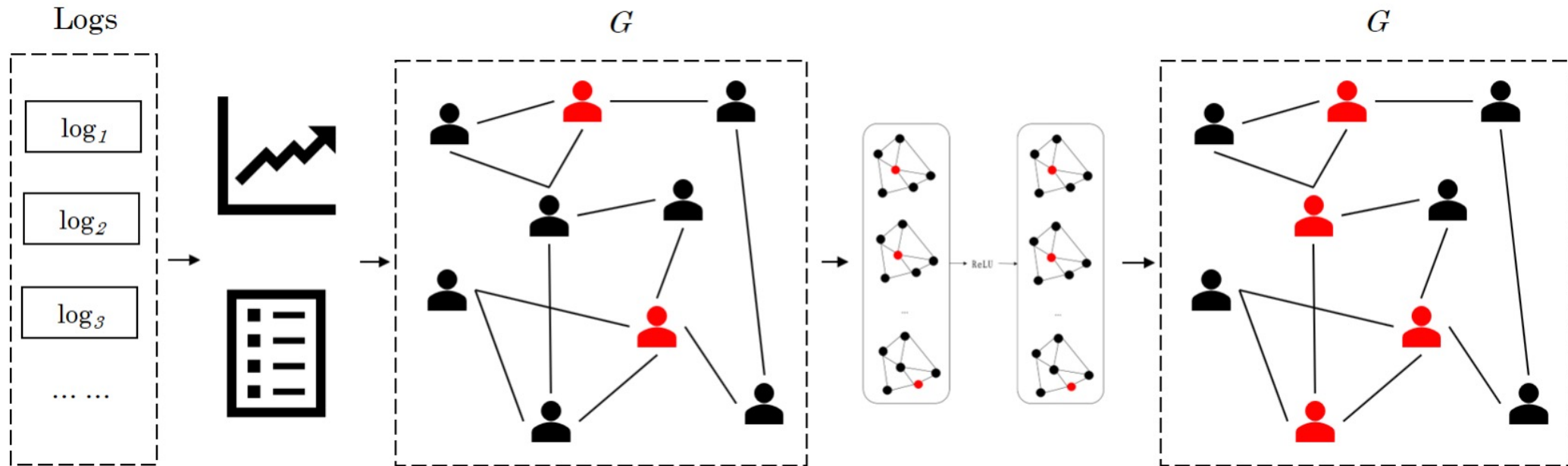
Finance

- Insurance Fraud
- Loan Defaulter
- Money Laundering
- Malicious Account
- Transaction Fraud
- Cash-out User
- Bitcoin Fraud

Others

- Advertisement
- Mobile Apps
- Ecommerce
- Crowdturfing
- Fake Clicks
- Game
- Account Takeover

GNN-based Fraud Detection



(1) Graph Construction.

(2) Training GNN on the Graph.

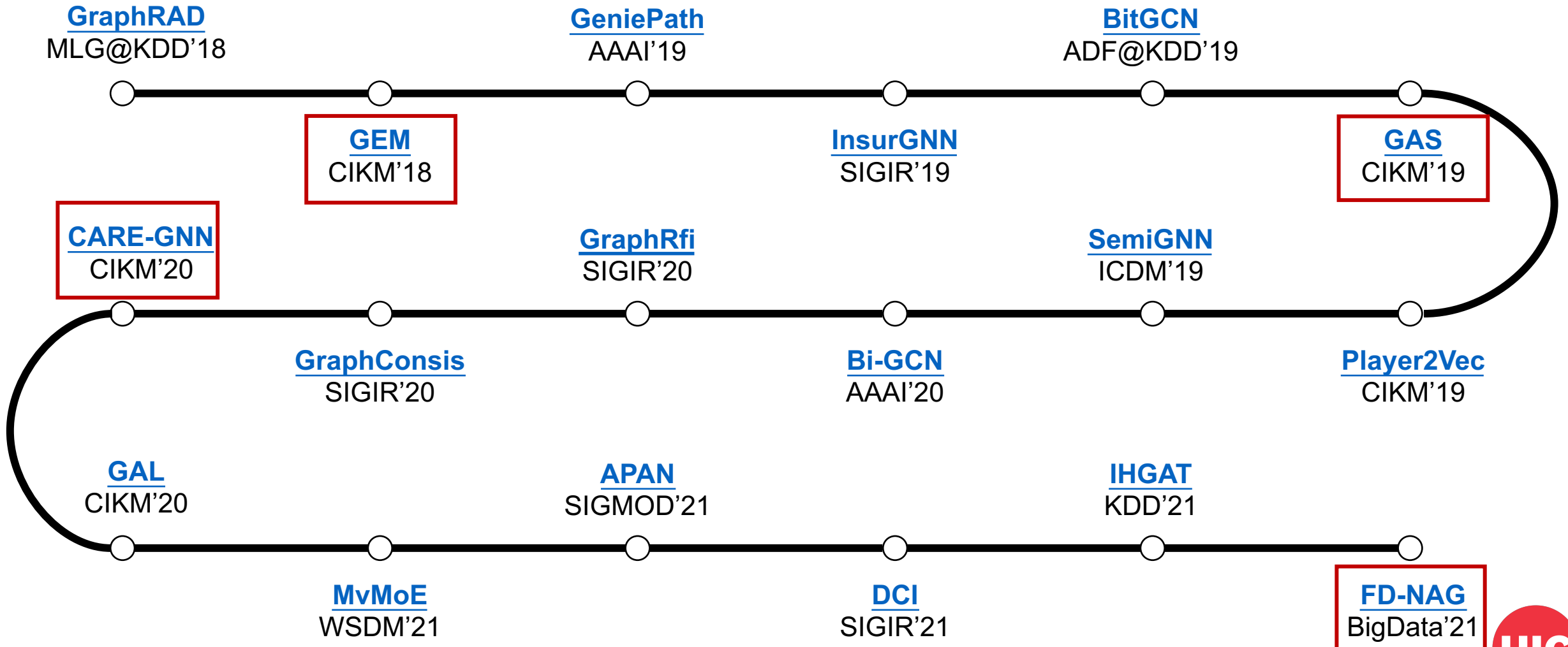
(3) Classifying Unlabeled Nodes.

Key idea: the connected nodes are similar (homophily assumption)

Background

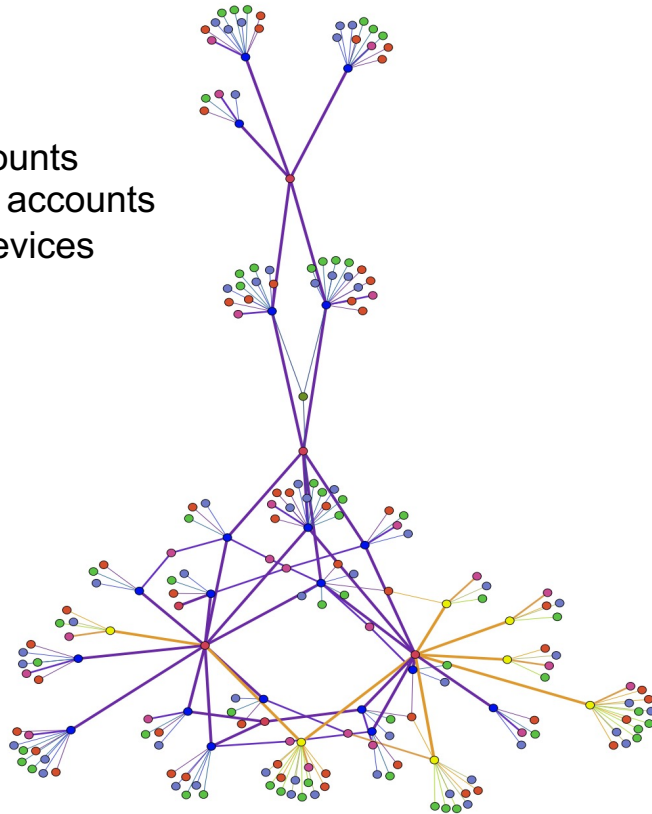
Research

A History of GNN Fraud Detection (75+Papers)



GEM (CIKM'18)

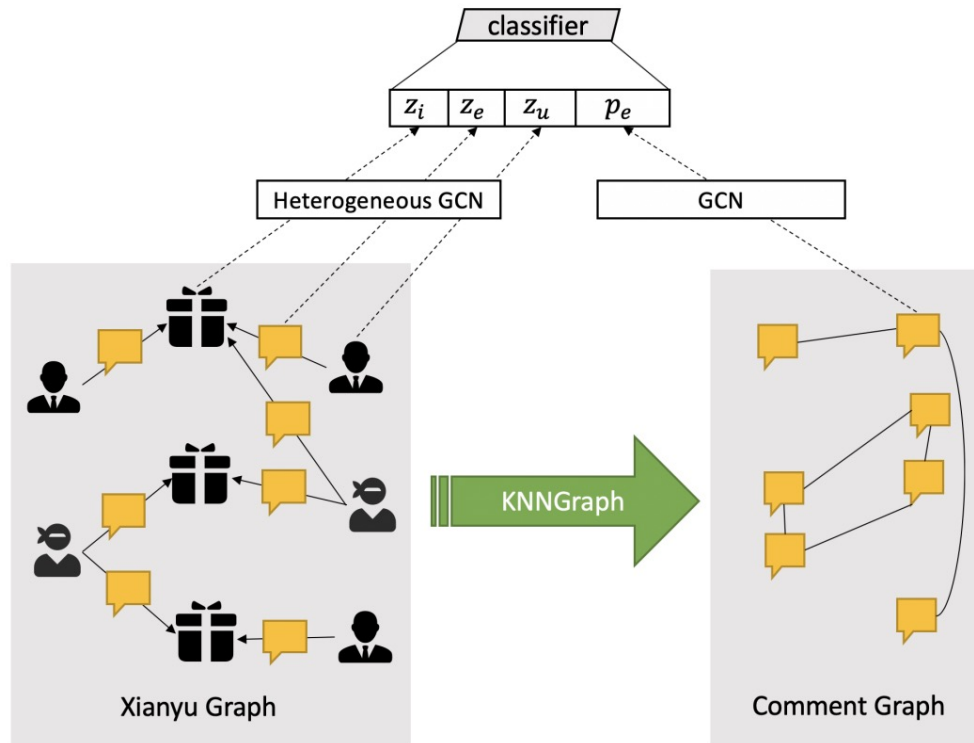
Blue: normal accounts
Yellow: malicious accounts
Other: different devices



**Account-Device
Heterogeneous Graph**

- Task: malicious accounts detection in Alipay.
- **The first paper leveraging the heterogeneous graph for fraud detection.**
- Device types include UMID, MAC address, IMSI, APDID (Alipay Fingerprint).
- Using attention mechanism to learn importance of different sub-graphs.
- Code is [available](#).

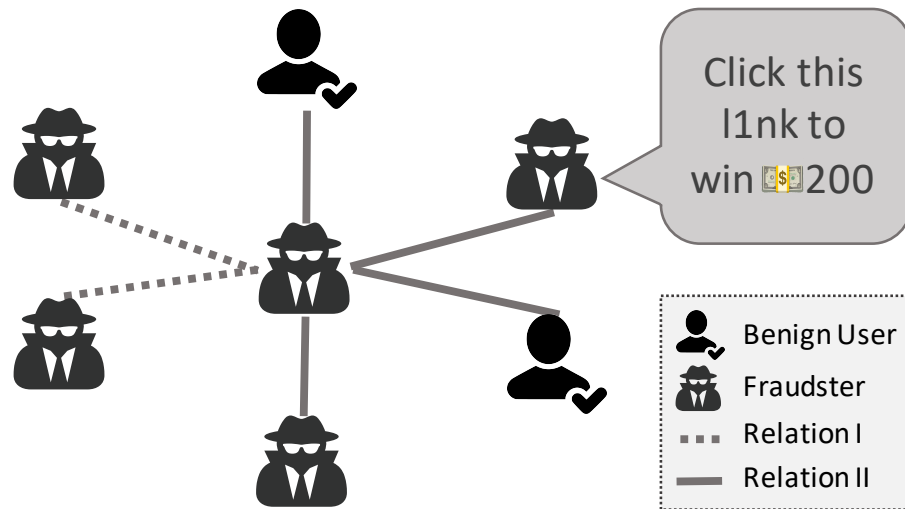
GAS (CIKM'19)



User-Comment-Product Graph
+
Comment-Comment Graph

- Task: spam review detection on the Xianyu Platform.
- CIKM'19 Industrial Track Best Paper.
- **Novel graph construction approach. Encoding each heterogeneous entity separately.**
- Verifying a sampling approach for graph construction.
- Code is [available](#).

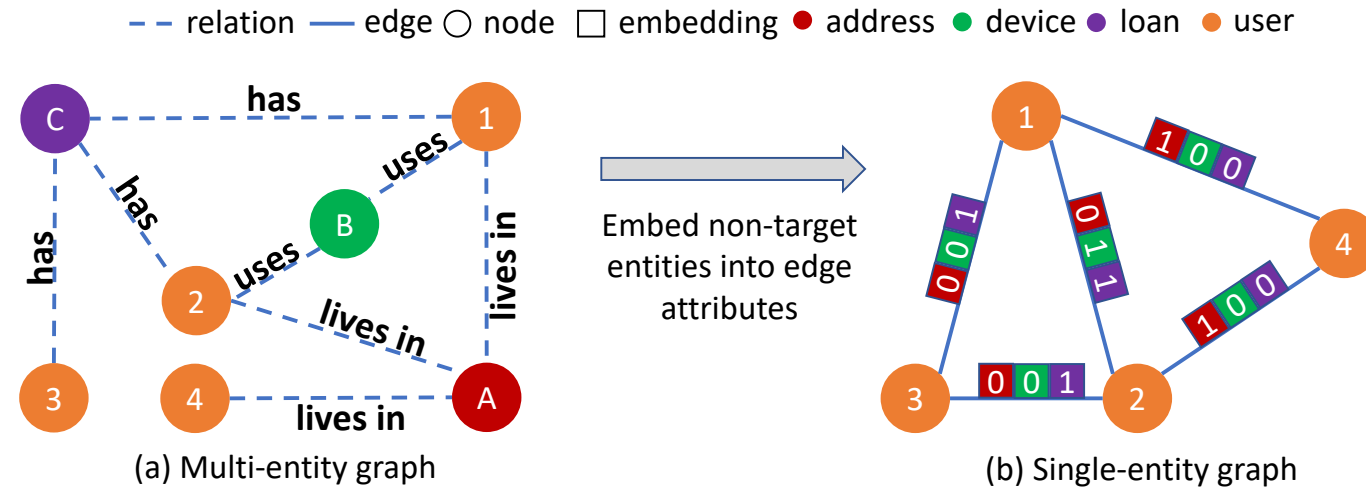
CARE-GNN (CIKM'20)



Fraudster Camouflage

- Task: spam review detection on Yelp; malicious reviewer detection on Amazon.
- Top 15 influential papers in CIKM'20.
- **Addressing the fraudster camouflage and class-imbalance problems in fraud detection.**
- Using reinforcement learning to select the most informative neighbors for GNNs.
- Integrated with [DGL](#), introduce two **public datasets**.
- Extended [TOIS version](#) with more applications.
- Code is [available](#).

FD-NAG (BigData'21)



Transferring a heterogeneous non-attributed graph to an edge-attributed homogeneous graph

- Task: fraudsters detection in ride sharing services.
- Designing node and edge features for non-attributed graphs.
- Empirically verified the effectiveness of **contrastive learning** in fraud detection.



Applying GNN into Fraud Detection

- **Problem Formulation**

- Whether using Graph&GNNs?
- Which task to choose?
- What graph schema is suitable?

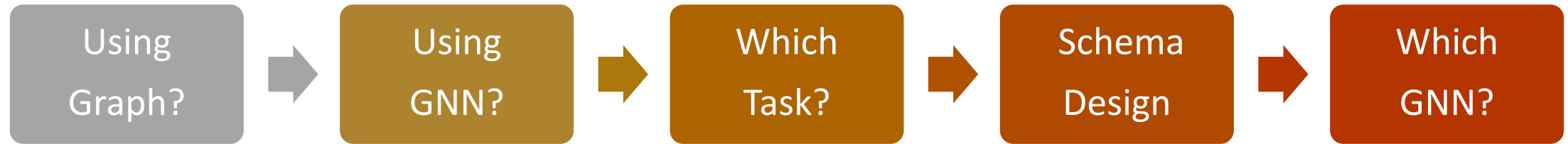
- **Key Issues and Solutions**

- Camouflage, Scalability, Class imbalance
- Label scarcity, Label fidelity, Data-scarcity

- **Novel Practices**

- Novel methods
- Industrial cases

Problem Formulation



- **Using Graph?**

- The fraudsters share common entities.
- The fraudsters have clustering behavior.
- The trade off between cost and effectiveness.

- **Using GNN?**

- The infrastructure.
- The feature availability and feature types.
- Integrating with other modules and tasks.

- **Which Task?**

- Node/edge/graph/subgraph classification.
- Community detection; anomaly detection.

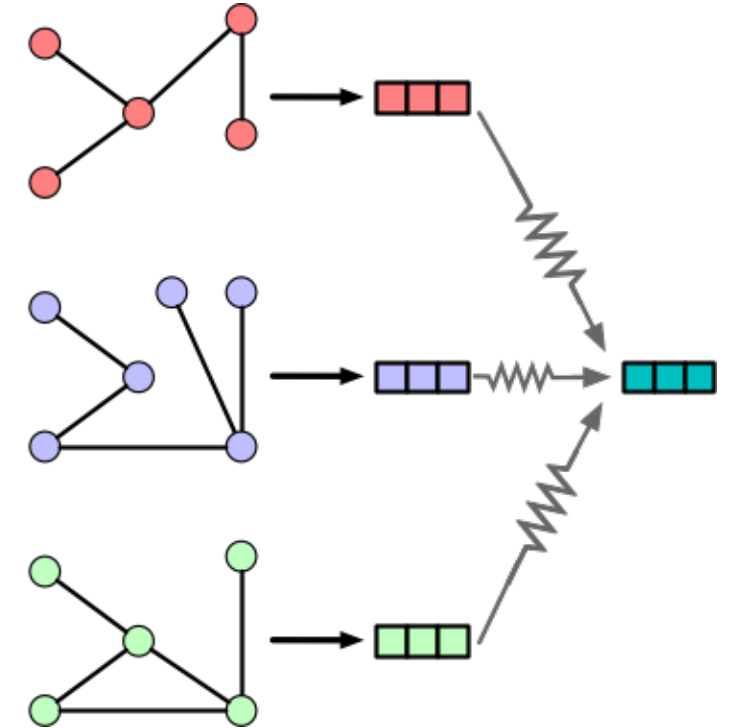
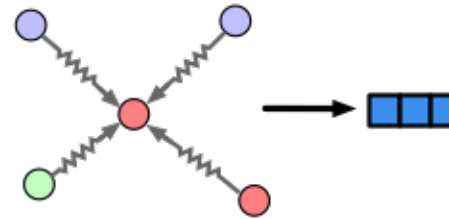
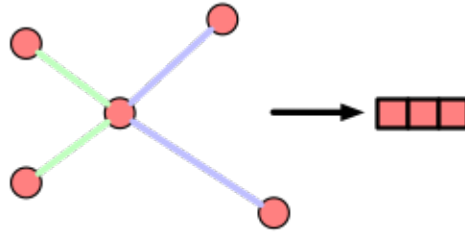
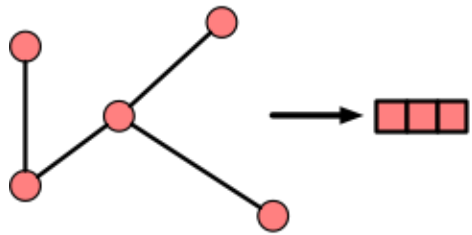
- **Schema Design**

- Node/edge type and node/edge feature.
- Graph schema, node sampling.
- Graph structure is flexible: [SIGIR'19](#), [ICDM'20](#).

- **Which GNN?**

- GNN is chosen based on task and schema.
- Simple GNN model is enough.
- GAT and Graph-SAGE are commonly used.

Graph Schema



Homogeneous

BitGCN
FdGars
GeniePath
FD-NAG

Multi-relation

GraphConsis
CARE-GNN
PC-GNN

Heterogeneous

GAS
mHGNN
IHGAT

Hierarchical

GEM
SemiGNN
Player2Vec
AA-HGNN

Key Issues and Solutions

- **Camouflage**

- Neighboring filtering: [SIGIR'20](#), [CIKM'20](#), [WWW'21](#).
- Aware of adversarial behavior: [IJCAI'20](#), [WWW'20](#).
- Active generative learning: [ACL'20](#).
- Bayesian edge weight inference: [ACL'21](#).

- **Scalability**

- GNN scalability: [MLF@KDD'20](#).
- Shallow graph models are more scalable: [MLG@KDD'18](#), [WWW'20](#).

- **Class imbalance**

- Under-sampling: [CIKM'20](#).
- Neighbor selection: [WWW'21](#).
- Data augmentation: [CIKM'20](#).

Key Issues and Solutions (Cont'd)

- **Label scarcity**

- Active learning: [ICDM'20](#), [TNNLS'21](#).
- Ensemble learning: [CIKM'20](#).
- Meta learning: [WSDM'21](#).

- **Label fidelity**

- Active learning: [TNNLS'21](#).
- Human-in-the-loop: [AAAI'20](#).

- **Data scarcity**

- Data augmentation: [CIKM'20](#), [CIKM'21](#), [ACL'20](#).

Novel Practices

- **Graph Pretraining (Contrastive Learning)**
 - Fraudster is distinguishable from its structural pattern.
 - [TNNLS'21](#), [SIGIR'21](#), [arXiv'21\(1\)](#), [arXiv'21\(2\)](#).
- **Dynamic/Temporal/Streaming Graph**
 - The historical information is useful for identifying fraudsters.
 - The efficiency and cost are bottlenecks.
 - [CIKM'21](#), [KDD'21\(1\)](#), [KDD'21\(2\)](#), [SIGMOD'21](#).
 - [arXiv'21](#), [SDM'21](#), [ICDM'20](#), [KDD'20](#).
 - [ROLAND](#).

Novel Practices (Cont'd)

- **Multi-task Learning**

- Credit limit forecasting and credit risk predicting: [WSDM'21](#).
- Fraud detection and recommender system: [SIGIR'20](#).

- **Explainable Fraud Detection**

- Explainable fraud transaction detection: [arXiv'20](#), [KDD'21](#).
- Explainable fake news detection: [ACL'20](#).

- **Recent Surveys**

- [A Comprehensive Survey on Graph Anomaly Detection with Deep Learning](#).
- [Anomaly Mining - Past, Present and Future](#).
- [Graph Computing for Financial Crime and Fraud Detection: Trends, Challenges and Outlook](#).

Industrial Cases

- Facebook
 - [WWW'20](#), [KDD'20](#), [Security'21](#).
- Amazon
 - [MLG@KDD'18](#), [KDD'21](#).
- Tencent
 - [WWW'19](#), [WWW'20](#), [KDD'21](#).
- Alibaba & Ant Group
 - [CIKM'18](#), [AAAI'19](#), [SIGIR'19](#), [CIKM'19](#), [ICDM'19](#), [IJCAI'20](#), [ACL'20](#), [CIKM'20\(1\)](#).
 - [CIKM'20\(2\)](#), [SIGMOD'21](#), [WSDM'21](#), [WWW'21](#), [AAAI'21](#), [KDD'21\(1\)](#), [KDD'21\(2\)](#).
- eBay
 - [Workshop@AAAI'21](#), [arXiv'20](#), [MLF@KDD'20](#).
- Others
 - [App Market](#), [Money Laundering](#), [Fake Invitation \(iQIYI\)](#), [Bitcoin](#), [Grab](#).

SafeGraph (<https://github.com/safe-graph>)

- **DGFraud**: a GNN-based fraud detection toolbox implemented TensorFlow 1.X.
 - 360 stars, ten GNN models.
- **DGFraud-TF2**: a GNN-based fraud detection toolbox implemented TensorFlow 2.X.
 - 31 stars, nine GNN models.
- **UGFraud**: an unsupervised graph-based fraud detection toolbox.
 - 71 stars, six classic models, deployed on Pypi.
- **GNN-FakeNews**: a collection of GNN-based fake news detection models.
 - 80 stars, benchmarking GNN-based fake news detection, integrated with DGL and PyG.
- **Graph-based Fraud Detection Paper List** (frequently updated).
 - 466 stars, more than 100 papers listed plus code, datasets, surveys, and other resources.
- **Graph Adversarial Learning Literature** (frequently updated).
 - 455 stars, more than 200 papers surveyed.

Other Toolboxes

- PyOD: Python Outlier Detection
 - <https://github.com/yzhao062/pyod>.
- PyODD: An End-to-end Outlier Detection System
 - <https://github.com/datamllab/pyodds>.
- DGL Fraud Detection Pipeline
 - <https://github.com/awslabs/realtime-fraud-detection-with-gnn-on-dgl>.
- PyG 2.0: A PyTorch-based GNN Library
 - https://github.com/pyg-team/pytorch_geometric.

Other Resources

- KDD Machine Learning in Finance Workshop
 - <https://sites.google.com/view/kdd-mlf-2021>.
- KDD Machine Learning on Graph Workshop
 - <http://www.mlgworkshop.org/>.
- KDD'20 Deep Anomaly Detection Tutorial
 - <https://sites.google.com/view/kdd2020deepeye/home>.
- Awesome Fraud Detection Papers
 - <https://github.com/benedekrozemberczki/awesome-fraud-detection-papers>.

Thanks!

Yingtong Dou
University of Illinois at Chicago

Email: ydou5@uic.edu

Homepage: <http://ytongdou.com>

Project Page: <https://github.com/safe-graph>

Wechat: ytongdou

Twitter: [@dozee_sim](https://twitter.com/dozee_sim)

