Università della Svizzera italiana
Faculty of Informatics
Institute of Computing CI

GAMLNet: a graph based framework for the detection of money laundering

Julien Schmidt, Dimosthenis Pasadakis, Madan Sathe*, Olaf Schenk

* Deloitte AG, Forensics.

- J. Schmidt, D. Pasadakis, M. Sathe, and O. Schenk, "GAMLNet: a graph based framework for the detection of money laundering," 11th IEEE Swiss Conference on Data Science (SDS), Zurich, Switzerland, 2024, in press.

➜ **We identify money laundering activity with a GNN model that exploits structurally aware & statistically significant features.**

Financial anomaly detection

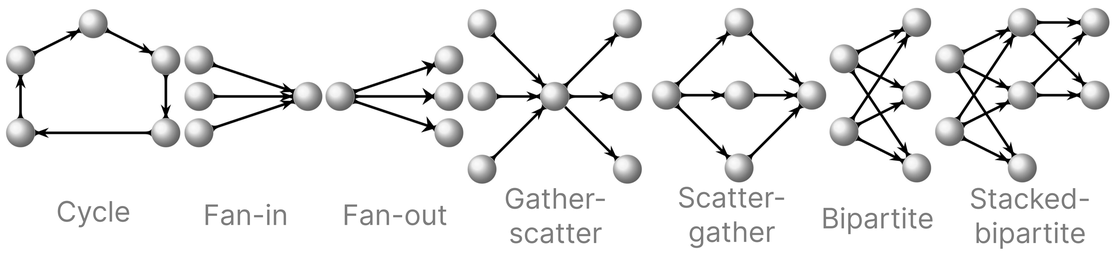
We consider financial transactions in the form of a directed graph, and propose a Graph Neural Network (GNN) model for the identification of money laundering activity. Our method generates a set of structurally aware and statistically significant features for each graph node (bank account), and utilizes them as input to the GNN.

Directed financial graphs

Financial network modeled as multigraph $G(V, E, \mathbf{W})$

- V nodes representing bank accounts
- E directed edges capturing the transactions between them
- Nodes: assigned account balances
- Edges: assigned a weight w_{ij}, with the transaction amount from node i to node j
- The generated feature set is computed from these transaction amounts, the account balances, and the information encoded in the adjacency matrix $\mathbf{W}$.

Embedded money laundering topologies



Feature generation

Structural Features

Degree-in, **-out**, and **-total** in both multi-graph (G) and simple-graph (G_s).

Degree frequency: Node's total degree in multi-graph G divided by nodes total degree in simple-graph G_s.

Standardized node degree: Z-score of node degrees in G_s.

Statistical Features

Based on a node's changes in account balance and the connected transaction amounts as edge weights w_{ij}.

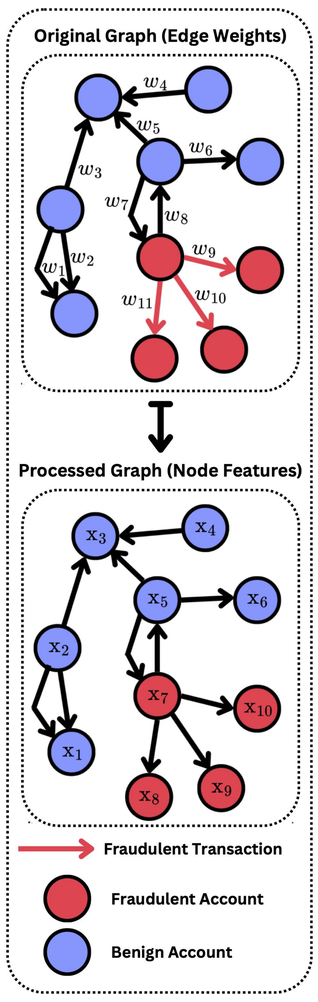
Geometric average of weights (GAW) of each node. Computed from the transaction edge weights w_{ij} and total degree d_i.

$$GAW_i = \left(\prod_{j \in N(i)} w_{ij} \right)^{-d_i}$$

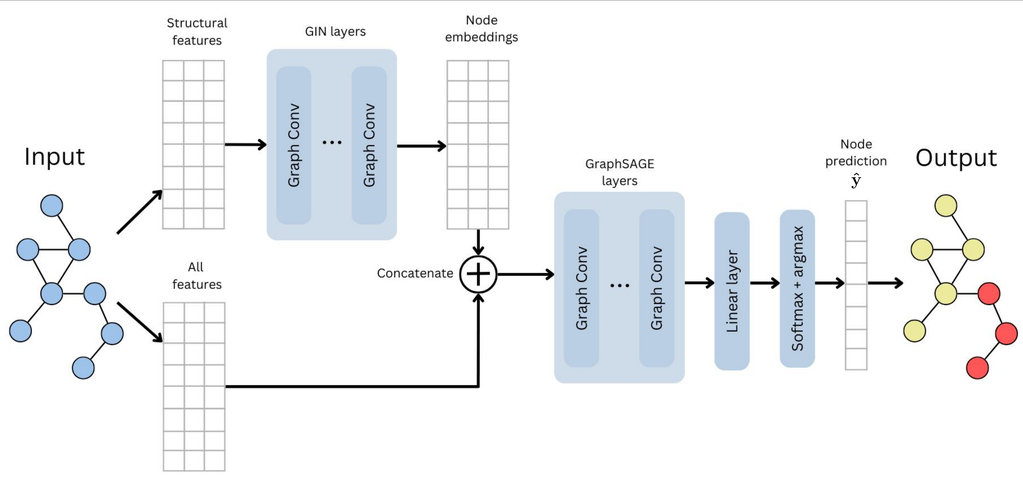
Min-, **max-**, **mean-**, **std.-**, **total-transaction amounts in** of each account.

Min-, **max-**, **mean-**, **std.-**, **total-transaction amounts out** of each account.

Starting-, **ending-**, **min-**, **max-balance** of each account.



Framework and model



We propose the **GAMLNet** (Graph Anti-Money Laundering Network) architecture, leveraging the strengths of two popular GNN variants: **Graph Isomorphism Networks (GIN)** [Xu et al., 2019] and **GraphSAGE** [Hamilton et al., 2017]. GIN performs exceptionally well at learning isomorphic graph substructures and GraphSAGE performs exceptionally well in environments with rich statistical node feature information.

We further compare our **GAMLNet** model to **GIN**, **GraphSAGE**, and two decision tree ensemble methods: **Random Forest** [Breiman, 2001] and **XGBoost** [Chen et Guestrin, 2016]. All methods receive the full feature matrix X. GNN models also provided the set of edges E, containing the connectivity of the graph.

Algorithm 1 GAMLNet forward pass algorithm

Input: Graph $\mathcal{G}(V, E, \mathbf{W})$, feature matrix $\mathbf{X} \in \mathbb{R}^{|V| \times l}$, degree structural features s_{idx}, number of layers K_1, K_2

Output: $\hat{\mathbf{y}} \in \mathbb{R}^{|V| \times 1}$ predicted node classification

$\overline{\mathbf{h}}^{(0)} = \mathbf{X}[:, s_{\text{idx}}]$
for $k = 1$ **to** K_1 **do**
  for $i \in V$ **do**
    $\overline{\mathbf{h}}_i^{(k)} = \tanh\left(\text{MLP}^{(k)}((1+\epsilon^{(k)})\overline{\mathbf{h}}_i^{(k-1)} + \sum_{j \in N(i)} \overline{\mathbf{h}}_j^{(k-1)}) \right)$
$\mathbf{h}^{(0)} = \text{concat}(\mathbf{X}, \phi_1(\mathbf{h}^{(K_1)}))$
for $k = 1$ **to** K_2 **do**
  for $i \in V$ **do**
    $\mathbf{h}_i^{(k)} = \text{ReLU}\left(\mathbf{\Theta}_1^{(k)} \mathbf{h}_i^{(k-1)} + \mathbf{\Theta}_2^{(k)} \cdot \underset{j \in N(i)}{\text{mean}} \mathbf{h}_j^{(k-1)} \right)$
$\hat{\mathbf{y}} = \text{argmax}(\phi_2(\mathbf{h}^{(K_2)}))$
return $\hat{\mathbf{y}}$

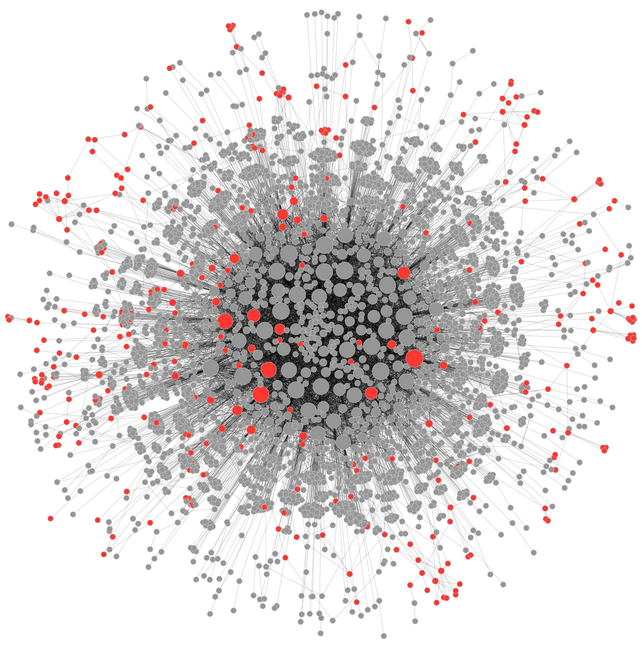
- GAMLNet predicts whether a node is fraudulent or benign, based on the graph G and node feature matrix X.
- Only structural information is passed through the K_1 graph convolution GIN layers.
- The output node embedding with detected subgraph isomorphisms is concatenated with X to form a new node representation $\mathbf{h}^{(0)}$.
- h is passed through K_2 graph convolution GraphSAGE layers.
- $\text{MLP}^{(k)}$, $\epsilon^{(k)}$, Θ_1, Θ_2 are the learned parameters matrices
- The final latent node embeddings are passed through a linear layer to retrieve the classification.
- The model is trained using Adam [Kingma et Ba, 2015] optimizer and weighted binary cross entropy loss.

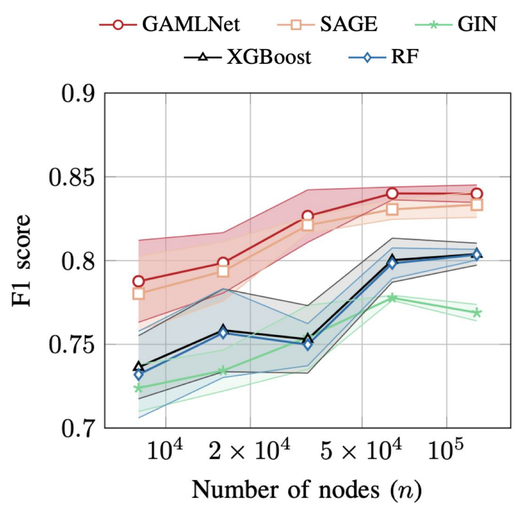
Numerical results

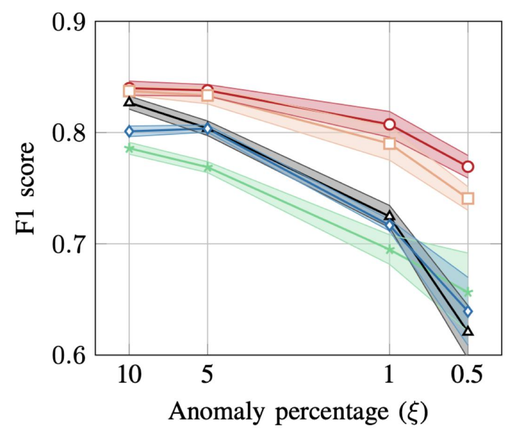
Datasets

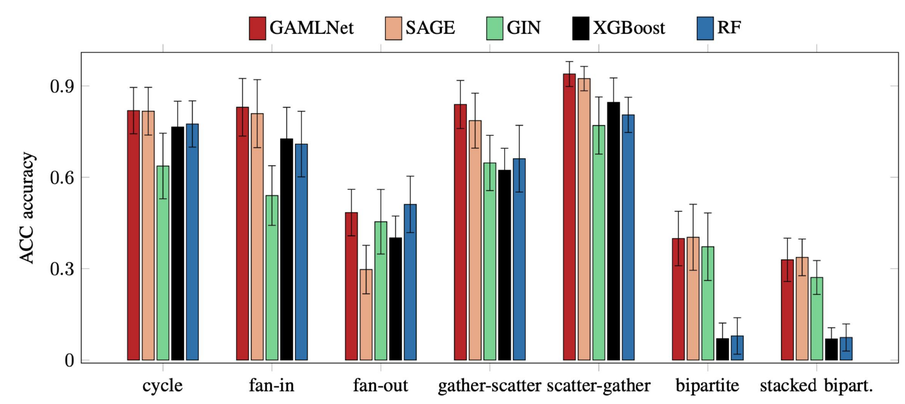
Dataset	% anomalous	$\lvert V \rvert$	$\lvert E \rvert$
1	4.38%	8,032	148,335
2	5.16%	15,995	358,146
3	4.67%	32,013	757,043
4	4.91%	64,026	1,510,182
5	4.96%	128,002	3,751,319
6	9.94%	128,002	3,768,588
7	0.99%	128,002	3,739,042
8	0.52%	128,002	3,737,308

AMLSim: Anti-Money Laundering Datasets [Suzumura et Kanezashi, 2021]









- We test GAMLNet performance against other models across datasets of increasing size and decreasing anomaly percentage.
- Model performance for node classification measured as F1 score (taken from precision and recall) and accuracy metrics.
- GAMLNet performs best across the datasets for both increasing size and decreasing anomaly percentage.
- GAMLNet achieves highest performance in detecting money laundering topologies across most cases.

Code and datasets available at https://github.com/schmidtjulien/GAMLnet.