Using Machine Learning for Anti Money Laundering

Gregor Kržmanc [1], Filip Koprivec [1,2], Maja Škrjanc [1]

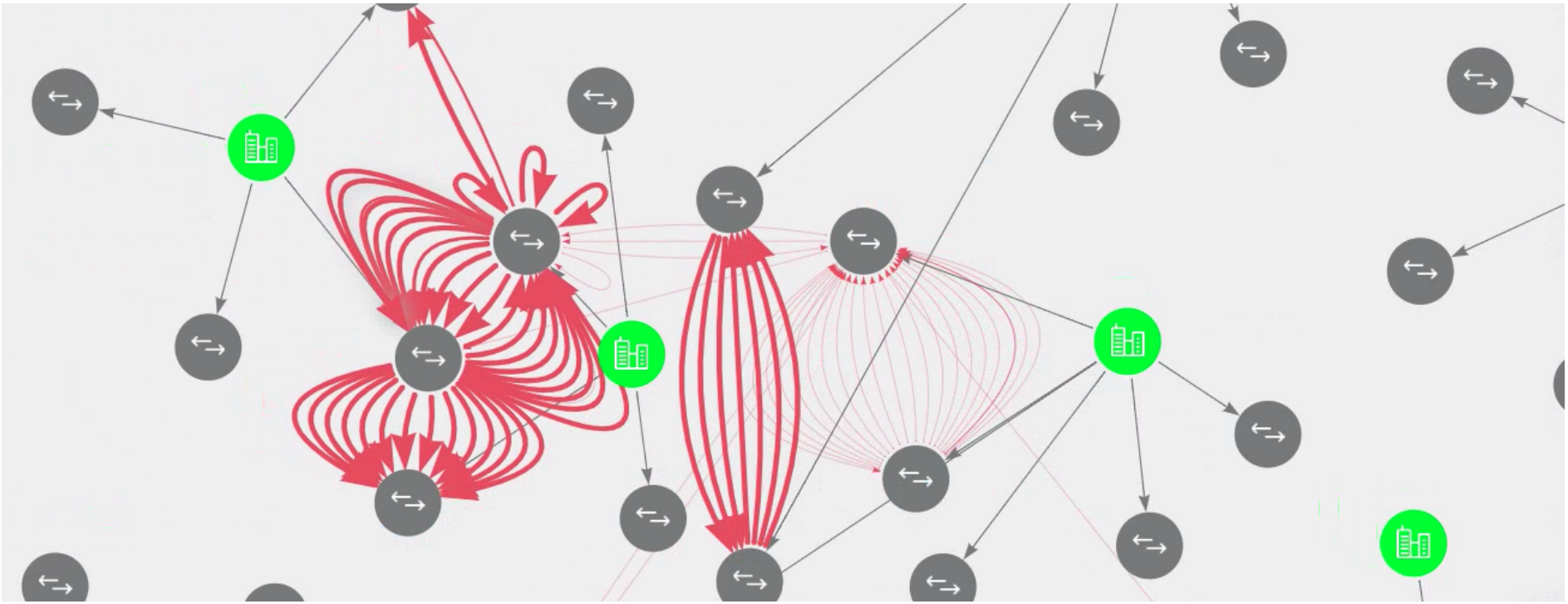
[1]Department of Artificial Intelligence, Jožef Stefan Institute

[2]IMFM

SiKDD 2022, Ljubljana, Slovenia

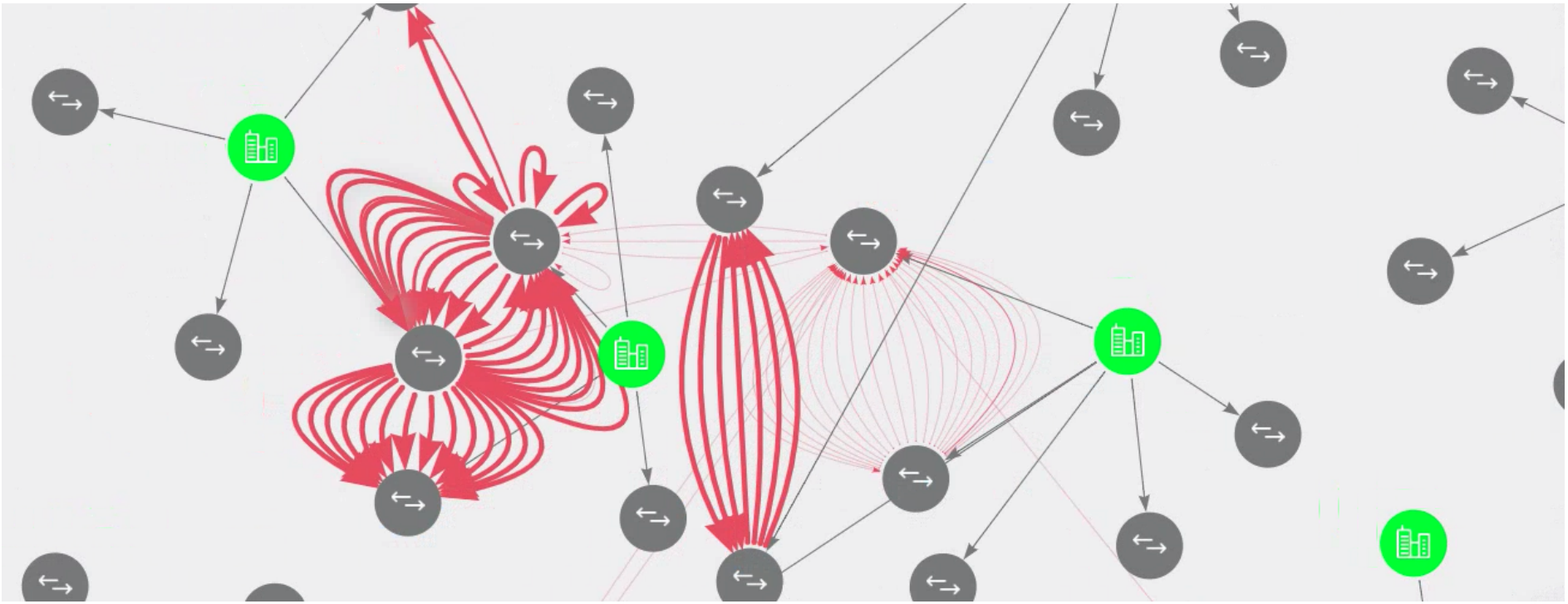
Financial networks

- Nodes: **banks / bank accounts**, edges: **payment transactions**

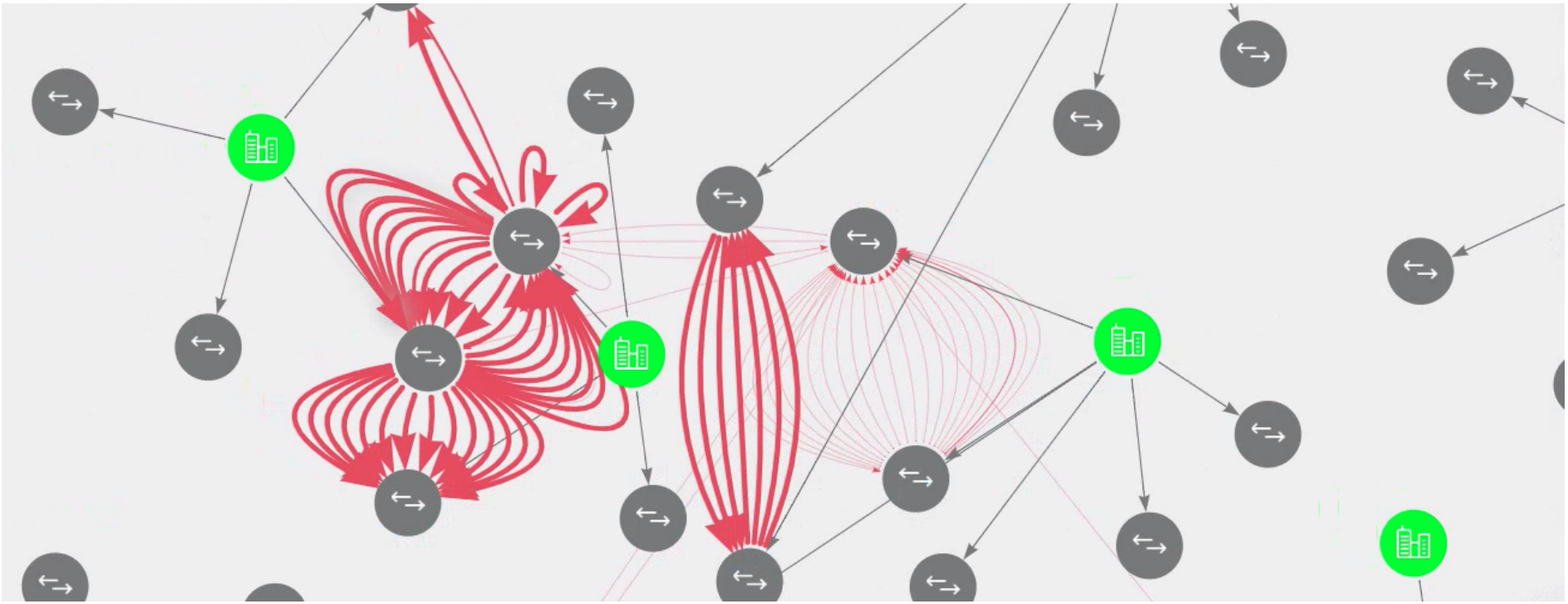


Financial networks

- Nodes: **banks / bank accounts**, edges: **payment transactions**
- **Data quality** issues

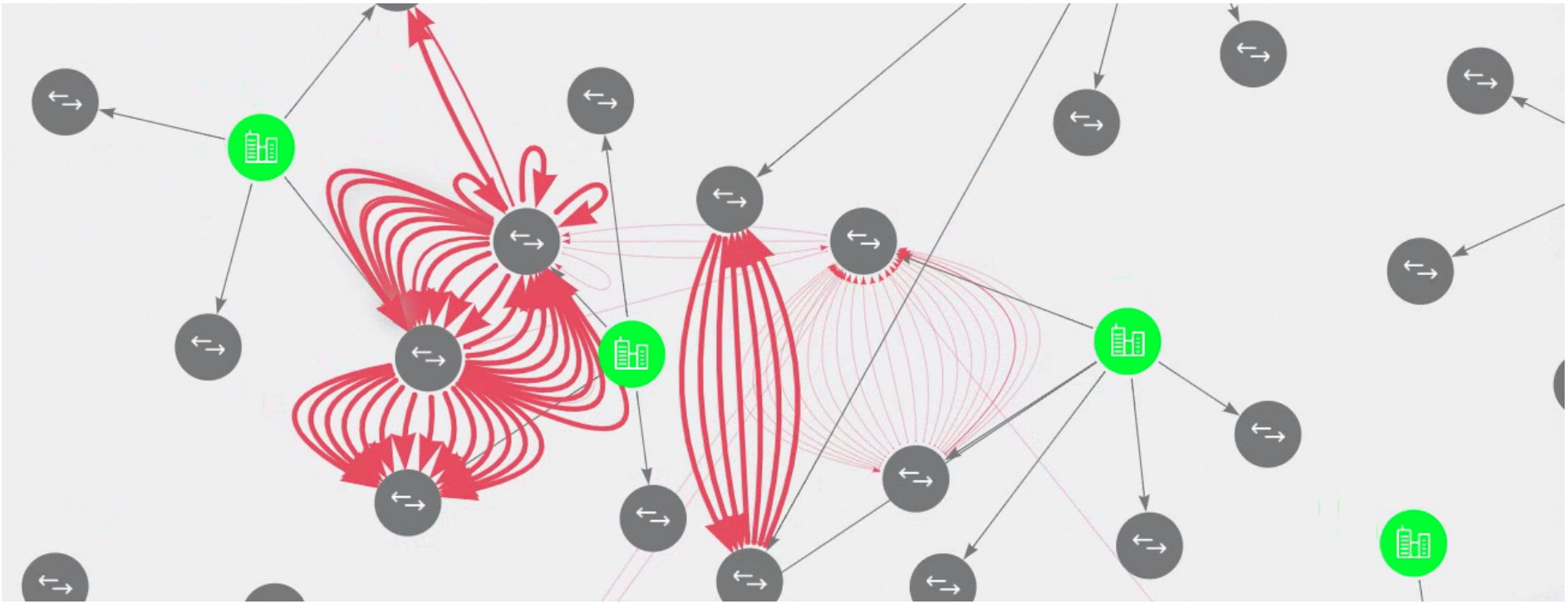


Detecting Money Laundering in Financial Networks



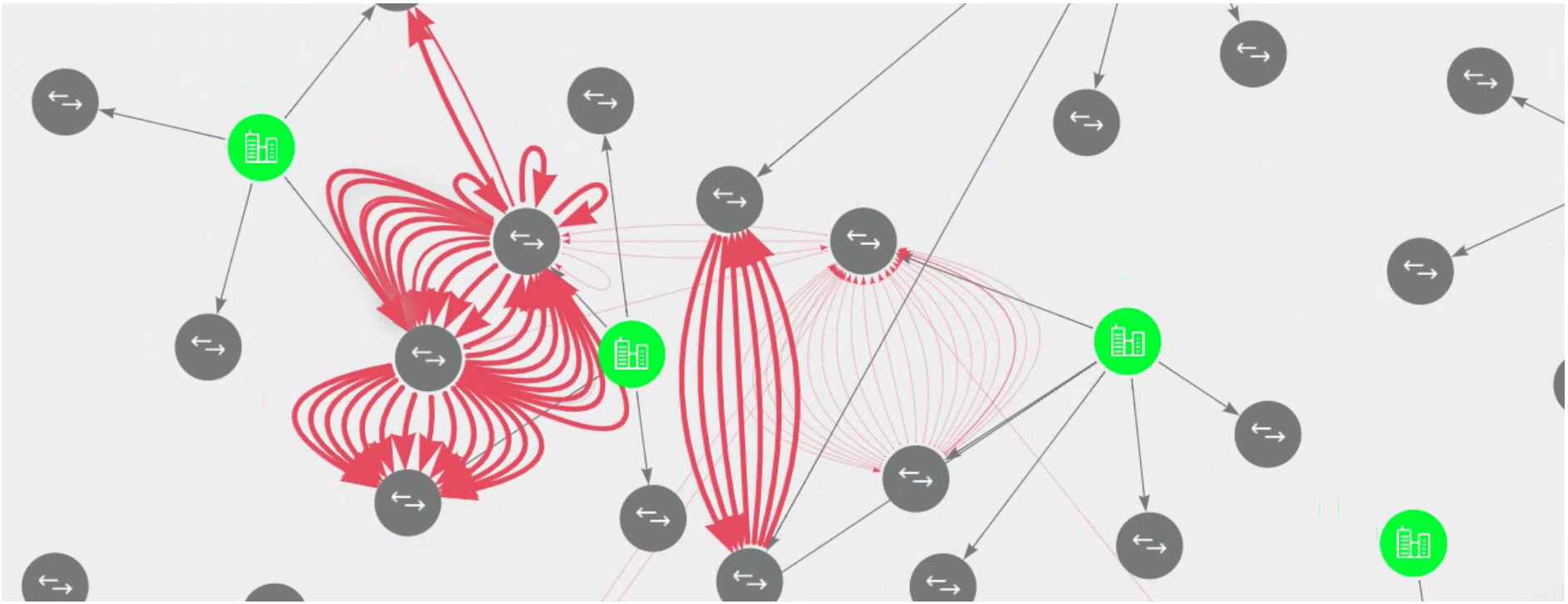
Detecting Money Laundering in Financial Networks

- **Rule-based queries**: based on historical experience



Detecting Money Laundering in Financial Networks

- **Rule-based queries**: based on historical experience
- Utilize structural information to **detect novel anomalous patterns**



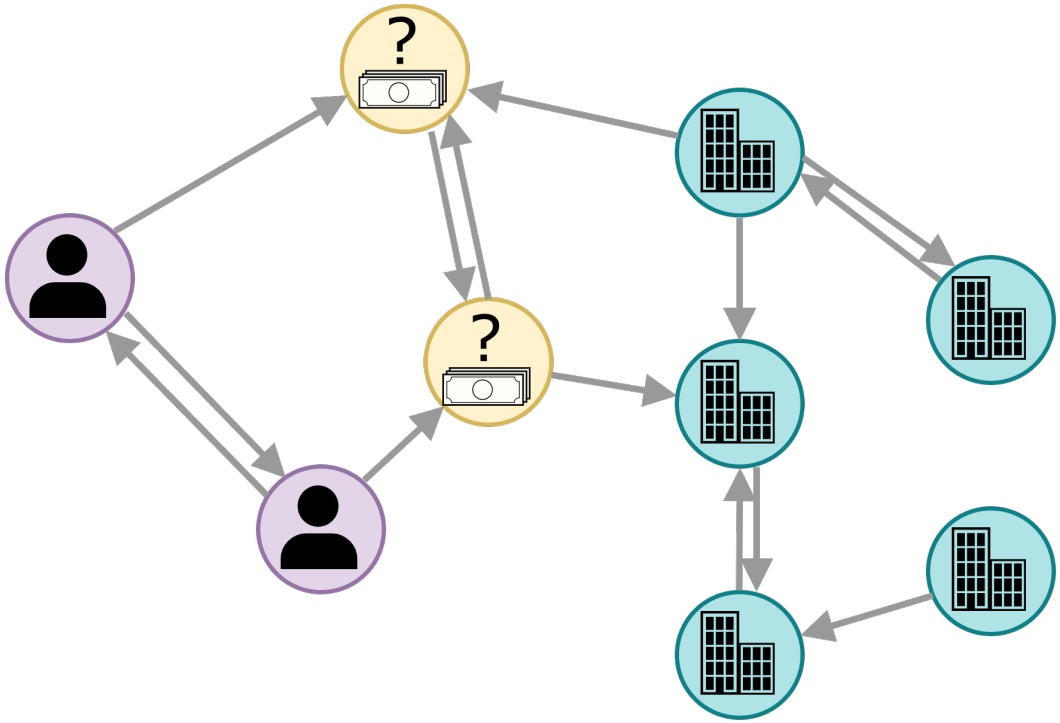
Overview of this talk

- Dataset
- Our network anomaly detection method
- Future work

Dataset

- Payment transactions captured by the payment system *Target2-Slovenija*
- Confidential data - pseudonymized, unpublished dataset

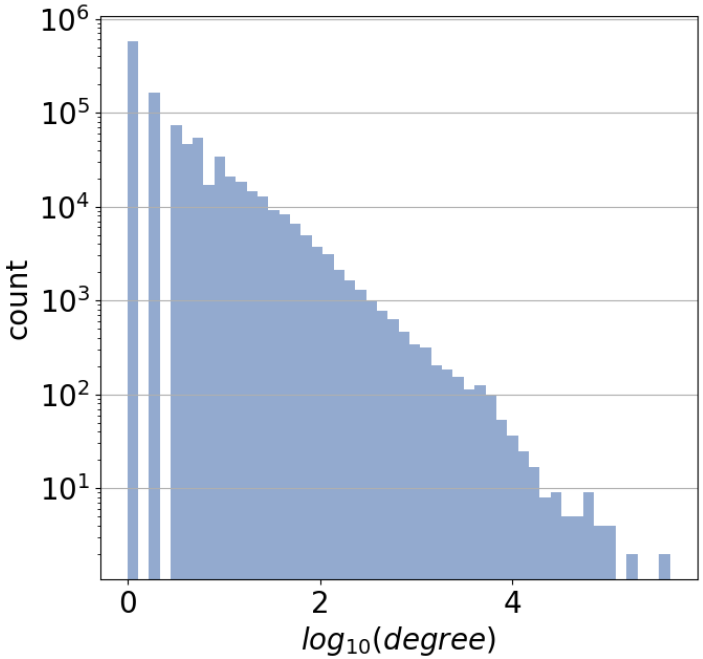
Dataset

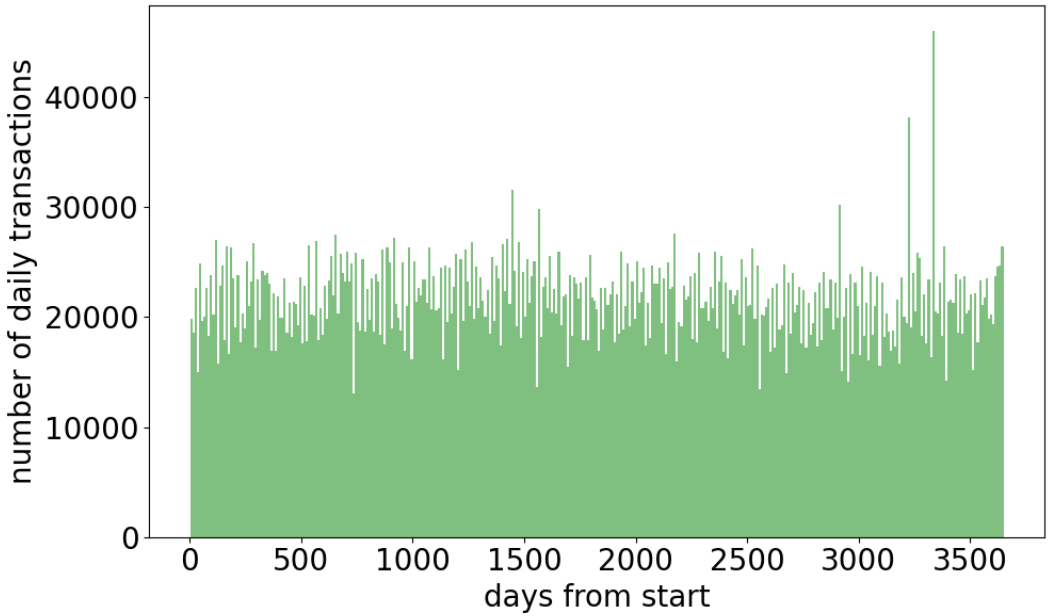


- Heterogeneous network
- Node types:
  - company
  - physical person
  - other/unknown
- Edge types: all possible combinations of node types

Dataset

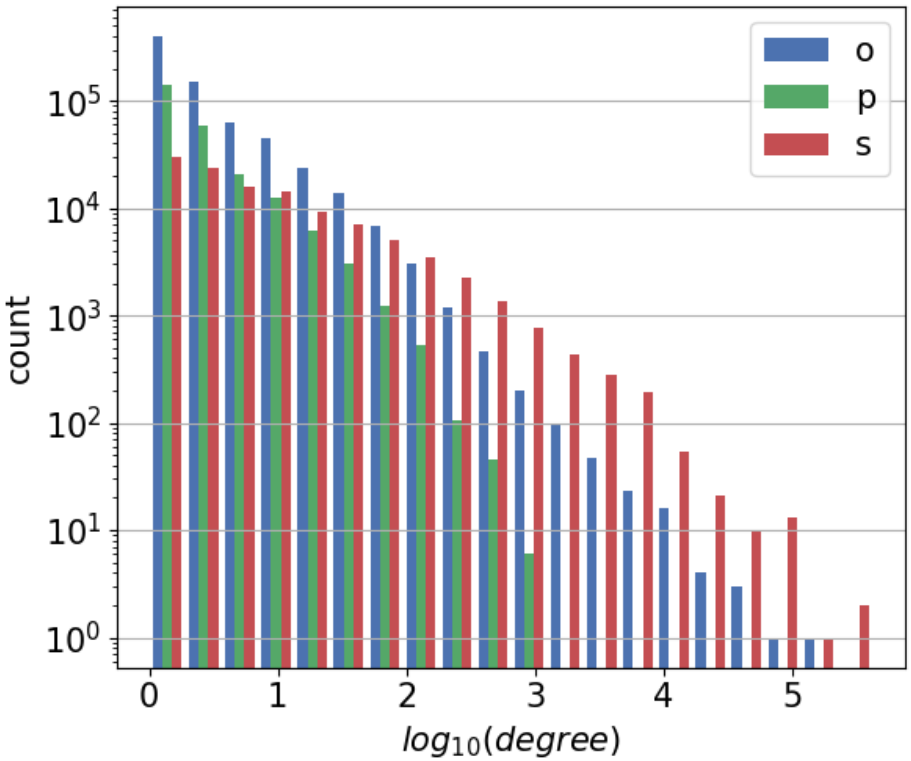
$\sim$ 8 mil. transactions (edges), $\sim$ 1 mil. nodes across $\sim$ 10 years



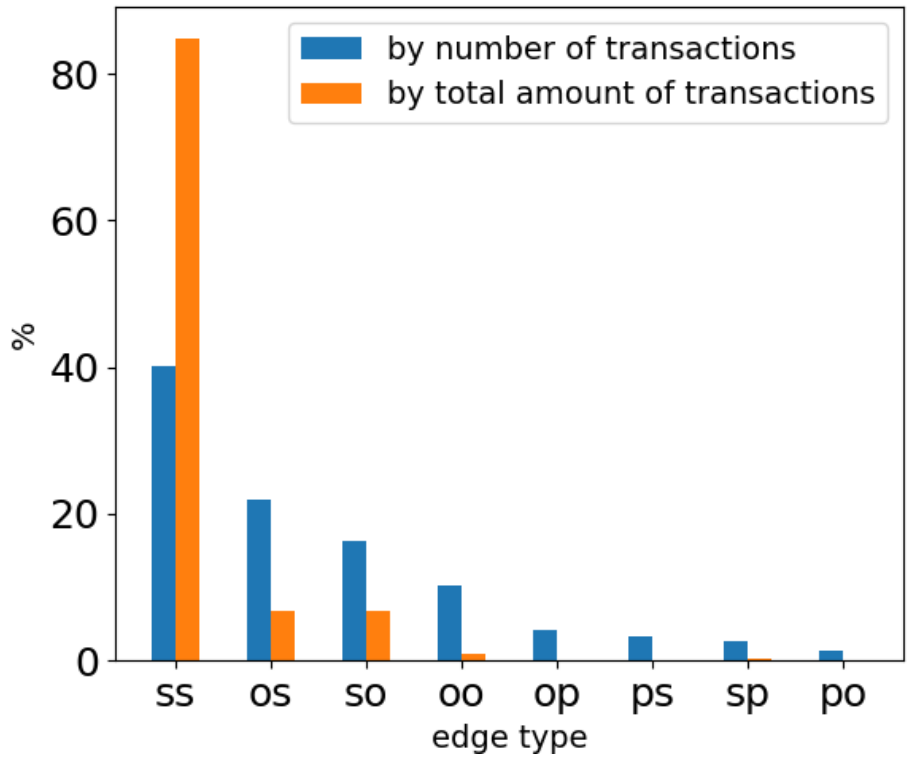


Degree distribution by node type

node type	avg. deg.	median deg.	99% deg.
s (company)	85	4	1095
o (other)	7	1	73
p (person)	4	1	41



Most transactions are company-to-company



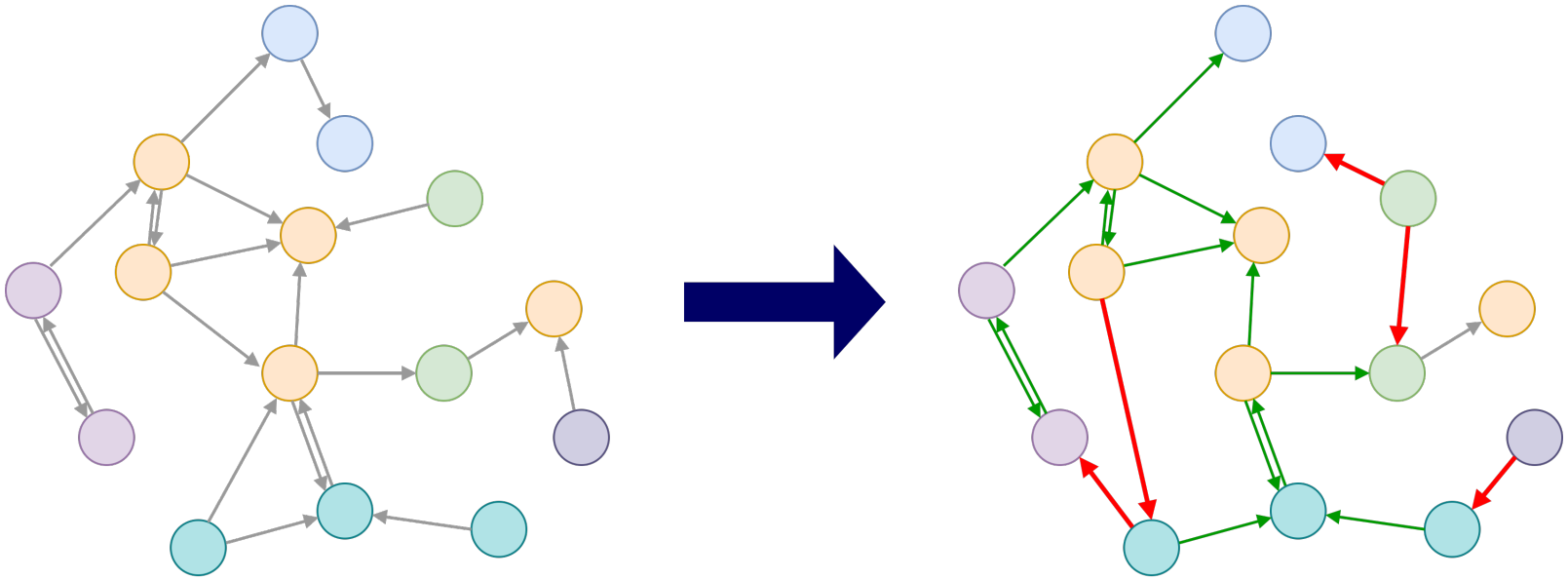
*Edge types are all possible combinations (*source node type*, *destination node type*)

Network anomaly detection

- Learn what links are typical and what are not (link prediction f)
  f () = 1
  f () = 0
- Anomaly score = 1 - f

Network anomaly detection

- Learn what links are typical and what are not (link prediction f)
  f (● → ●) = 1
  f (● ●) = 0
- Anomaly score = 1 - f



Structural feature generation

Manually encode node role

- PageRank centrality
- Degree

Compute these metrics on

- The entire network
- Induced subgraph of the given node's own type

Quantify closeness between two nodes

- Jaccard Coefficient $J(A, B) = \frac{|A \cap B|}{|A \cup B|}$
- Adamic-Adar Index

$$A(x, y) = \sum_{u \in N(x) \cap N(y)} \frac{1}{\log |N(u)|}$$

Results → **Link prediction**

- AUROC for classification on ∼10 different year-long time windows
- Differences between edge types!
- MLP or Gradient Boosting perform best

edge type	DecTree	GradBoost	LogReg	MLP
ss	0.87 ± 0.01	0.90 ± 0.01	0.79 ± 0.01	**0.92 ± 0.01**
oo	0.80 ± 0.01	**0.80 ± 0.02**	0.51 ± 0.01	0.74 ± 0.01
so	0.82 ± 0.01	**0.83 ± 0.01**	0.65 ± 0.01	0.82 ± 0.01
os	0.75 ± 0.01	**0.76 ± 0.01**	0.58 ± 0.02	0.73 ± 0.01
sp	0.81 ± 0.02	**0.85 ± 0.02**	0.55 ± 0.02	0.83 ± 0.02
ps	0.70 ± 0.02	**0.74 ± 0.02**	0.54 ± 0.02	0.69 ± 0.01
po	0.72 ± 0.02	**0.78 ± 0.02**	0.54 ± 0.02	0.67 ± 0.01
op	0.85 ± 0.01	**0.89 ± 0.01**	0.51 ± 0.03	0.87 ± 0.01
all	0.81 ± 0.01	**0.84 ± 0.01**	0.66 ± 0.02	0.82 ± 0.01

Results → **Link prediction**

- GNN$^+$: add reversed edges into the computational graph of GNN

edge type	best non-GNN	no struct. feat.	GNN	GNN$^+$
ss	0.92 ± 0.01	0.89 ± 0.01	0.92 ± 0.02	**0.94 ± 0.01**
oo	**0.80 ± 0.02**	0.57 ± 0.01	0.79 ± 0.02	0.53 ± 0.04
so	0.83 ± 0.01	0.75 ± 0.01	**0.88 ± 0.02**	0.74 ± 0.04
os	0.76 ± 0.01	0.64 ± 0.01	0.81 ± 0.01	**0.83 ± 0.02**
sp	**0.85 ± 0.02**	0.69 ± 0.03	0.78 ± 0.05	0.73 ± 0.02
ps	0.74 ± 0.02	0.67 ± 0.01	**0.87 ± 0.02**	0.75 ± 0.04
po	0.78 ± 0.02	0.66 ± 0.01	**0.84 ± 0.04**	0.54 ± 0.08
op	**0.89 ± 0.01**	0.53 ± 0.01	0.78 ± 0.05	0.50 ± 0.05
all	0.84 ± 0.01	0.72 ± 0.01	0.86 ± 0.02	**0.89 ± 0.01**

Results → **Anomaly detection**

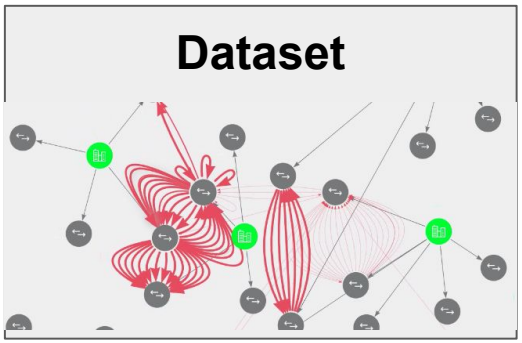
$F_1^{-1} = \frac{\text{precision}^{-1}+\text{recall}^{-1}}{2}$

- non-GNN methods work better
- Perhaps too easy training objective for GNNs? - not able to capture anomalies

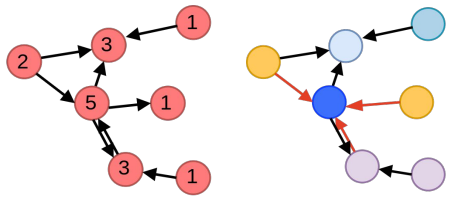
edge type	best non-GNN	no struct feat.	GNN	GNN$^+$
ss	**0.19 ± 0.02**	0.16 ± 0.02	0.01 ± 0.0	0.01 ± 0.00
oo	**0.11 ± 0.02**	0.02 ± 0.01	0.05 ± 0.02	0.03 ± 0.02
so	**0.11 ± 0.02**	0.06 ± 0.01	0.01 ± 0.01	0.01 ± 0.01
os	**0.14 ± 0.02**	0.06 ± 0.01	0.01 ± 0.00	0.01 ± 0.01
sp	**0.08 ± 0.04**	0.02 ± 0.02	0.02 ± 0.01	0.02 ± 0.02
ps	**0.05 ± 0.02**	**0.05 ± 0.02**	0.01 ± 0.01	0.01 ± 0.01
po	**0.07 ± 0.04**	0.07 ± 0.05	0.02 ± 0.02	0.01 ± 0.02
op	**0.18 ± 0.04**	0.02 ± 0.01	0.02 ± 0.01	0.03 ± 0.02

$F_1(\text{naive baseline}) \approx 0.02$

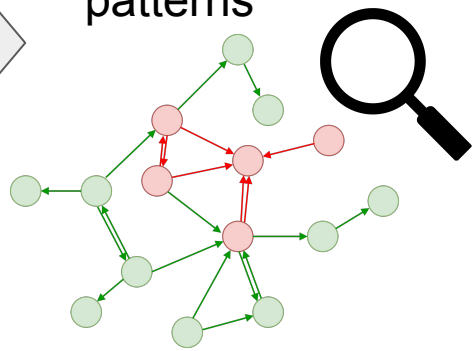
Anti Money Laundering pipeline



- Company features
- Topological graph features / Graph Neural Network (GNN)



Inspect anomalous patterns



Future work

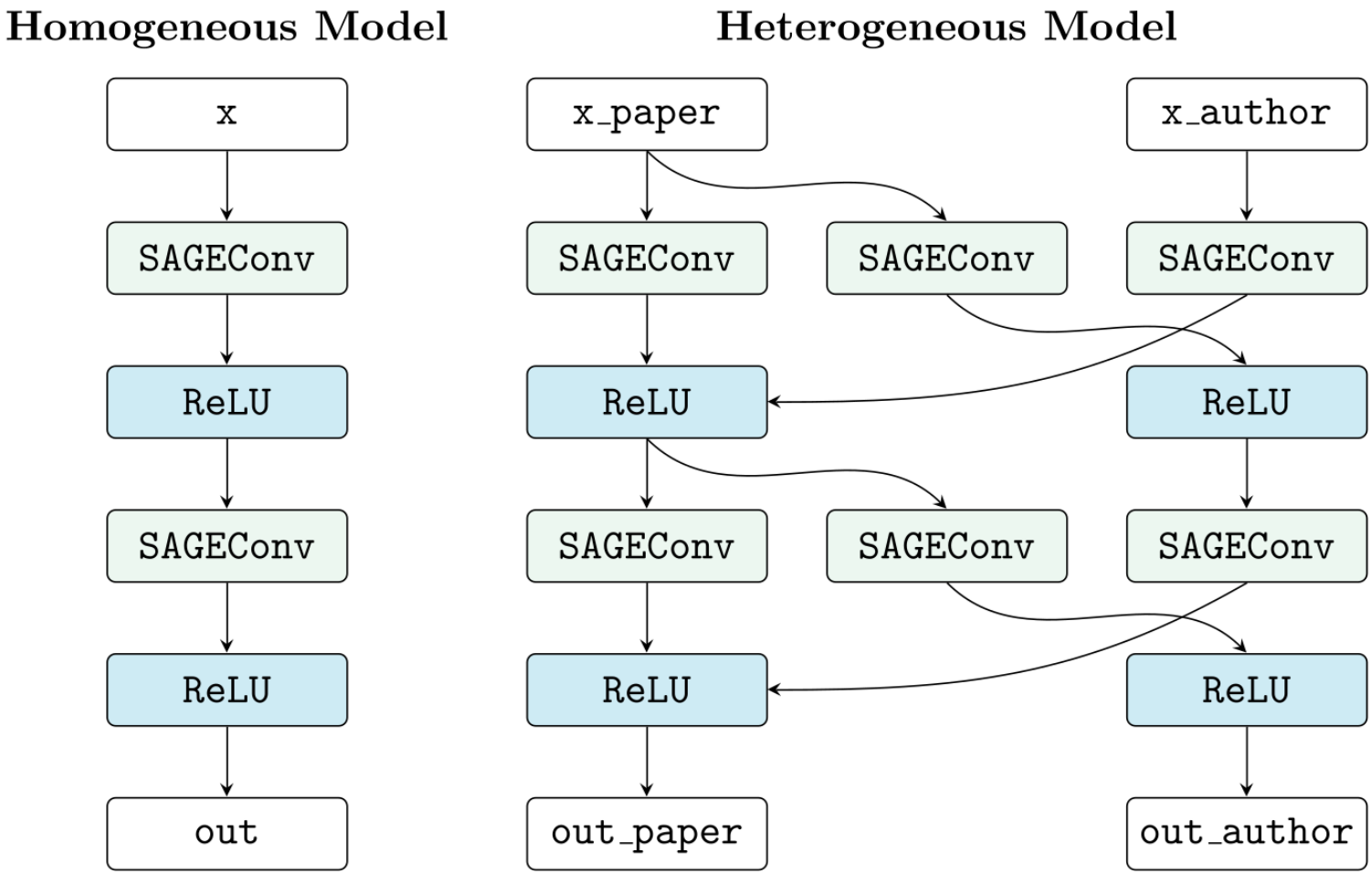
- Include supervised learning
- Supervised learning strategies
  - Active learning
  - Synthetic training sample generation
  - Self-supervised model pretraining

Conclusion

- Payment transaction network as a heterogeneous graph
- Network anomaly detection approach
- Evaluate models using model ability to detect initial dataset (link) corruption
- Goal: Detect data quality issues and potential money laundering cases

Evaluation for anomaly detection

- Corrupt the network by randomly rewiring 1% of edges
- Anomaly score of edge = Probability(edge is classified as non-existent)
- Adjust threshold to match fixed anomaly rate (2%)
- Report precision and recall on detecting the corrupted edges



https://pytorch-geometric.readthedocs.io/en/latest/notes/heterogeneous.html

Evaluation

- Corrupt the network by randomly rewiring 1% of edges
- Anomaly score of edge = Probability(edge is classified as non-existent)
- Adjust threshold to match fixed anomaly rate (2%)
- Report precision and recall on detecting the corrupted edges

(non-GNN methods)

edge type	DecTree	GradBoost	LogReg	MLP
ss	0.12 ± 0.01	0.13 ± 0.02	0.04 ± 0.01	**0.19 ± 0.02**
oo	0.07 ± 0.01	**0.11 ± 0.02**	0.01 ± 0.01	0.10 ± 0.02
so	0.08 ± 0.01	0.10 ± 0.02	0.04 ± 0.01	**0.11 ± 0.02**
os	0.06 ± 0.01	0.12 ± 0.02	0.04 ± 0.01	**0.14 ± 0.02**
sp	0.06 ± 0.01	0.07 ± 0.04	0.02 ± 0.02	**0.08 ± 0.04**
ps	0.04 ± 0.01	**0.05 ± 0.02**	0.01 ± 0.01	**0.05 ± 0.02**
po	0.04 ± 0.01	**0.07 ± 0.04**	0.02 ± 0.03	0.04 ± 0.03
op	0.09 ± 0.01	0.14 ± 0.04	0.01 ± 0.01	**0.18 ± 0.04**