Example Workflow: Fraud Detection and Investigation

Why Fraud Detection?

- Most popular use case in financial industry
- Easier to understand without prior knowledge of financial domain
  - *e.g. Everyone has a Credit Card!*
- GDS workflow with a sequence of Graph Algorithms to answer a specific set of questions

Fraud Categories

Fraud occurs

when an individual or group of individuals or a business entity

intentionally

deceives another individual or business entity with

misrepresentation

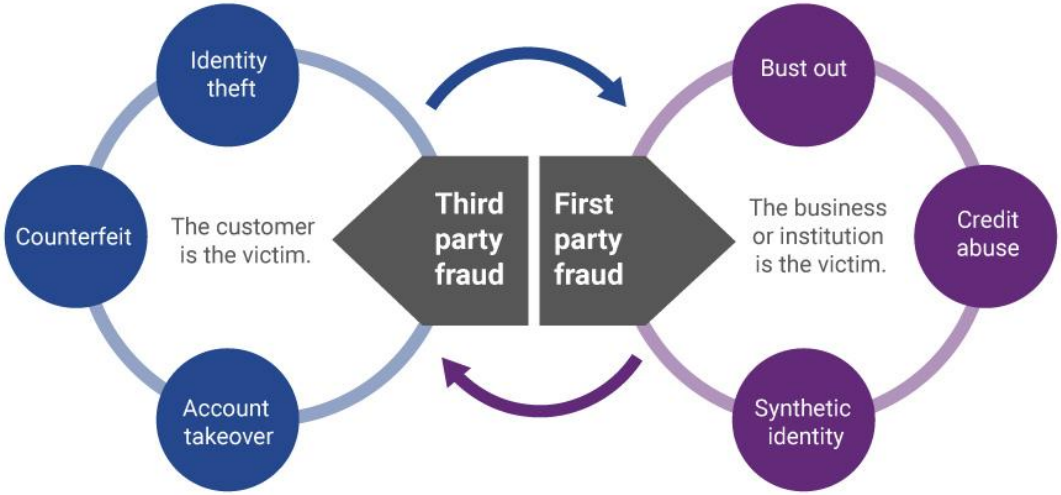
of identity, products, services, or financial transactions and/or

false promises

with no intention of fulfilling them.

Financial Fraud

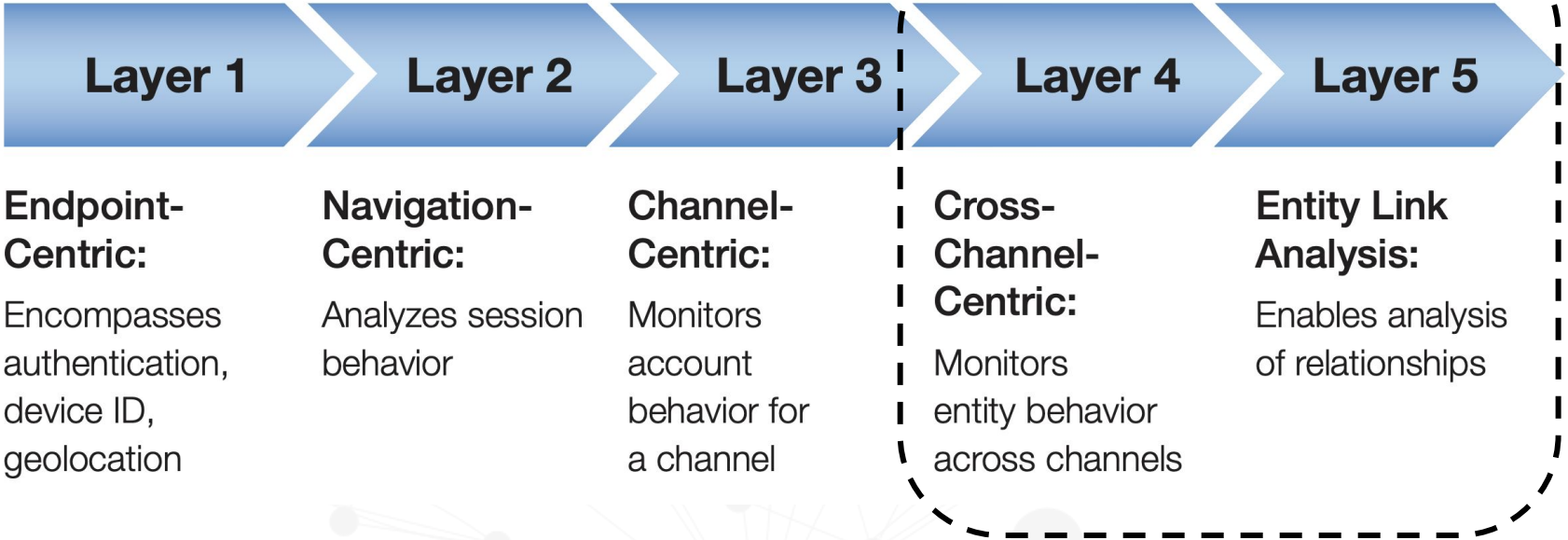
- **First Party Fraud**
  - Fake information
  - Customer and fraudster are the same
- **Second Party Fraud**
  - Money mules
  - Customer and fraudster are both involved
- **Third Party Fraud**
  - Stolen identity
  - Customer is the victim



http://www.experian.com/blogs/insights/2017/09/whos-on-first/

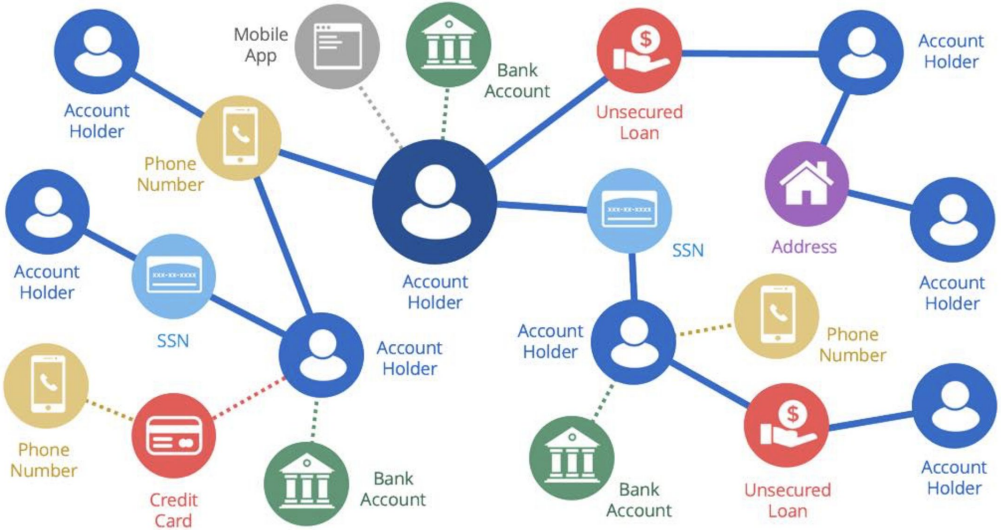
Fraud Detection and Investigation

LAYERED APPROACH



https://www.sas.com/content/dam/SAS/en_us/doc/whitepaper1/layered-approach-fraud-detection-prevention-106072.pdf

Why Neo4j?



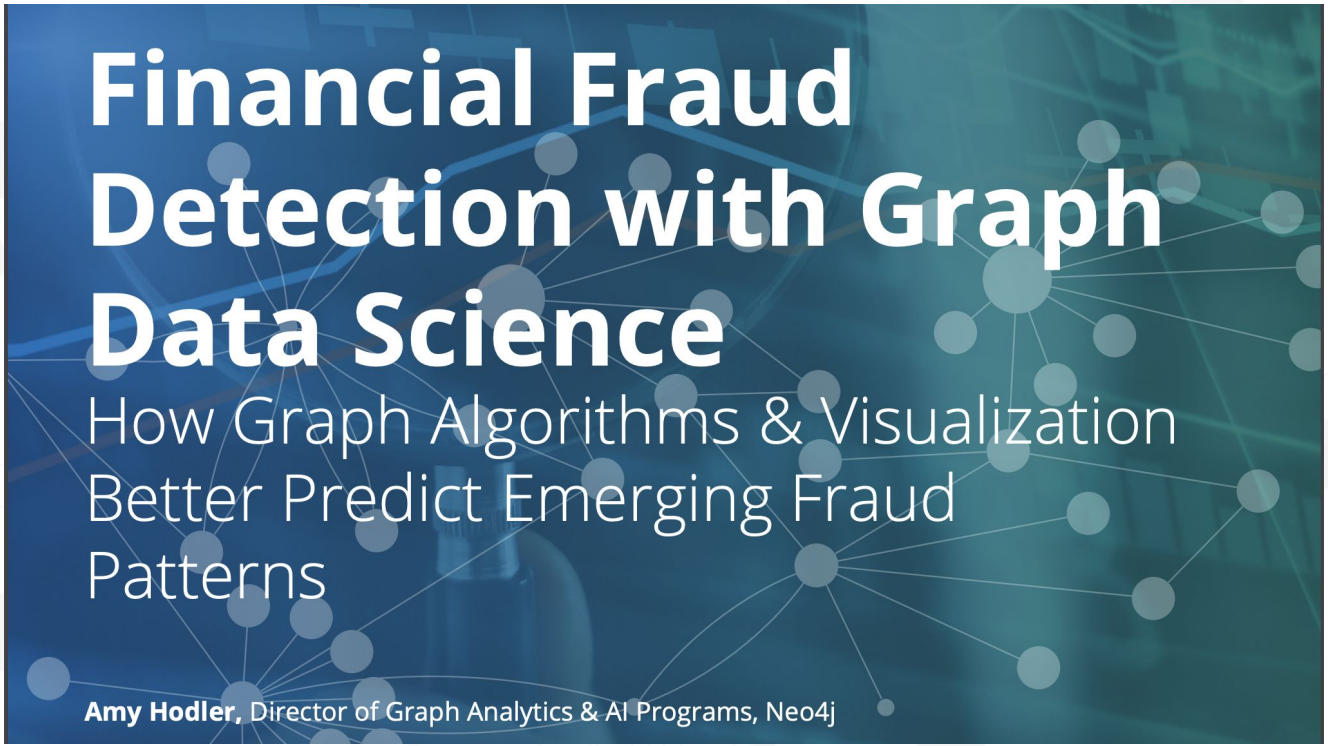
Connected Data

- Ideal for investigating cross-channel fraud

Relationships

- Entity resolution, Entity link analysis, first, second and third party fraud detection

Neo4j Resources



https://neo4j.com/whitepapers/financial-fraud-detection-graph-data-science/

neo4j

Let’s Start..

- What Questions?
- What Data?
- Which Algorithms?

What Questions?

- First Party Fraud Detection
  - Identifying clusters of clients sharing personally identifiable information (PII) like SSN, Phone Number and Address
  - Pairwise similarity calculation on the clients sharing PII
  - Computing a first party fraud score

What Questions?

- Second Party Fraud Detection
  - Identify transactions between first-party fraudsters and other clients
  - Explore the type of transactions between these two groups
  - Compute a Second-party Fraud Score

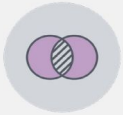
What Algorithms?

Community Detection

Identify disjointed groups that share identifiers.

Identify communities that frequently interact

Louvain Modularity, Weakly Connected Components, Label Propagation, ..



Similarity

Measure account similarity or fraud ring similarity

Jaccard similarity, Cosine similarity,..

Centrality

Measure influence and transaction volumes.

PageRank, Betweenness, ..

Heuristic Link Prediction

Find unobserved relationships and add them to your data.

Common Neighbors, Preferential Attachment, ..

Pathfinding & Search

Filter transactions with extremely short paths between people.

Shortest Path, A*, Random Walk, ...

What Data?

Ref: ***https://www.sisu.io/posts/paysim/***

Paysim

- Synthetic financial data set using an agent-based model and aggregate transactional data from a real mobile money network operator
- Agents perform Transactions exchanging money at each step in time with other Agents
- Agents: Clients (End Users), Merchants (Vendors) and Banks (Financial Institutions)
- Transactions: CashIn, CashOut, Debit, Transfer and Payment
- Steps: one hour of time and agents can perform one or more than one transactions per step

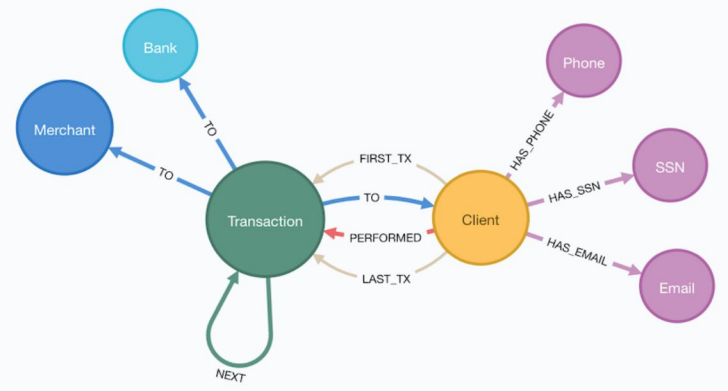
Lopez-Rojas, Edgar Alonso & Elmir, Ahmad & Axelsson, Stefan. (2016)

What are we going to do?

1. Framing the problem
2. Data Exploration
3. First Party Fraud
4. Second Party Fraud

Let’s explore..

Browser Guide Ref: 5



SCHEMA

```
`(:Client) - [:PERFORMED] -> (:Transaction) - [:TO] ->
(:Merchant | :Bank | : Client)

`(:Client) - [:HAS_SSN | :HAS_EMAIL | :HAS_PHONE] ->
(:SSN | :Email |:Phone)

`(:Client) - [:FIRST_TX] -> (:Transaction) - [:NEXT]
-> (:Transaction) - [:NEXT] -
```

Questions?