How to Identify

What's Important, What's Unusual, and What's Next

Using Graph Algorithms

Answering the Big Questions

Smarter Predictive Analytics with Graph

A graph gives data science teams the ability to look at, understand, and analyze the connections *between* each data point. This adds context to the data that is impossible to get from a tabular data model. In a graph, you can see how strong connections are, where groups of connections form, how important each connection is, and how connections influence one another.

Graph data science uses relationships to add context and answer pressing questions.

Finding Signal in the Noise

Effective data science teams know the best way to analyze their data based on the business need, the type of problem, and the data available. Graphs are a valuable tool when decision makers ask **What's important? What's unusual?**

What's next? When relationships provide the context to answer these questions, you have a graphy problem.

> *80% of data and analytics innovations will use graph technologies by 2025.*
>
> *Gartner**

Common Uses for Graph Data Science

The big questions help you understand the power of graph data science, and because these are general, they can be applied across multiple lines of business and industries. Here are some specific examples of how to apply the big questions to your area of the business.

Finance

- Fraud Detection
- Resource Allocation
- Churn Prediction

Operations

- Cybersecurity: Data Loss Prevention
- Data Unification
- Network Monitoring

Marketing

- Funnel Operations
- Customer 360
- Market Segmentation
- Influencer Identification

Supply Chain

- Route Optimization
- Risk Identification
- Inventory Management

People Operations

- Employee Training & Upskilling
- Internal Backfill Optimization

Customer Service

- Next Best Action
- Optimize Knowledge Base

*Top 10 Trends in Data and Analytics, 2021, Gartner, Inc.

What's Important? (Prioritization)

There are numerous examples of decision makers trying to determine project urgency and therefore, prioritization. For example:

Marketing: What is the most important piece of content, the most important webpage, the most important call to action?

Product Teams: Where is the most friction?

Support: Which article is the most important?

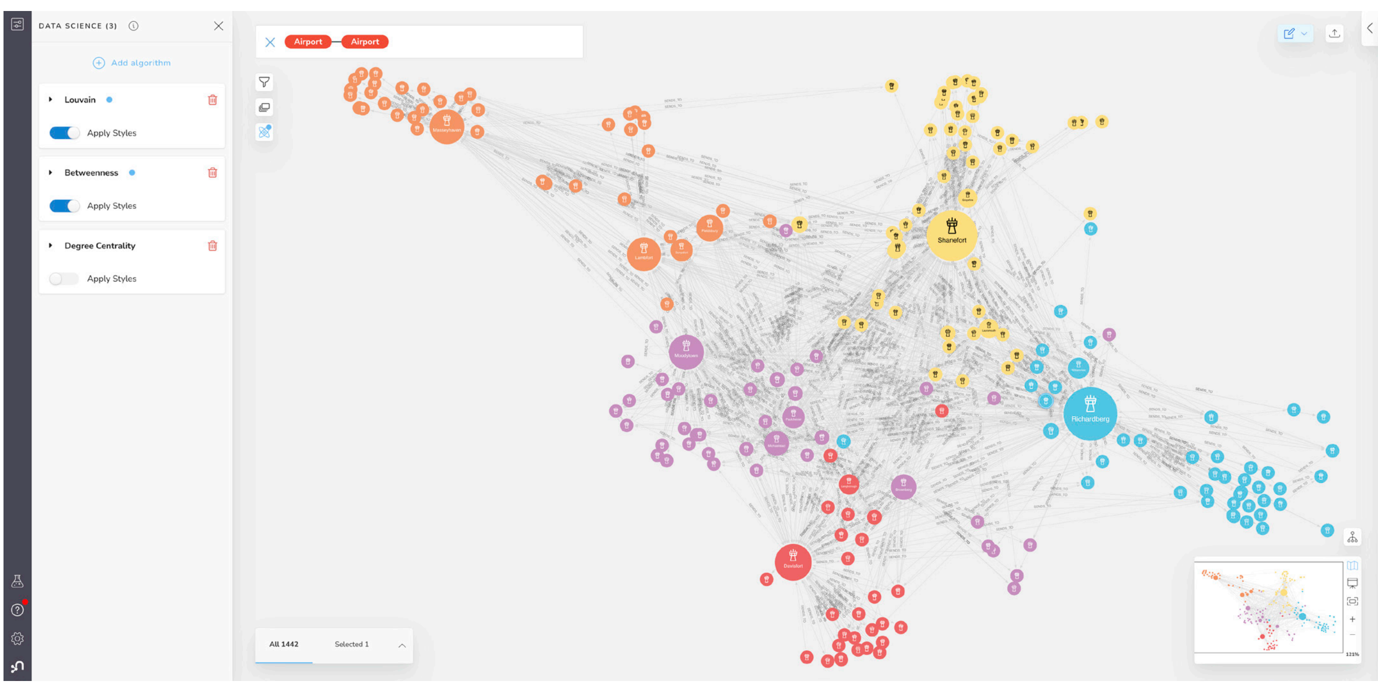
Finance: Which report is most important for leadership teams?

If you're hearing words like *best, top performing, converting*, or *challenging* your decision makers are asking you about importance.

Example: Most important product

Knowing your most important product can help you to position against the competition, improve lifetime value of customers, and strengthen brand loyalty. Identifying your most important product can be a challenge because sales data may not tell the whole story. By combining sales data with customer reviews, a more holistic picture emerges about what products are the most important to your organization.

Graph data science helps you see the most connected and influential products in your catalog by using graph algorithms and connected data techniques to strengthen your model.



Visualize and analyze data in a graph to show which nodes are most well connected and influential.

What's Unusual? (Anomalies)

It is easy to immediately think of fraud detection when we see the question "what's unusual?" But other departments ask their data science counterparts to identify unusual behavior as well.

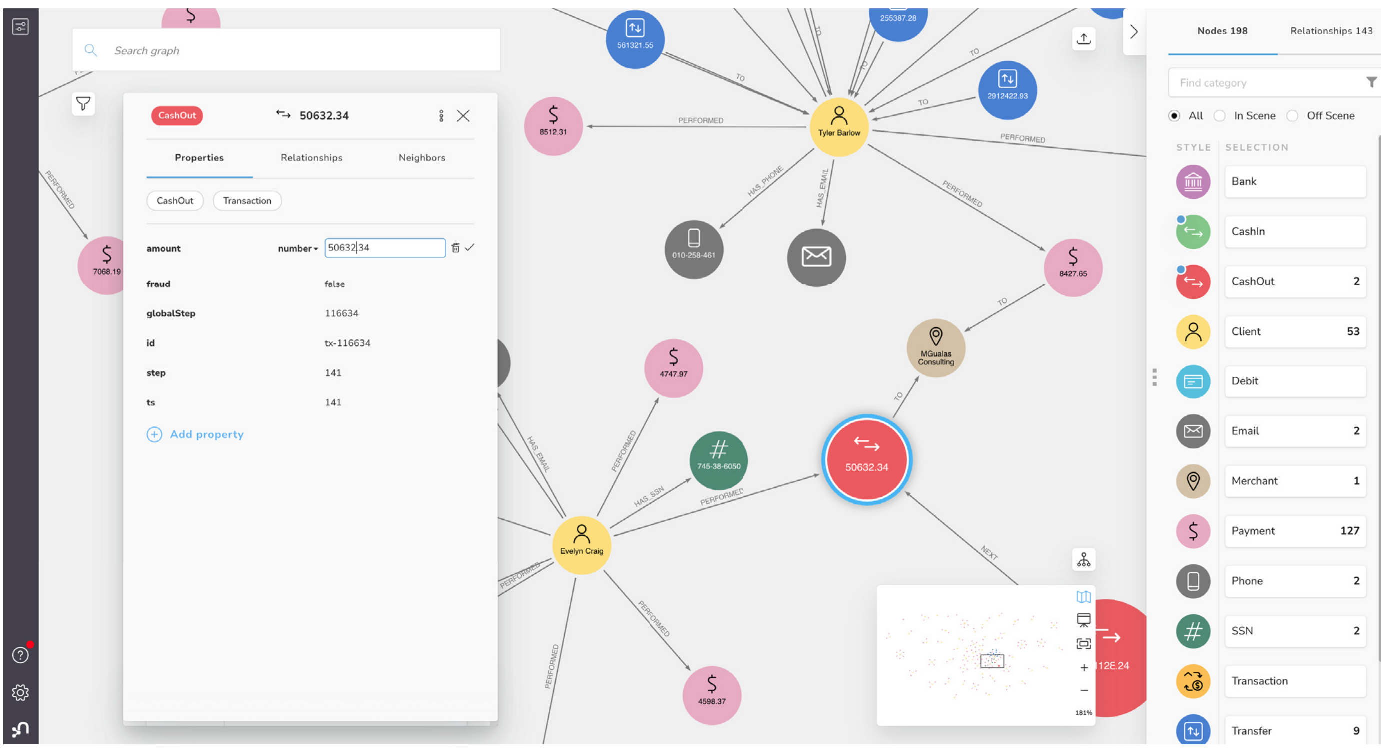
- IT: Where is unusual activity on my network devices?
- Finance: Where is unusual activity in my accounting department?
- SecOps: Where is unusual activity in my data center?
- Compliance: Is there unusual activity in contract language?

Finding anomalous behavior is one of the most beneficial uses of graphs.

The same strategy used to identify threat actors in a cybersecurity context can be used to detect fraud in banking, insurance, and government programs by analyzing the relationships and behaviors in a graph.

Viable fraud detection remains one of today's most challenging data science problems. Fortunately, graph-based approaches explicitly model relationships between entities in the data and this, coupled with a graph data science engine, can empower practitioners to rapidly explore, analyze, resolve, and predict fraud entities and patterns. Such patterns would otherwise remain obfuscated and are challenging to infer in other data models.

Example: Fraud Detection



What's Next? (Predictions)

Looking ahead and predicting the future is something most of us wish we could do with ease. Recommender systems are perhaps the most applicable example across every area of the business.

Predictive insights using graphs can deliver answers to these questions and more.

Marketing
- What email should we send customers next?

Product Teams
- What product should we build next?

Retailers
- What product should we sell next?

Human Resources
- What training should an employee take next?

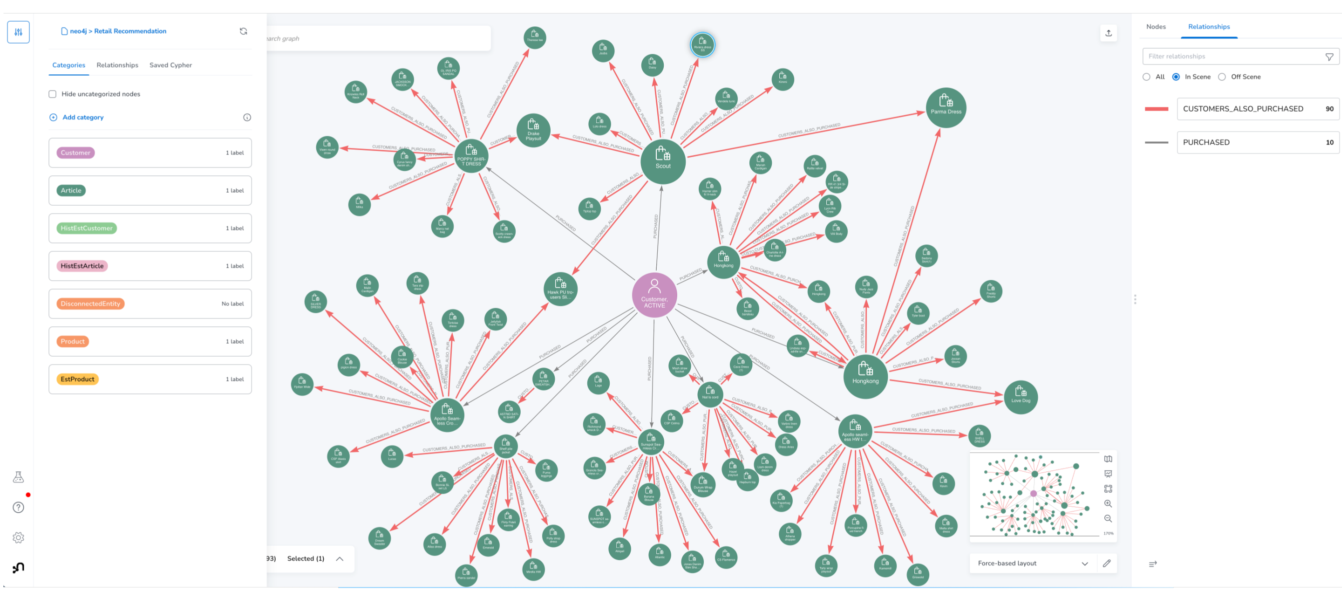
Finance
- How should we price our products next quarter?

Operations
- What is the fastest path from point A to point B?

Hearing questions about how to *improve* or *optimize* will tip you off that your stakeholders are looking for what to do next.

Example: Recommendation Engines

Recommendation engines are well known through Netflix and online shopping experiences. Companies must be concerned not only about the quality of their recommendations (are we suggesting the right thing?), but also about how quickly they can derive relevant recommendations and serve them to their users (are we suggesting the most up-to-date thing?). No one likes seeing an advertisement for a pair of shoes they bought three weeks ago following them around on the web.



Provide rapid personalized product recommendations based on shared customer interactions and product associations.

A graph data science engine can analyze the connections in your data to power recommendation engines that fuel customer satisfaction and business growth, gaining otherwise unattainable insights from the relationships that exist in the data you already have.

The Power of Graph Algorithms

Graph algorithms are a set of instructions that visit nodes of a graph to analyze the relationships in connected data. Below are some of the primary algorithm categories.

Community detection algorithms cluster your graph based on relationships to find communities where members have more significant interactions. Detecting communities helps predict similar behavior, find duplicate entities, or prepare data for other analyses.

These include Louvain, Triangle Count, Strongly Connected Components, Label Propagation, and more.

Centrality algorithms reveal which nodes are important based on graph topology. They identify influential nodes based on their position in the network and are used to infer group dynamics such as credibility, rippling vulnerability, and bridges between groups.

These include Page Rank, Betweenness Centrality, Closeness Centrality, and others.

Similarity algorithms employ pairwise comparisons to score how alike individual nodes are based on their neighbors or properties. This approach is used in applications such as personalized recommendations and developing categorical hierarchies.

These include Node Similarity, K-Nearest Neighbors, and others.

Pathfinding algorithms find the shortest paths between two or more nodes or evaluate the availability and quality of paths. Use them to analyze complex dependencies and evaluate routes for uses such as physical logistics and least-cost call or IP routing.

These include Dijkstra Shortest Path, Yen's Shortest Path, Depth and Breadth First Search, and many others.

Node embedding algorithms transform the topology and features of your graph into fixed-length vectors that represent each node. They capture the complexity and structure of a graph and transform it for use in various ML tasks.

These include Node2Vec, FastRP, and GraphSAGE.

Graph-native ML techniques like **link prediction** and **node classification** fill in the blanks in your data and predict changes in the structure of your graph. They enable use cases such as fraud detection, drug discovery, entity resolution, and more.

These include Node Classification Pipelines, Link Prediction Pipelines, and Node Regression Pipelines.

Conclusion

Data scientists bring value to the business by building and improving analytics and ML models to help stakeholders make better decisions. Graphs help data scientists answer big questions like what's important, what's unusual, and what's next to get better predictive accuracy and improve their models. Specialty graph algorithms help data scientists see the signal in the noise.

Neo4j Graph Data Science

Neo4j Graph Data Science is an analytics and ML engine that uses the relationships in your data to improve predictions and plugs into enterprise data ecosystems so you can get more data science projects into production quickly.

Using a library of over 65 pretuned graph algorithms, data scientists can explore billions of data points in seconds to identify hidden connections and generate compelling visualizations that lead to better stakeholder decision making.

Neo4j Graph Data Science can be deployed on-premises or in the cloud as a self-managed and fully-managed cloud offering. It fits in easily with your existing data stack and pipelines, complementing your investment and offering groundbreaking insights.

Learn More!
Read our Use Case Selection Guide

Connect with one of our Graph Data Science Specialists.

Questions about Neo4j? Contact us around the globe:

info@neo4j.com
neo4j.com/contact-us