GRAPH+AI SUMMIT organized by TigerGraph

Using Graph to Boost AI

Uri Lapidot - Senior Product Manager

Agenda

Intuit - Introduction

The Challenge
Description of the business problem and success criteria

How Intuit Risk Leverage Graph Technology
High level overview of Intuit graph system

Graph Based Features
What are graph based features? And why we need them to fight fraud?

Serving Graph Based Features to our Models
The challenge of integrating graph-based features with our models

Q&A

Intuit - AI Driven Expert Platform

Intuit - AI Driven Expert Platform

The Challenge

Integrate Graph Features with ML Models

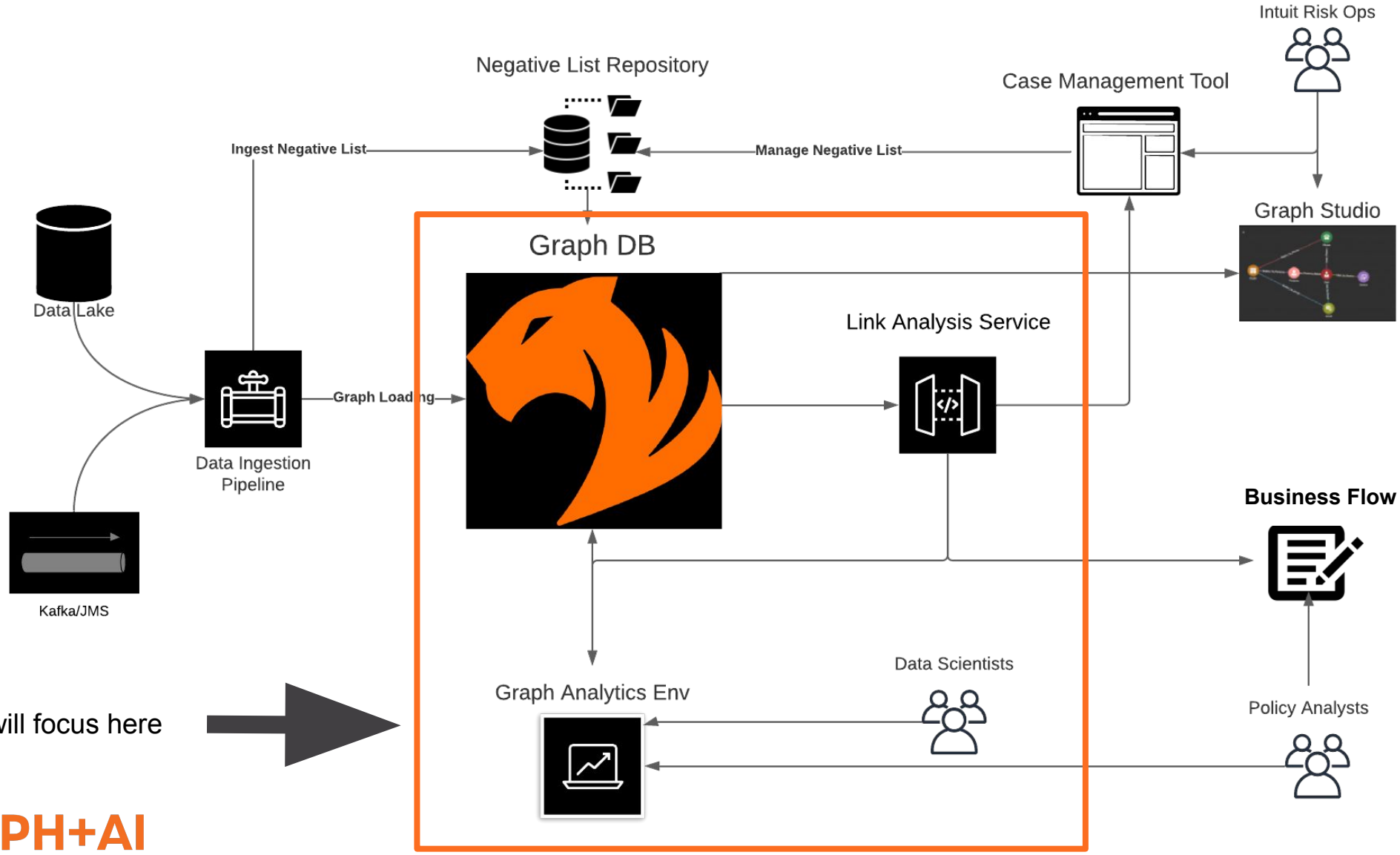
- **The Challenge:** "Use of Graph Insights on linking Fraudulent entities in our ML/AI models" to turbocharge our fraud and risk controls, reduce fraud across multiple checkpoints in our end-to-end ecosystem and improve customer experience.

- **Two Main Challenges:**
  1. Data Scientists needs to explore & develop graph-based features, but the GSQL syntax is not so common.
  2. To train models, we need the ability to simulate relations status at any given point-in-time.

- **Means:** Provide the tools to explore, develop and produce graph-based features, support historical features values to train & validate models.

How Intuit Risk Leverage Graph Technology

Link Analysis Using Tiger Graph



Today's talk will focus here

Graph Based Features

Graph Based Features

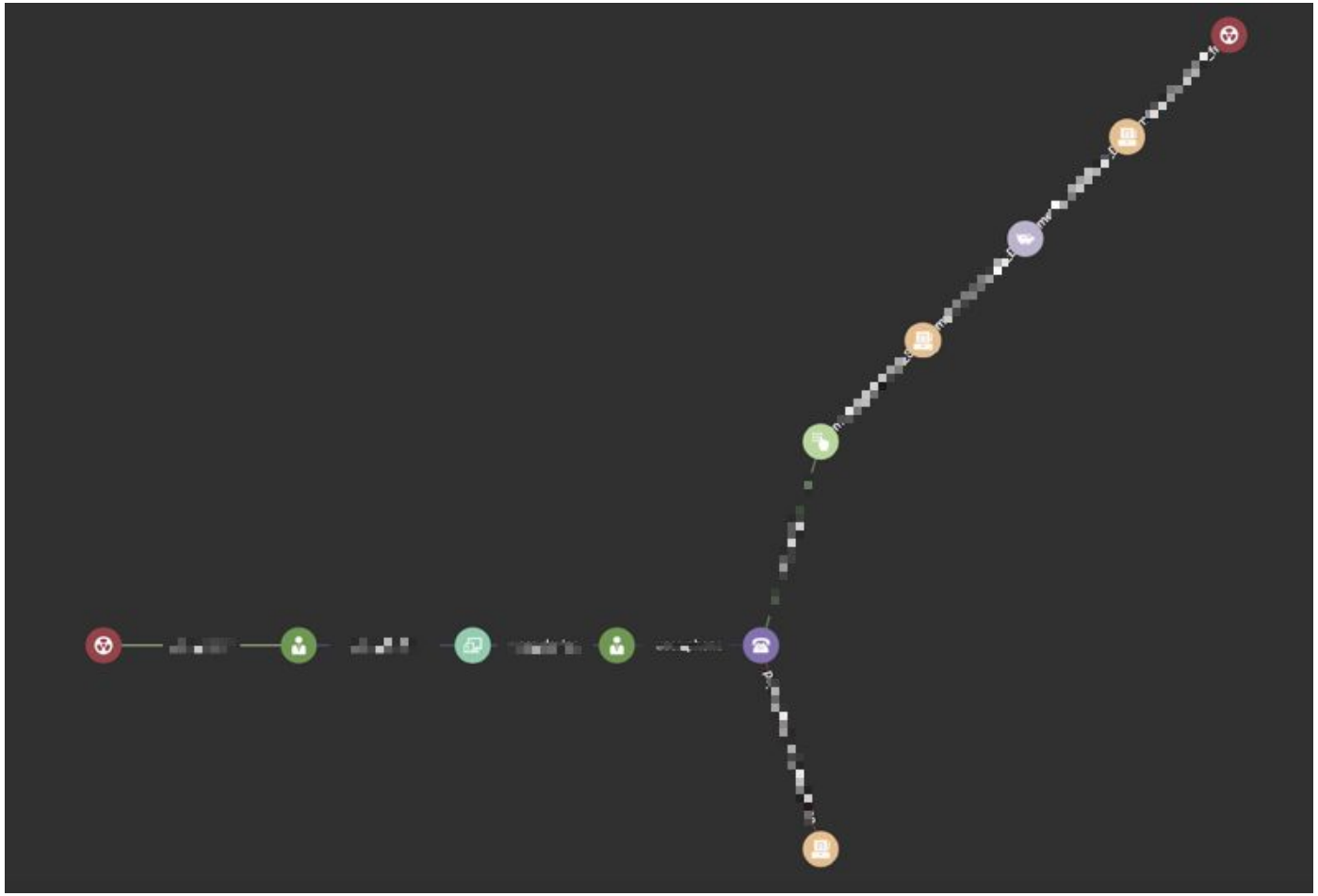
- Graph based features brings a new perspective to the prediction game
- By adding graph features to fraud detection models we can detect delicate fraud patterns

5 hops to fraud

Graph Features Examples

Here's a few graph basic features:

1. Min linked distance
2. Average linked distance
3. Number of links to target
4. Link strength
5. Path to target



Serving Graph Based Features to our Models

AI/ML Models - Classic Development Flow

A classic model development flow

- Feature exploration
- Develop the feature
- Test the feature

Data Scientists uses a SQL/Python interface to explore and develop the feature

- Calculate feature history
- Store feature historical values

Using historical data and aggregation functions to produce historical feature values

- Integrate model with features
- Train, Test & Validate model

Based on historical feature values, train, test and validate the model in an offline environment

- Integrate platform w/ source
- Deploy new feature code

Integrate model execution environment with the feature source and deploy the new model code

Graph Based Features - Development Flow

A graph based model development flow

Feature Creation

- Feature exploration
- Develop the feature
- Test the feature

Data Scientists required to use a GSQL interface to explore and develop graph based features

Challenge #1

Produce Training Data

- Calculate feature history
- Store feature historical values

Require the ability to simulate the graph status at any point-in-time and produce historical feature values

Challenge #2

Train, Test & Validate

- Integrate model with features
- Train, Test & Validate model

Based on historical feature values, train, test and validate the model in an offline environment

Deploy

- Integrate platform w/ source
- Deploy new feature code

Integrate model execution environment with the graph API and deploy the new feature code

Graph-Based Features Exploration

Explore and develop graph-based features

- **The Challenge:** Data Scientists needs to explore & develop graph-based features, but the GSQL syntax is not so intuitive.
- **Success Criteria:** Reduce the time for feature exploration and development of graph based features by 50%.
- **Means:** Provide the tools to explore & develop graph-based features, using the common Graph QL code and an accessible user interface.

Graph QL Feature Code

Query Result

Graph Based Features - Development Flow

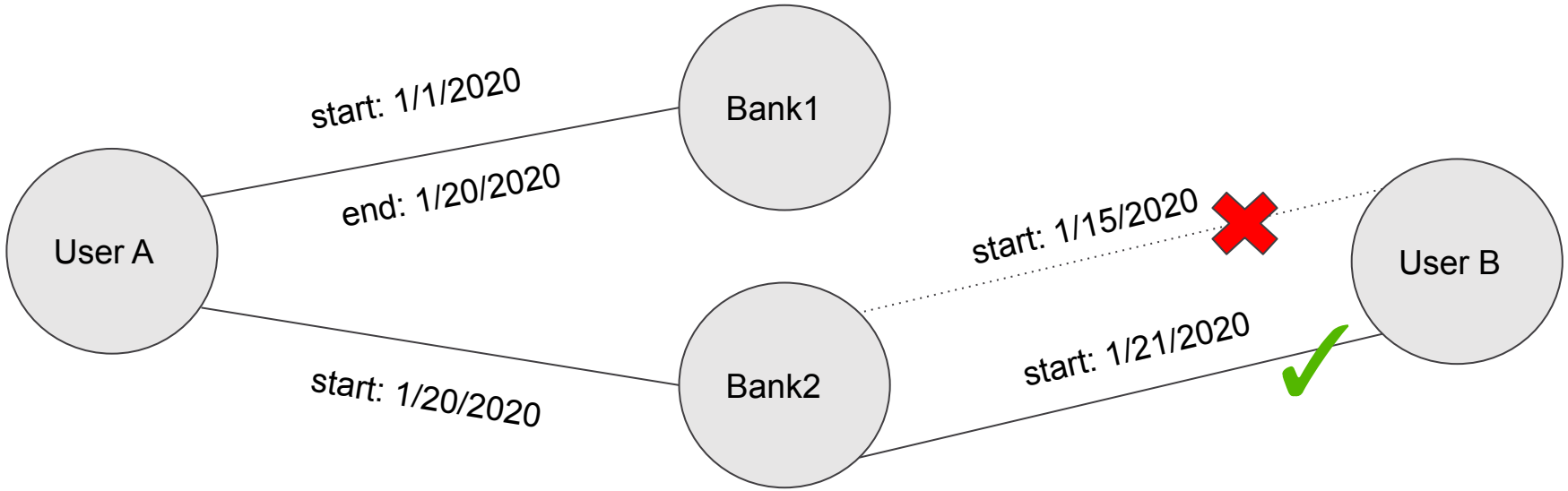
A graph based model development flow

Feature Creation	Produce Training Data	Train, Test & Validate	Deploy
– Feature exploration – Develop the feature – Test the feature	– Calculate feature history – Store feature historical values	– Integrate model with features – Train, Test & Validate model	– Integrate platform w/ source – Deploy new feature code
Data Scientists uses a Graph QL interface to explore and develop graph based features	Using “point-in-time” logic to reflect relations status at any point-in-time and produce historical feature values using a batch query system	Based on historical feature values, train, test and validate the model in an offline environment	Integrate model execution environment with the graph API and deploy the new feature code

Produce training data sets for graph-based features

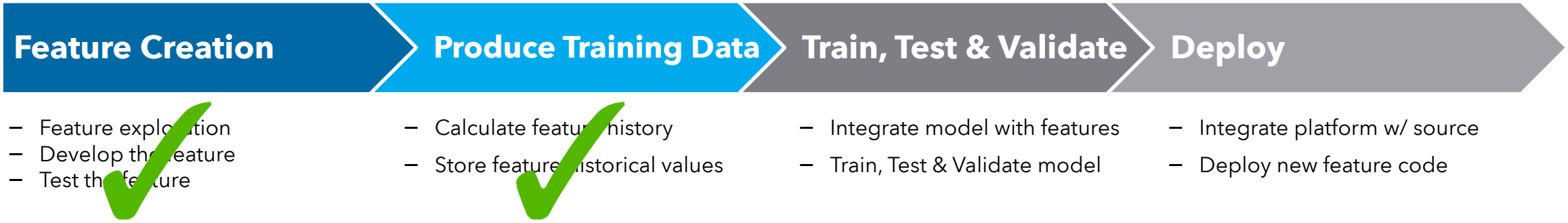
Generate training data sets of graph-based features

- **The Challenge:** Produce historical graph-based feature values for model training and testing
- **Success Criteria:** Train & Test our ML models with graph-based features.
- **Means:** Provide the tools to produce graph-based features for any given point in time in history using



Graph Based Features - Development Flow

A graph based model development flow



Data Scientists uses a Graph QL interface to explore and develop graph based features

Using “point-in-time” logic to reflect relations status at any point-in-time and produce historical feature values using a batch query system

Based on historical feature values, train, test and validate the model in an offline environment

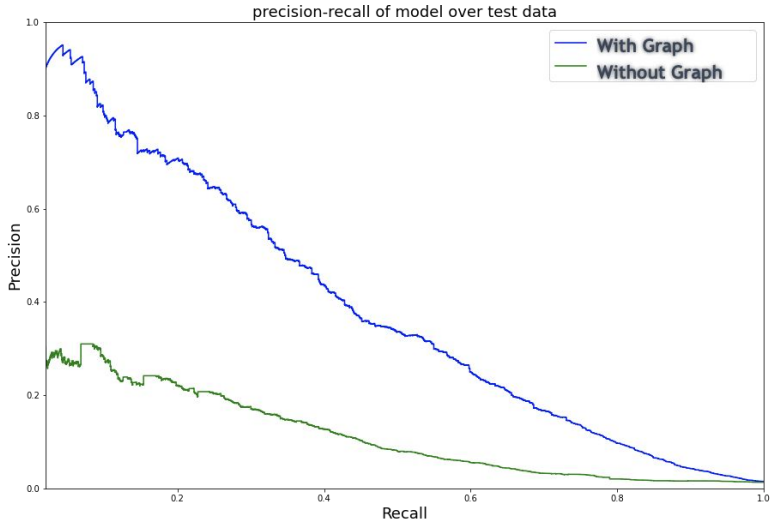
Integrate model execution environment with the graph API and deploy the new feature code

Graph Based Features Boost AI

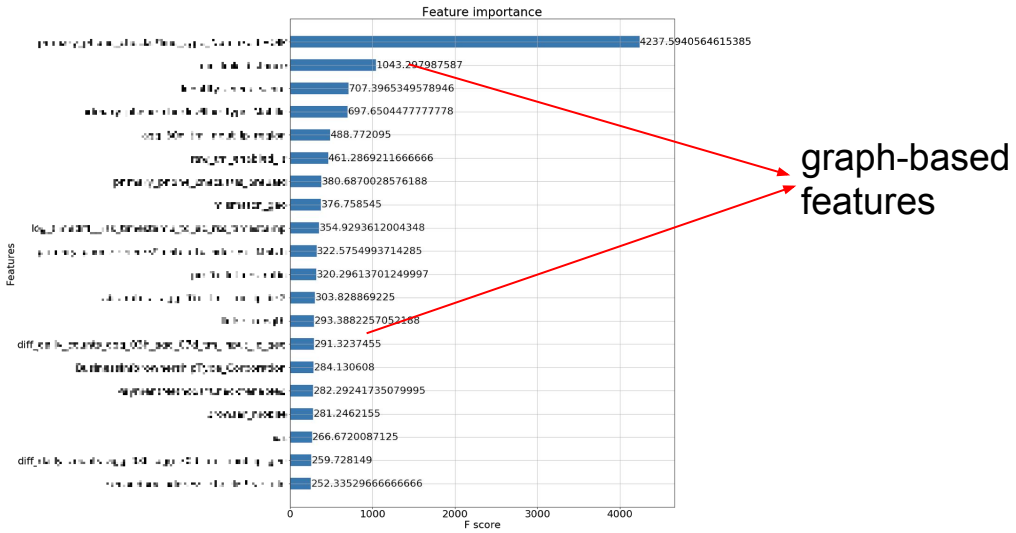
Graph-Based Features - Improved Performance

Benefit of using graph-based features

- **Model recall increased by ~50%**
- **Model precision increased by ~50%**



- **Two out of the top 20 features are graph based features**



Summary & Key Takeaways

Key Takeaways from that Talk

Summary of main topics discussed

- **Listen to your Customers**

> "Customers don't care about your solution, they care about their problems"
>
> Dave McClure

- **Think through the Process**

> "If you think good design is expensive… you should look at the cost of bad design"
>
> Ralf Speth

Thank You!

Q&A