
# Data Science with Graph Algorithms and Machine Learning

Xinyu Chang, Director of Customer Solutions

Nalu Zou, Solution Architect

# Overview: Why graphs for data science and machine learning?

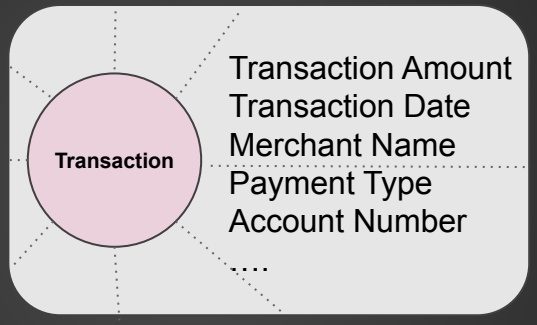
# Why Graphs?

TigerGraph can bring unique value to your machine learning process.

1. Realtime feature extraction + prediction (e.g. anti fraud)
2. Introduce new features to the model.
3. Unique insights from graph algorithm.
4. Improved machine learning model accuracy.

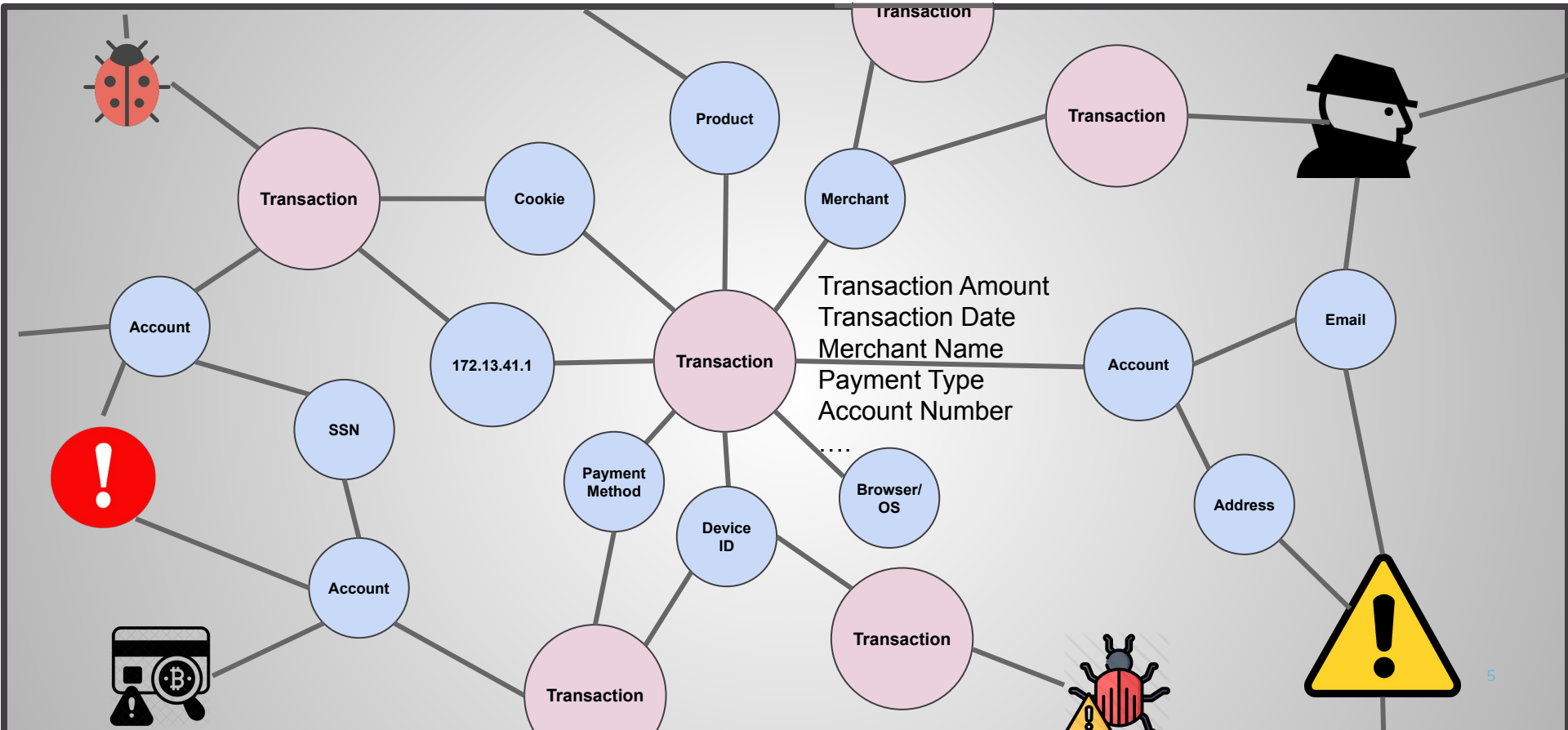
# Why Graphs?

An entity is not just an entity by itself...



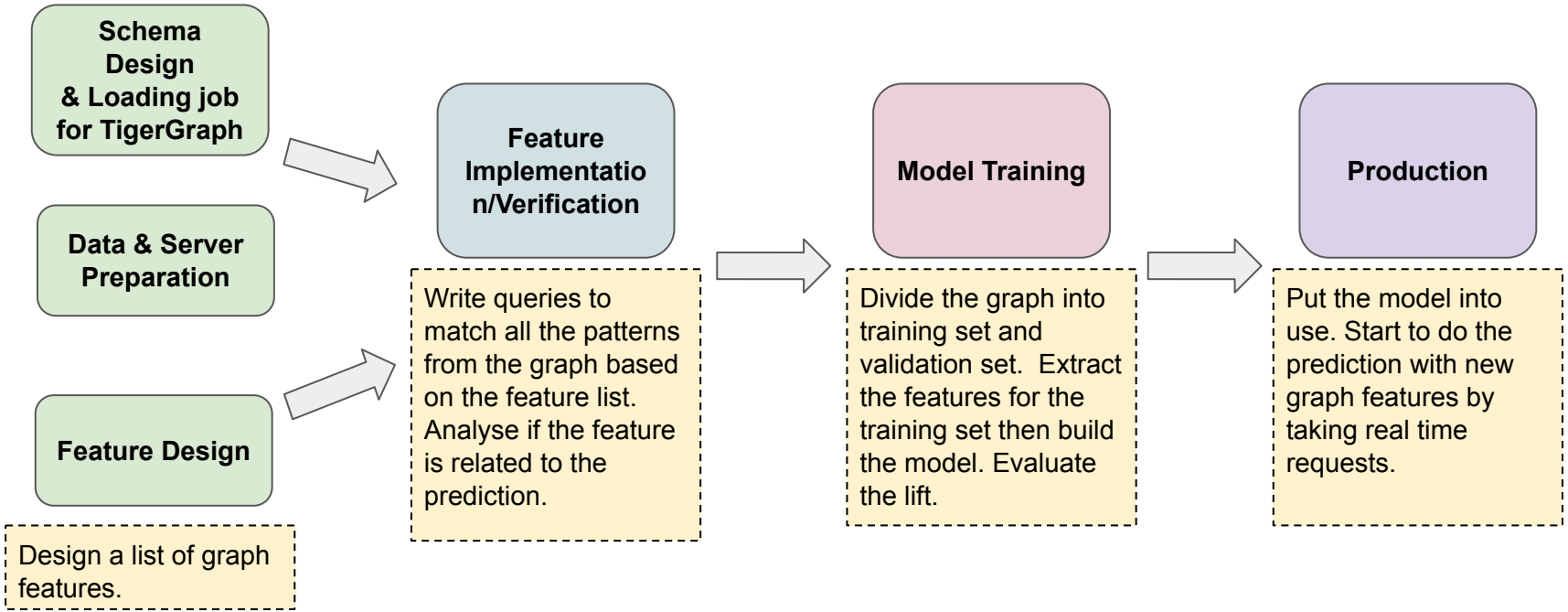

It is everything that connected...

# Why Graphs?


Transaction
Product
Transaction
Transaction
Cookie
Merchant
Account
172.13.41.1
Transaction
Transaction Amount
Transaction Date
Merchant Name
Payment Type
Account Number
....
Account
Email
SSN
Payment Method
Device ID
Browser/ OS
Address
Account
Transaction
Transaction

# Overview: How does TigerGraph work in your ML flow?

# TigerGraph ML Working Flow

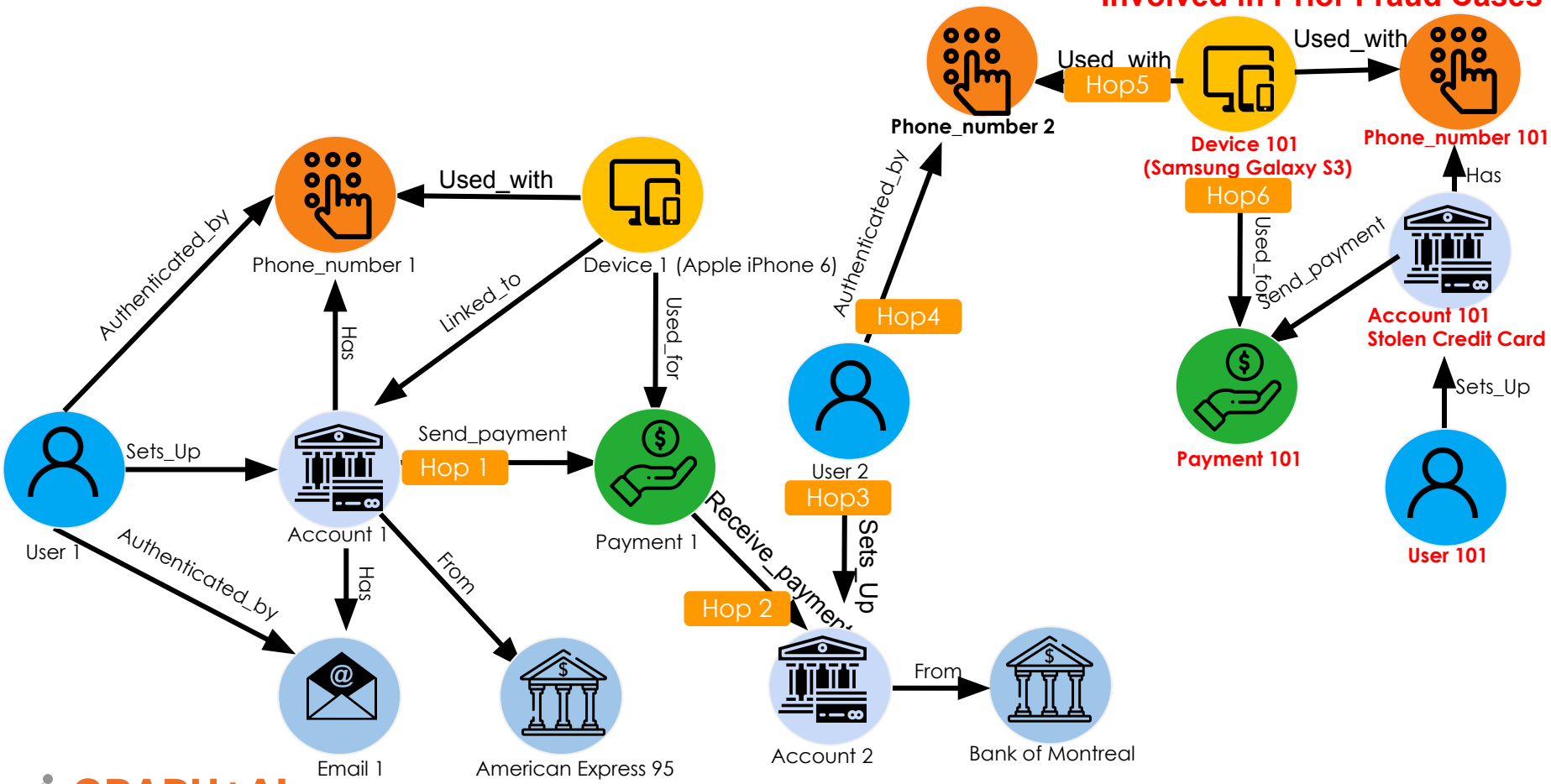


# What are graph features.

The features that show the interconnectedness of relationships between entities. Usually it takes many hops on graph to discover that kind of hidden insight which makes other data platform very difficult to finish this kind of computation within reasonable time.

The common graph features are:

1. Manually designed graph features
2. Graph algorithm result
3. Graph embeddings


Involved in Prior Fraud Cases
Phone_number 2
Used_with
Hop5
Used_with
Device 101
(Samsung Galaxy S3)
Phone_number 101
Hop6
Has
Used_for
Send_payment
Account 101
Stolen Credit Card
Payment 101
Sets_Up
User 101
Authenticated_by
Hop4
Phone_number 1
Used_with
Device 1 (Apple iPhone 6)
Linked_to
Has
Used_for
Authenticated_by
Sets_Up
Send_payment
Hop 1
User 2
Hop3
Sets_Up
User 1
Account 1
Payment 1
Receive_payment
Hop 2
Authenticated_by
Has
From
From
Email 1
American Express 95
Account 2
Bank of Montreal

# Manually Designed Graph Features

Here are more manually designed graph feature examples, usually this kind of features is designed based on the behavior or pattern to be predicted.

1. Number of fraudulent accounts in a community
2. Number of device/IP/cookie in 2/4/6 hops
3. Number of shortest paths to fraudulent accounts.
4. A phone call made to a person that called your other callees.

Many **JOINs** need to be performed if using RDBMS.

# Graph Algorithm Features

The result of graph algorithm can serve as machine learning features as well since they provide signals of the graph topology from different perspectives.

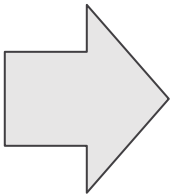
1. PageRank score -> Vertex influence
2. Closeness Centrality -> Position in a community, close to center or boundary
3. Betweenness Centrality -> Number of shortest path through a vertex
4. Louvain -> Community/Group/Ring detection
5. Weakly Connected Component -> Detection maximal subgraph connected via undirected edges.

Many **JOINs** need to be performed if using RDBMS.

# Graph Embedding Features

## Random Walk + Node2Vec

```
3156 1408 460 5705 4845 2786 3867 8099 9902 7098 7120 4839 7442 3510 9905 1971
5429 9653 9413 394 5298 4085 1531 3198 739 806 446 9960 9484 4374 7098 6568 262
8589 7490 3479 1705 2625 3932 7356 7720 6786 8095 2775 9297 2663 8270 5131 1858
7359 7515 1438 1970 217 2613 2405 9919 286 1681 3374 2059 448 3528 3504 4669 82
9 6360 2241 4805 2733 7132 6505 7806 9037 8533 176 6773 4652 589 6749 3579 3713
5153 36 548 283 2165 9832 750 2904 9517 5429 5770 3339 2834 3869 3561 6909
5386 2454 2869 3397 9935 9847 1801 109 2638 381 1621 4374 1398 4221 4374 10259
94 6073 6567 8212 4708 8208 4374 5686 4519 5974 7515 1625 3820 8919 2241 4550 2
3284 5346 902 3813 4496 5681 7496 4669 5264 4674 667 5499 3133 6715 276 856 282
2364 5326 5686 8116 7373 1555 3608 1943 7423 4667 587 4157 10240 9997 3643 9047
1 7061 6004 9355 13 7989 9318 7373 3534 1464 26 4669 7205 9526 8390 176 8893 34
2 1136 1248 886 28 1136 1231 446 77 4839 7208 1666 6764 4512 8259 7376 9844
7684 9358 119 991 7946 8157 3911 8736 8229 2112 8662 8248 8296 8408 6664 2041 9
358 1811 1553 2369 8476 5091 3608 1597 6318 5906 1829 4407 7659 7709 5014 9419
751 9905 7373 9821 1226 8812 9526 4770 5589 4446 7806 792 4414 267 3262 2698
5112 5918 9203 2690 4997 5000 4374 7872 739 4109 6376 739 8859 9128 7806 1291 5
375 3526 7288 8525 9187 9794 9961 5546 2622 5374 4849 2638 7262 2558 9650 7098
58 3244 2085 2945 2774 8025 8591 1226 4125 7417 1417 1203 2501 2561 739 535
10143 9697 10257 10269 8456 1597 1283 6952 5484 3262 4707 4908 645 1463 645 141
5 1935 916 8099 8344 9607 7333 8317 176 8557 3701 3224 2085 7132 2796 6872 5377
 6395 3775 3366 3037 5467 2751 902 8476 3242 533 4719 4862 3510 5976 5775 6054
858 2499 3198 2201 2796 6956 5974 2743 1887 2548 1291 4176 10192 7093 5429 5596
 8122 8171 3652 6959 7800 5259 7633 7373 730 6141 968 4984 2983 6959 6427 6959
5360 4601 739 5109 7806 1665 1232 667 4095 5681 7076 7237 1050 1876 2327 2781 7
```



```
4839 0.0092476 -0.0299002 -0.0354903 0.01166
176 0.0301448 -0.0458716 -0.104322 -0.165577
4374 0.135287 -0.161183 -0.0627228 -0.011731
8157 0.189139 -0.128897 0.0255433 -0.0461897
1226 0.0708157 -0.0472967 0.0306176 -0.09009
4997 -0.065632 -0.0706793 -0.103198 0.045023
4984 0.06196 0.0126079 -0.136605 -0.0582171
8859 0.0587633 -0.113262 0.0174828 -0.051839
645 0.0569722 -0.0658353 -0.189108 -0.073751
446 0.191666 -0.113103 -0.0390296 -0.0614881
7806 0.0684731 -0.105188 -0.0602134 -0.10622
7098 0.0951419 -0.185009 -0.0272823 -0.02497
3198 0.0488285 -0.00531387 -0.0953742 -0.165
2521 0.0650091 -7.95217e-05 -0.0843067 0.030
233 -0.0581332 -0.104116 -0.105006 -0.156256
667 0.0778987 0.0663187 -0.0168721 -0.051862
739 0.0577648 -0.0433264 0.0137341 0.0037011
```

# ML Feature Library

Our plan is to build a easily accessible and deployable ML feature library that can help the users to build their graph feature table for ML model.

We will start with

1. Credit card/ Loan application fraud.
2. Payment Fraud.
3. Incentive Fraud.
4. Phone Call Fraud.
5. And many more...