TigerGraph

TigerGraph for Machine Learning

INCREASE THE ACCURACY OF MACHINE LEARNING WITH TIGERGRAPH

International Data Corporation (IDC) forecasts that spending on Artificial Intelligence and Machine Learning will grow from $12B in 2017 to $57.6B by 2021(Source: Forbes, Feb 2018). Machine learning is being applied to a variety of use cases including fraud prevention, anti-money laundering (AML) and eCommerce product recommendation. As you apply machine learning to identify anomalous behavior such as finding fraudsters or money launderers, it is akin to finding needles in a massive haystack - companies must sort and make sense of massive amounts of data in order to find the “needles” or in this case, the fraudsters. Consider a phone company which has billions of calls occuring in its network on a weekly basis. How do we train the machine learning algorithms to identify signs of fraudulent activity from a mountain or haystack of calls?

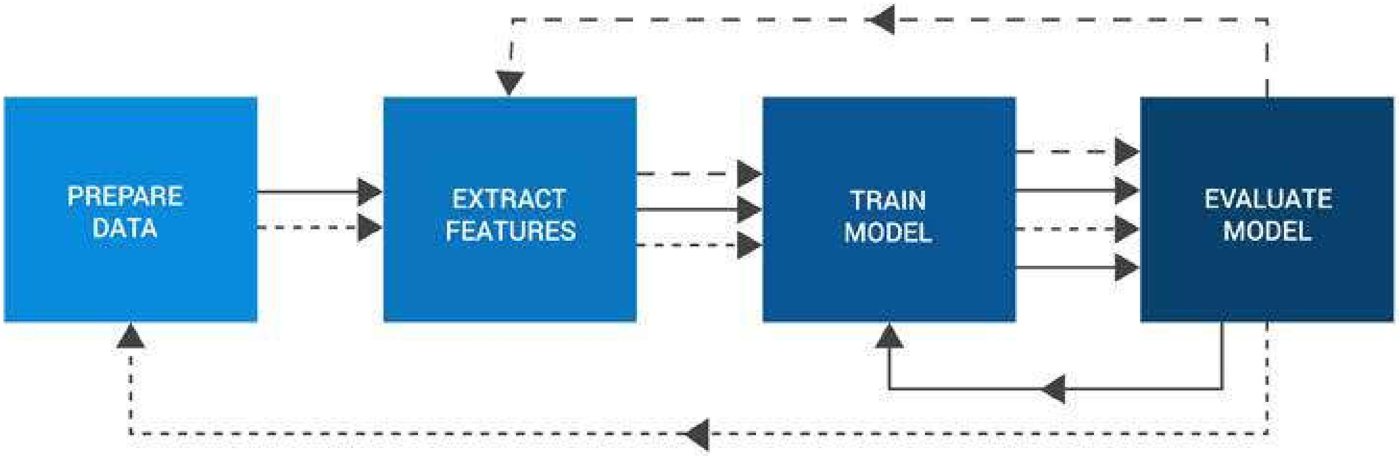
CURRENT MACHINE TRAINING APPROACHES ARE MISSING THE MARK

Let’s explore the phone company example to consider current approaches for identifying fraudsters based on machine learning. Supervised machine learning algorithms need training data – in this case phone calls identified as calls from confirmed fraudsters. There are two problems with the current approach, including both the quantity and quality of training data.

Confirmed fraudulent activity in phone networks currently constitutes less than 1% of total call volume. So, the volume or the quantity of training data with confirmed fraud activity is tiny. Having a small quantity of training data in turn results in poor accuracy for the Machine Learning algorithms.

Features or attributes for finding a fraudster are based on simple analysis. In this case they include calling history of a particular phones to other phones that may be in or out of the network, the age of a pre-paid SIM card, percentage of one-directional calls made (cases where the call recipient did not return a phone call) and the percentage of rejected calls. These simplistic features tend to result in a lot of false positives. It’s no wonder when you consider how in addition to a fraudster, these features may also fit the behavior of a sales person or a prankster.

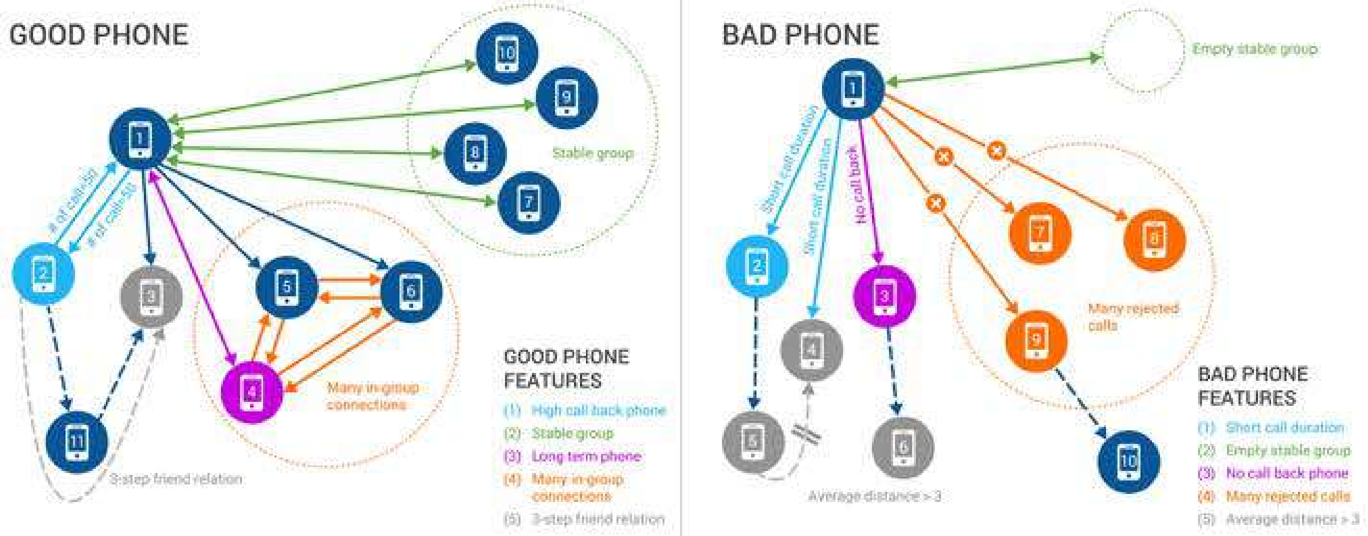
ITERATIVE MACHINE LEARNING PROCESS



TRAINING THE MACHINE FOR FRAUD DETECTION WITH GRAPH BASED FEATURES

A large mobile operator uses TigerGraph, the next generation graph database with Real-Time Deep Link Analytics, to address the deficiencies of current approaches for training machine learning algorithms. The solution analyzes over 10 billion calls for 460 million mobile phones, and generates 118 new graph features for each mobile phone. These features are based on deeper analysis of the calling history, and go beyond the immediate recipients for calls to the extended network.

Detecting phone-based fraud by analyzing network or graph relationship features



This diagram illustrates how the graph database identifies a phone as a “good” or a “bad” phone. A bad phone requires further investigation to determine whether it belongs to a fraudster. A customer with a good phone calls other subscribers, and majority of their calls are returned. This shows familiarity or trusted relationships between the users. A good phone also regularly calls a set of others phones every week or month and this group of phones is fairly stable over a period of time (“Stable Group”). Another feature indicating good phone behavior is when a phone calls another that has been in the network for many months or years and receives return calls. We also see a high number of calls between the good phone, the long-term phone and other phones within a network calling both these numbers frequently. This indicates many in-group connections for the good phone.

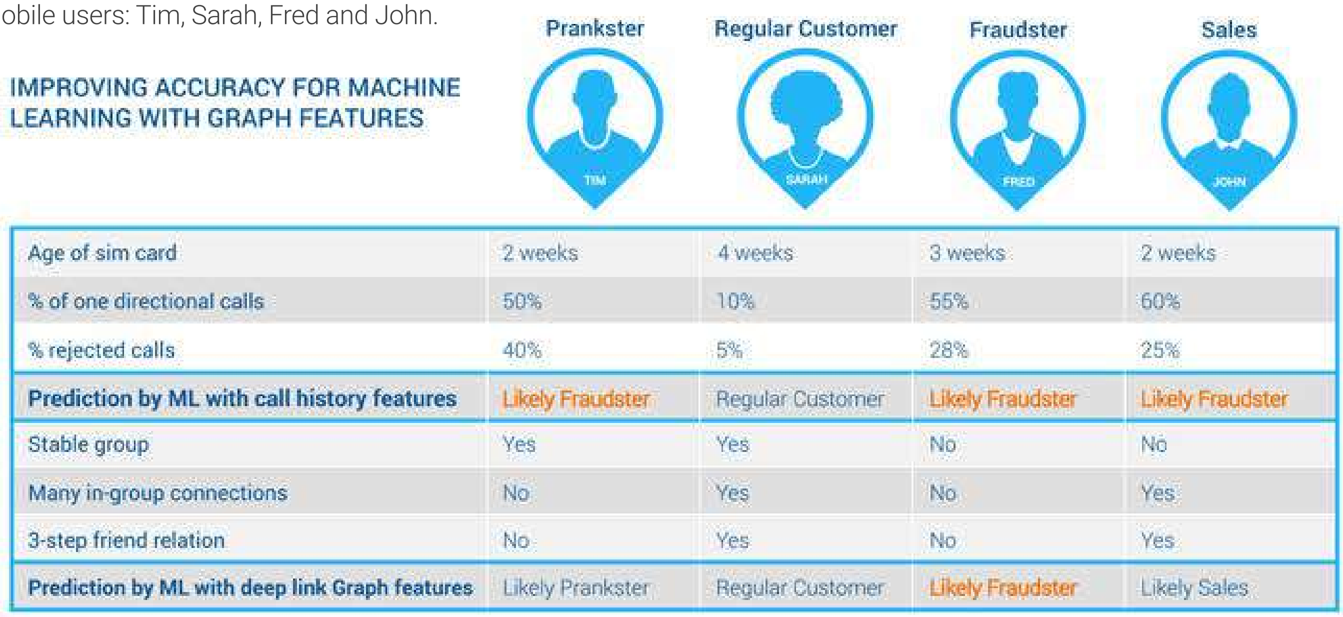
Lastly, a good phone is often involved in a three step friend connection – meaning the good phone calls another phone, phone 2, which calls phone 3. The good phone is also communicates directly with phone 3. This indicates a three step friend connection, indicating a circle of trust and interconnectedness.

By analyzing such call patterns among phones, TigerGraph can easily identify bad phones, which are phones that are likely to be involved in a scam. These are phones that have short calls with multiple good phones, but do not receive a call back. They also do not have a stable group of phones called on a regular basis (representing an “empty stable group”). When a bad phone calls a long-term customer in the network, the call is not returned. The bad phone also receives many rejected calls and lacks three step friend relationships.

TigerGraph creates more than 118 new features that have correlation with the good and the bad phone behavior for each of the 460 million mobile phones in our use case. In turn, it generates 54 billion new training data features to feed Machine Learning algorithms. This has led to dramatic improvement in the accuracy of machine learning for fraud detection, resulting in fewer false positives as well as lower false negatives.

IMPROVING MACHINE LEARNING ACCURACY WITH GRAPH BASED FEATURES

To see how graph based features improve accuracy for machine learning, let's consider this example using profiles for four mobile users: Tim, Sarah, Fred and John.



IMPROVING ACCURACY FOR MACHINE LEARNING WITH GRAPH FEATURES

TIM
SARAH
FRED
JOHN

	Prankster	Regular Customer	Fraudster	Sales
Age of sim card	2 weeks	4 weeks	3 weeks	2 weeks
% of one directional calls	50%	10%	55%	60%
% rejected calls	40%	5%	28%	25%
Prediction by ML with call history features	Likely Fraudster	Regular Customer	Likely Fraudster	Likely Fraudster
Stable group	Yes	Yes	No	No
Many in-group connections	No	Yes	No	Yes
3-step friend relation	No	Yes	No	Yes
Prediction by ML with deep link Graph features	Likely Prankster	Regular Customer	Likely Fraudster	Likely Sales

Traditional calling history features, such as the age of the SIM card, percentage of one directional calls and percentage of total calls rejected by their recipients, result in flagging three out of four of the customers, Tim, Fred and John as likely or potential fraudsters as they look very similar based on these features. Graph based features with analysis of deep link or multi-hop relationships across phones and subscribers helps Machine Learning classify Tim as a prankster, John as a sales person, while Fred is flagged as a likely fraudster.

Tim has a stable group, which means he is unlikely to be a sales guy, since sales people call different numbers each week. Tim doesn't have many in-group connections, which means he is most likely calling strangers. He also doesn't have any 3-step friend connections to confirm that the strangers he is calling know each other. It is very likely that Tim is a prankster based on these features.

John doesn't have a stable group, which means he is calling new potential leads every day. He calls people with many in-group connections. As John presents his product or service, some of the call recipients are most likely introducing him to other contacts if they think that the product or service would be interesting or relevant to them. John is also connected via 3-step friend relations among call recipients. It indicates that John is closing the loop as an effective sales guy, navigating the friends or colleagues of his first contact within a group to reach the final buyer for his product or service. The combination of these features classifies John as a sales person.

Fred doesn't have a stable group, nor does he interact with a group that has many in-group connections. Fred also does not have 3-step friend relations among the people he calls. This makes him a very likely candidate for investigation as a phone scam artist or a fraudster.

Going back to the original analogy, we are able to find the needles in the haystack, in our case it's Fred the potential fraudster, by leveraging graph analysis. This is achieved by using the graph database to analyze and identify new features from interconnected data. The machine in turn is trained with the highly correlated and large volume of training data (graph based features), making it smarter and more successful in recognizing potential scam artists and fraudsters.

TigerGraph

TigerGraph for Machine Learning

LEARN MORE

Graph based features generated in real-time by TigerGraph are being used for a host of use cases beyond identifying phone-based scam. These include training the machine learning algorithms to detect various other types of anomalous behavior, including credit card-related fraud which affects all the merchants selling products or services via eCommerce, and money laundering violations spanning the entire financial services ecosystem including banks, payment providers and newer crypto currencies such as Bitcoin, Ether and Ripple.

eCommerce companies are also using graph based features to create product recommendations with analysis of customer's buying behavior, other customers in their extended network and also those who have similar buying preferences. These new features are fed as the training data to the machine learning algorithms to improve accuracy for future recommendations.

CONTACT

TigerGraph
3 Twin Dolphin Drive, Suite 225
Redwood City, California 94065
United States

www.tigergraph.com
Twitter @TigerGraphDB

CUSTOMERS AND USE CASES

TigerGraph's real-time analytics on giant graphs is the engine behind fraud prevention at the world's largest e-commerce provider, recommendations at the world's largest mobile e-commerce company, and network management at the world's largest electric grid company.

ANTI-FRAUD & ANTI-MONEY LAUNDERING:

TigerGraph's deep link analytics and big graph capabilities uncovers hard-to-find patterns and connections. Financial crimes teams can investigate specific transactions, high-risk customers or counterparty relationships using a graph modeling approach, in real-time.

MASSIVE-SCALE TRANSACTION PROCESSING:

One of the world's largest e-payment companies uses TigerGraph to handle a graph with 100B+ vertices and 2B+ real-time updates/day. 20-node cluster, 2+ years in production, full ACID.

SUPPLY CHAIN INTELLIGENCE:

TigerGraph provides real-time visibility and analytics into key supply chain operations including order management, shipment status and other logistics.

CUSTOMER INTELLIGENCE:

TigerGraph empowers organizations to quickly deploy powerful relationship analysis capabilities. Real-time capabilities allow retailers to quickly synthesize and make sense of customer behavior and activities, smartly clustering products and make real time, personalized recommendations.

SMART GRID:

Working closely with leading energy and utility companies, TigerGraph has pioneered Native Parallel Graph approaches that help companies monitor and analyze power flows, detect bottlenecks, and alert personnel about grid performance issues.

About TigerGraph

TigerGraph is the world's fastest graph analytics platform powered by Native Parallel Graph (NPG) technology. TigerGraph fulfills the true promise and benefits of the graph platform by tackling the toughest data challenges in real time, no matter how large or complex the dataset. TigerGraph supports applications such as IoT, AI and machine learning to make sense of ever-changing big data. For more information, visit www.tigergraph.com.