# Catching Bad Guys with Graph Mining

**Suspicious network patterns may be the key to detecting criminals and fraudsters on e-commerce sites.**

*By Polo Chau*

DOI: 10.1145/1925041.1925044

The Internet opened a new operational channel for many services, like online auctions, shopping, and banking. Every day, millions of transactions happen over these services, collectively known as e-commerce, each in the blink of an eye. Unfortunately, the monetary incentives intrinsic to e-commerce attract the attention of criminals (the bad guys), leading to some new types of crime. For instance, multiple online identities are easy to create: a perpetrator could use his many alter-egos to execute sophisticated schemes, burying his trail deep under false covers, and evading traditional detection methods that only examine identities individually. Furthermore, e-commerce generates so much data that discovering the bad guys, or their alter-egos, among the overwhelming amount of data seems daunting. On the bright side, bad guys leave trails. If we look closely, sometimes we can identify their suspicious operation patterns. What are the patterns? How do we detect them?

Perhaps these questions would be easier to answer if we view the world a little differently—as a giant graph (or network) of nodes (people) and edges (relationships among people). Detecting bad behaviors then becomes locating some suspicious patterns as collections of incriminating relationships in the graph. This process of locating useful information and patterns in graph data is called *graph mining*, and it has been successfully applied to many domains. Here we look at how it works in e-commerce to help catch the bad guys.

## DETECTING FRAUD BY AGGREGATING INCRIMINATING EVIDENCE

Online auctions like eBay are popular avenues for buying and selling almost any items imaginable. Most items will be delivered, but some unfortunately will not, because some sellers are crooks who never intend to do so. How do they convince the buyers that they are legitimate? They game the *reputation systems* that most online auctions set up to help buyers gauge sellers' trustworthiness. One type of fraud scheme works as follows. The bad guy first creates multiple identities

**"E-commerce generates so much data that discovering the bad guys, or their alter-egos, among the overwhelming amount of data seems daunting. On the bright side, bad guys leave trails..."**

**"E-commerce has redefined crime. We now see new breeds of online crime where technologically savvy criminals exploit not only the weaknesses of human nature, but also the systems originally designed to protect online shoppers."**

in the online auction, dividing them into two groups ("fraudsters" and "accomplices"). The *fraudsters* rarely trade among themselves, and neither do the *accomplices*. Then, the bad guy uses the accomplices to artificially boost the reputation of the *fraudsters*. The *accomplices* typically act like normal, honest users who buy and sell items (usually cheap, to lower operating costs), but they sometimes sell expensive items to the *fraudsters*, leaving glowing comments about how the buyers (*fraudsters*) are *good guys* ("paid on time", "easy to communicate with", etc.). After the *fraudsters* have reached high reputation, they launch deceptive auctions to sell expensive items (e.g., big-screen TVs), usually at bargain prices, to the victims (*honest* people). Those items will never be delivered.

We call the above interaction pattern a "bipartite core" (**Figure 1**), where two types of nodes (*fraudsters* and *accomplices*) only interact with nodes of different types, but not with their own. This pattern forms the *infrastructure* that criminals set up *before* they carry out auction fraud. But to the naked eye, the associations between the identities involved in the deceptive *bipartite core* pattern might not be apparent.

The NetProbe system [1] was developed to dig out these identities in this pattern, by automatically scanning connections between buyers and sellers, several layers deep, to look for artificial feedback, revealing identities and their associations that match the *bipartite* core. The system ran through over one million transactions and correctly picked out dozens of previously identified criminals; it also identified tens of probable fraudsters and apparent accomplices.

Under the hood, NetProbe uses an inference algorithm called *Belief Propagation* to infer which nodes in the auction graph are most likely to be *fraudsters* and *accomplices*. The system first uses heuristics to assign a vector of three probabilities—called the node's belief—to each node: a *fraudster* probability, an *accomplice* probability, and an *honest* probability. For example, if an identity has been active for many years and has not received any negative comments from other people, then that identity has a high *honest* probability; if an account was recently shut down right after it received many complaints, then it has a high *fraudster* probability. These three probabilities sum up to 1.

**Table 1. Conditional probability table describing a "bipartite core"; ε is a small constant close to zero. For example, entry (F, A), with a value of 1-2ε, describes a very high probability of a node's neighbor being an accomplice (A) given the node itself being a fraudster (F).**

| | F | A | H |
|---|---|---|---|
| Fraudster (F) | ε | 1-2ε | ε |
| Accomplice (A) | 0.5 | 2ε | 0.5-2ε |
| Honest (H) | ε | (1-ε)/2 | (1-ε)/2 |

NetProbe's algorithm then uses the matrix in Table 1 to transform each node's belief into a *message* (also a probability vector) that the node will send to each of its neighbors; the message represents what the node thinks about its neighbors. The transformation is similar to multiplying the node's belief with the matrix. For example, if a node has high *fraudster* probability, then applying the transformation on it will create a message for each neighbor that says the neighbor is likely an accomplice. All nodes simultaneously send out messages to their neighbors. Each node gathers its incoming messages, multiplies them into one vector (which also resolves competing messages similar to *majority voting*), then sets that vector as the node's new belief. Finally, the node generates new messages for its neighbors using its updated belief. This whole process continues until all node beliefs do not change anymore. NetProbe then calls out the likely fraudsters and accomplices, and warn off potential bidders.

The idea of propagating information across a graph and aggregating it to produce high-level conclusion is powerful. It inspired the creation of the generalized Snare system [2] applicable for various kinds of fraud and anomaly detection tasks. Snare was used on some general ledger data (a network of interconnected accounts) to detect financial fraud, boosting the detection rates of misstated accounts by 5.5 times.

## USER-CENTERED AND AUTOMATIC PATTERN DETECTION

Sometimes, analysts need to experiment with multiple patterns that, hopefully, would match the actual incriminating patterns. Creating a separate algorithm for each such pattern is costly and time-consuming, especially since most patterns will end up not being useful. Can we provide one tool that detects a wide range of patterns quickly and easily?

The Graphite system [3] aims to meet this challenge. It provides a direct-manipulation user interface for the user to construct the query pattern by placing nodes on the screen, assigning types to them, and connecting the nodes with edges. For example, the query pattern in **Figure 2** asks for money laundering rings of alternating businessmen and bankers. Graphite then locates the pattern's exact and approximate matches in a large graph of the user's choosing. Graphite advances over existing algorithms that detect only structural patterns without considering the types of the nodes that compose the patterns; it enables more specific patterns to be found. Consider a communication network where each node is a person from a country (country is the node type). Our analyst Laura wants to locate four

collaborators who are from Japan, Italy, Canada, and Greece respectively, and she believes they likely form a clique (i.e., every pair has communicated). With Graphite, Laura sketches a 4-node clique as the query pattern and assigns the countries as the node types. But if she was to use another tool where the node types cannot be specified, any 4-node cliques will be returned (like a family of four who all reside in the US), overwhelmingly Laura with irrelevant information.

The Holy Grail of anomaly detection is that the detection happens automatically and human does not need to do anything at all. While this may seem to be a distant goal, the Oddball system [4] is a big step towards such goal. The main idea behind Oddball is that it extracts a set of features (without human intervention) that summarize each node's neighborhood subgraph, called the node's "egonet," which includes the node's immediate neighbors and all edges in the neighborhood. Then, Oddball uses unsupervised methods that automatically correlate pairs of features and pinpoints nodes whose features significantly deviate from those of the rest of the nodes. Oddball can detect several important patterns, such as near-cliques and near-stars (by correlating the total edge weight and total edge count in the egonet). For example, in a who-called-whom network, the center of a near star could be a telemarketer who has called many random people, and a near clique could be a close-knit group of friends.

E-commerce has redefined crime. We now see new breeds of online crime where technologically savvy criminals exploit not only the weaknesses of the human nature, but also the systems originally designed to protect online shoppers. Many criminals have learned to cover their tracks with the large amount of data generated by e-commerce, and obfuscate law enforcement with multiple fake virtual identities. As e-commerce thrives and the online world becomes even more connected, tools and methods such as those from graph mining will play an increasingly important role in untangling the many layers of sophisticated organization and schemes crafted by criminals. Will online crime be eliminated? Perhaps not. But our effort will force crooks to resort to more complex schemes that incur more effort and higher cost, so crime will be increasingly difficult to commit. Then, perhaps, fewer bad guys would attempt to get on the wrong side of the law.

**Figure 1. A "near bipartite core" of *fraudsters* and *accomplices*. *Honest* identities are not shown. Note that each *fraudster* has traded with most, but not all, *accomplices*; hence it is a *near*, but *incomplete*, core.**

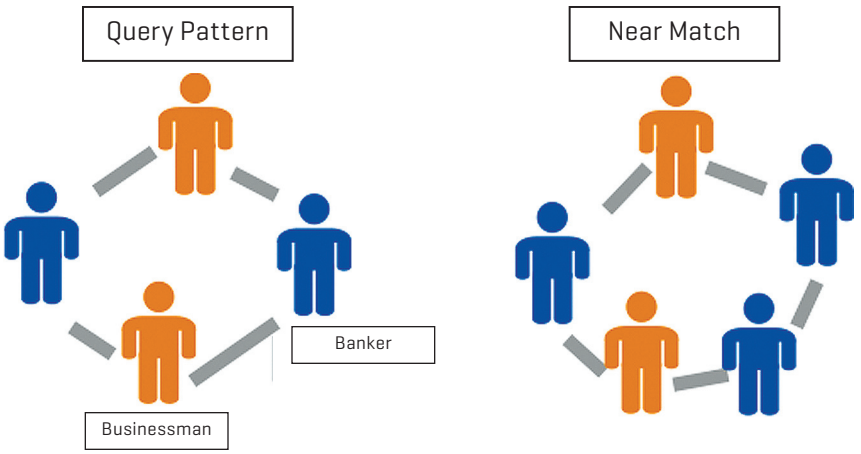
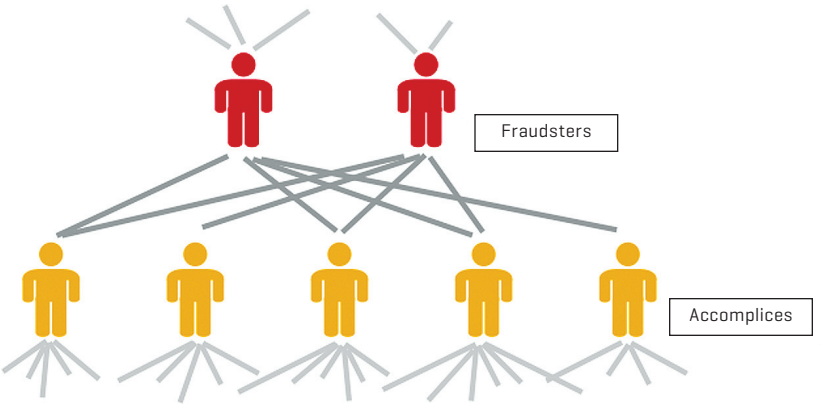



**Figure 2. Given a query pattern, such as a money laundering ring (left), the Graphite system can find both exact and near matches that tolerates a few extra nodes (right).**




**Biography**

Polo Chau is a Ph.D. student in the Machine Learning Department at Carnegie Mellon University. His research intersects graph mining and human-computer interaction. He builds interactive systems that help analysts explore and make sense of large graph data, find patterns, detect fraud, and spot anomalies. His work on fraud detection in online auctions appeared in the *Wall Street Journal* and many other media outlets. He was a Symantec fellow for two consecutive years, and is an avid designer, having won many awards.

**References**

1. Pandit, S., Chau, D.H., Wang, S. and Faloutsos, C. NetProbe: A fast and scalable system for fraud Detection in Online Auction Networks. In *Proc. WWW 2007*, 201-210.
2. McGlohon, M., Bay, S., Anderle, M., Steier, D. and Faloutsos, C. SNARE: A link analytic system for graph labeling and risk detection. In *Proc. KDD 2009*, 1265-1274.
3. Chau, D.H., Faloutsos, C., Tong, H., Hong, J.I. Gallagher, B. and Eliassi-Rad, T. GRAPHITE: A visual query system for large graphs. In *Proc. ICDM 2008*, 963-966
4. Akoglu, L. McGlohon, M. and Faloutsos, C. OddBall: Spotting anomalies in weighted graphs. In *Proc. PAKDD 2010*, 410-421.