Fraud Detection in Telecommunications: A Historical Perspective and Lessons Learned

Richard A. Becker, Chris Volinsky, and Allan R. Wilks*

April 16, 2009

Abstract

Fraud detection is an increasingly important and difficult task in today's technological environment. As consumers are putting more of their personal information online and transacting much more business over computers, the potential for lost revenue from fraud is in the billions of dollars, not to mention the damage done by identity theft. This paper examines the future by taking a look back at the history of fraud detection at AT&T, which was one of the first companies to have to address fraud in a systematic way to protect its revenue stream. We will review some of the major fraud initiatives and the techniques employed to address them, leading to generic conclusions about fraud detection. Specifically, we advocate the use of simple, understandable models, heavy use of visualization, a flexible environment, the importance of data management, and the need to keep humans in the loop.

1 Introduction

> *"Food gained by fraud tastes sweet to a man, but he ends up with a mouth full of gravel."*
>
> Proverbs 20:17 (New International Version)

Fraud, the act of deceiving others for personal gain, is certainly as old as civilization itself. The word comes from the Latin *fraudem*, meaning deceit or injury, and over the years has come to represent a wide array of injustices, including forged artwork, confidence schemes, academic plagiarism, and email advance-fee frauds (such as the well known Nigerian email scam). Although these forms of fraud are very different in nature they all have in common a dishonest attempt to convince an innocent party that a legitimate transaction is occurring when in fact it is not. Usually the fraud is for monetary gain, but not always, as fraud may be perpetrated for political causes (electoral fraud), or personal prestige (plagiarism), or self-preservation (perjury).

In the 20th century, fraud matured in the area of transactional businesses, most notably in the telecommunications and credit card industries. Due to the sheer volume of transactions in these businesses, fraud could go unnoticed fairly easily since it was such a small proportion of the overall business. In the early days of the telecommunications business, "social engineering" was used to convince operators to give access to lines or complete calls that were not authorized. In the 1950's AT&T started automating direct-dial long distance calling, exposing themselves to the first generation of hackers. Since fraud could now be done without speaking to a human, it could now be automated, and a new business was born.

In the early days of fraud, perpetrators (hereafter referred to as *fraudsters*) were able to inflict significant losses on telecommunications companies, causing hundreds of millions of dollars in uncollectable revenue. As time went on, these companies tried many techniques to combat the fraudsters, sometimes with some success. Hence began an 'arms race' that has continued to this day, where the legitimate businesses and the fraudsters aim to stay one step ahead of each other.

AT&T has played a significant role in fraud detection throughout the years. As a telephone monopoly, and one of the first companies to employ technology automating communications, AT&T was a prime target

*Authors are all in the Statistics Research Department at AT&T Labs-Research, Florham Park, New Jersey

of the technologically curious. A new subculture of telecommunications fraudsters (known as *phreakers*) was born when Joe Engressia, a famous early phreaker, stumbled into the world of fraud in 1957. He realized that by whistling at certain frequencies, he could control the trunks that automated call routing (**?**). In the following years, he perfected the art of breaking into AT&T's phone system. Joe is quoted as saying:

> I want to work for Ma Bell. I don't hate Ma Bell the way some phone phreaks do. I don't want to screw Ma Bell. With me it's the pleasure of pure knowledge. There's something beautiful about the system when you know it intimately the way I do.

Although Engressia's interest was purely academic, others used similar knowledge to create devices to hack into the phone systems and provide free calls to users at AT&T's expense.

In the decades that followed communications got more complicated, and so did the fraud schemes that targeted them. Large company telecom systems (called "Private Branch Exchanges" or PBX for short), cellular communications, and virtual private networks are a few examples of some of the new technologies that telecommunications companies have had to protect from fraudsters. Today, AT&T continues to be a main target of fraudsters. Although the incremental cost of carrying a single cost has become quite small in today's large networks, there is significant money lost through foreign settlement costs and access charges between companies. [RAB/ALLAN: Can we find a reference for the overall amount of telecom fraud, at T, in the US or in the world? If we can] Other industries, most notably the credit card industry, have also been challenged with fighting sophisticated fraud schemes, with varying levels of success.

One characteristic of fighting fraud, which reaches across most transaction-based industries, is the massive scale of the problem. "Massive" is a hard phrase to define and is clearly relative to the analyst's resources and the era. In the case of current telecommunications and credit card fraud, there are billions of transactions per day plus associated metadata. It is clear that the data storage, compression, and management have to be a well designed part of any fraud detection scheme.

With the rise of the Internet and Internet-based businesses in the 1990's and early 2000's, new types of fraud began to emerge. More transactions and purchases began to take place online and new companies such as eBay and Amazon.com had to concern themselves with fraud detection. The Internet era has spawned new types of fraud, including click-fraud, email spam, phishing schemes, and denial of service attacks. In addition, the Internet gives access to data about individuals that was previously much more difficult to collect and consolidate, resulting in the serious problem of identity theft.

In recent years, fraud modelling and detection has become an academic pursuit; some attempts to summarize the state of the art include **?**) and **?**). In this paper, we review some of the strategies AT&T has employed to fight fraud. Although fraud cuts across many industries and comes in many shapes and forms, we believe some of the lessons learned in the decades of experience at AT&T are relevant - and that there are some common philosophies that apply across many different kinds of fraud. Some of these philosophies are specific to problems where there are massive data sets, and therefore quite relevant to new types of fraud emerging in the 21st century.

The structure of the paper is as follows: Section **??** discusses the nature of fraud. Section **??** presents detail on how we have gone about fighting fraud over the years. Section **??** goes over issues in implementation of fraud systems. In Section **??**, we summarize the lessons learned at AT&T and how these might apply to the fraud detection problems of today and tomorrow.

2 The Nature of Fraud

The most difficult part in fighting fraud is identifying it. In the context of telecommunications, a fraudulent phone call is one where there is no intent to pay – theft of service. When fraudulent phone usage is found, the normal response is to shut down whatever avenue the fraudster found, although in egregious cases law enforcement may be involved.

Usage management is a business function closely related to fraud detection, dealing with accounts that are likely to be uncollectable. There is a fine line between intent to pay and ability to pay. Thus, it is often reasonable to have the same detection tools look for both fraud and usage management problems. In fact, often fraud will masquerade as a usage management problem; rather than seeking clever ways to subvert the telecom networks, the fraudster will simply subscribe to legitimate service, but with no intention to pay.

Historically, there have been many different types of fraud, ranging from teenagers hacking into systems from their bedroom to sophisticated organized crime rings (**?**). Our goal was to create a fraud management system that was powerful enough to handle the many different types of fraud that we encountered, and flexible enough to potentially apply to things we had not seen yet. Here are examples of common varieties of fraud in the telecommunications world.

1. **Subscription Fraud.** Subscription fraud happens when someone signs up for service (a new phone, extra lines, etc) with no intent to pay. In this case, all calls associated with the given fraudulent line are fraudulent, but are consistent with the profile of the user.

2. **Intrusion Fraud.** This occurs when an existing, otherwise legitimate account, typically a business, is compromised in some way by an intruder, who subsequently makes or sells calls on this account. In contrast to subscription calls, the legitimate calls may be interspersed with fraudulent calls, calling for an anomaly detection algorithm. Another example of intrusion-type fraud is online auction fraud. This can occur when a new user acts as a legitimate user for a period of time, with low-value transactions, in order to build up a respectable user rating and gain the trust of other users. Once the rating reaches a certain level, the fraudster lists an expensive item and either defaults on payment or collects payment without sending goods. In this case, anomaly detection must be fast, since it may be a single outlier that comprises the fraud.

3. **Fraud based on loopholes in technology.** Consider voicemail systems as an example. Voice mail can be configured in such a way that calls can be made out of the voice mail system (to return a call after listening to a message, for example), as a convenience for the user. However, if inadequate passwords are used to secure the mailboxes, it creates a vulnerability. The fraudster looks for a way into a corporate voice mail system, compromises a mailbox (perhaps by guessing a weak password), and then uses the system to make outgoing calls. Legally, the owner of the voice mail system is liable for the fraudulent calls; after all, it is the owner that sets the security policy for the voice mail system.

4. **Social engineering.** Instead of exploiting technological loopholes, social engineering exploits human interaction with the system. In this case the fraudster pretends to be someone they are not, such as the account holder, or a phone repair person, in order to access a customer's account. Recently, this technique has been used by "pretexters" in some high-profile cases of accessing phone records to spy on fellow board members and reporters (**?**).

5. **Fraud based on new technology.** New technology such as Voice over IP (VoIP) enables international telephony at very low cost, and allows users to carry their US-based phone number to other countries. Fraudsters realized that they could purchase the service at a low price, and re-sell it illegally at a higher price to consumers who were unaware of the new service, unable to get it themselves, or technologically unsophisticated. Detecting this required monitoring and correlating telephony usage, IP traffic and ordering systems.

   Click-fraud has recently caught on as a new type of fraud targeting online advertisers. New services like Google's AdWords, where companies pay for online advertising every time a user clicks on an ad, have quickly grown in popularity. Now there is an incentive for adversaries to click on these ads, perhaps using a script, to create large advertising bills for a competitor.

6. **Fraud based on new legislation.** Occasionally, legislative acts intended to promote fairness end up spawning new types of fraud. In 1996, the FCC modified payphone compensation rules requiring payphone operators to be compensated by the telecommunication providers. This allowed these operators to help cover the cost of providing access to phone lines such as toll-free numbers, which do not generate revenue for the payphone operator. This spawned a new type of fraud – payphone owners or their associates placing spurious calls from payphones to toll-free numbers simply to bring in compensation income from the carriers. This scenario has many similarities to click-fraud in that the compensation model is the motivation for fraudulent behavior.

7. **Masquerading as another user.** Credit card numbers can be stolen by various means ("shoulder surfing" for example – looking over someone's shoulder at a bank of payphones, say) and used to place calls masquerading as the cardholder.

There are many more fraud techniques, some of them quite sophisticated, which can combine more than one known method. Telecommunications fraud is not static – new techniques evolve as the telecom companies put up defenses against existing ones. The fraudsters are smart opponents, continually looking for exploitable weaknesses in the telecom infrastructure. Part of their motivation is accounted for by the fact that once an exploit is defined, there are thousands (or millions) of potential targets. New types of fraud appear regularly, and these schemes evolve and adapt to attempts to stop them.

3 Detecting Fraud

A fraud detection system must be flexible to respond quickly and effectively to a variety of fraud types. As new services are offered or telecom regulations change, new exploits may appear. As an example of the adaptibility of fraudsters, consider their ability to infer the levels of detection thresholds. Many early versions of fraud detection software involved thresholds. In a simple example, more than n calls averaging k minutes to a particular country in an hour might trip an alert, leading to an investigation. Fraudsters may deal with this sort of situation by trying to keep their calls short, at least in the initial part of an exploit, to lessen the chances of an investigation. Once they have gone undetected for a while, the fraudsters may feel comfortable raising their own average call length. They may even be able to infer the carrier's threshold by observing when their activity is distrupted as the carrier detects and blocks them.

There is one easy rule to follow in detecting new fraud: "Follow the money". When bank robber Willie Sutton was asked "Why do you rob banks?" he said "Because that's where the money is." Similarly, telecom fraud most often involves international calls, because those calls are the most costly. Although early phone phreakers often did their exploits for the "glory", much of current telecom fraud is involved with making money. If a fraudster can steal a phone call and then sell it to someone else, it's pure profit. This is also why there are various schemes associtated with audiotext numbers: high-cost, international chat lines or lines with adult content. They are expensive services that can generate lots of revenue quickly, and fraudsters can receive a cut of the proceeds from these expensive calls.

Fighting fraud is complicated by the existence of multiple telecom carriers. Since equal access was instituted in 1982 (**?**), hundreds of long-distance carriers have come on the scene and they can be used on a per-call basis by dialing a carrier prefix when placing a call. Thus, fraudsters can easily migrate from one carrier to another; just as paying customers have a choice of long-distance carriers, so do the fraudsters. That means that once a carrier detects and stops fraudulent usage on its network, that usage is likely to migrate onto another network. Notifying the PBX owner to secure the compromised system is often the only way to ensure that all of the leaks are plugged.

However, there is an interesting phenomenon associated with the differential ability of telecom companies to carry out this task. In fact, if one company is better than others, this will tend to drive fraudsters onto the other networks, where the fraudster's "cost of doing business" will be lower. In the fraud detection game; if one telecom company is better than the competition at detecting and stopping fraud, the fraud tends to move to the competitors.

Ideally, fraudulent calls are detected as soon as a call is made, but depending on the type of fraud, detection may require a sequence of several calls. In any case, the sooner the fraud is detected, the sooner corrective action can be taken. If a person is involved in the detection/correction phase of the process, it is important that the person be available quickly, too.

Interestingly, with any consumer-related transactional service, there is an excellent distributed fraud detection system: customers' bills. Every month, millions of customers look over their bills and complain if they see fraudulent calls there. Unfortunately, the one-month billing cycle may allow fraud to run rampant for a few weeks, which is unacceptable from the company's perspective. Also, with more consumers turning to online billing, there are more people who never look at their bill. However, it is a great safety net, identifying fraud that might have been overlooked by a fraud detection system.

Recently there has been an increased academic interest in modeling fraud and documenting the successes (**?**; **?**; **?**; **?**; **?**; **?**). Most of these attempts in the statistical literature focus on the core fraud detection algorithm. In our experience at AT&T, in the context of fraud detection for massive data streams, there is no way to effectively separate the data analysis from the larger issues of data delivery, data management and data storage. In this section we will describe the framework that has developed over the years and how

the data analysis fits into this broader scheme.

There are several key components to a fraud detection system:

- a continuing source of call detail data
- a database to store the data
- a set of detection algorithms
- people to verify fraud and implement corrective measures
- visualization tools to help people make diagnoses

We will describe our experience in each of these areas.

3.1 Data Sources

The data that we analyze in our fraud detection system consists of telecommunication records, or more simply phone calls. A phone call can be well described by a small number of variables: originating and terminating parties, time stamp and duration of the call, and perhaps some other flags denoting the type of call it was (toll-free, operator assisted, etc). However, even for something as simple as a phone call, the scale of the network and the legacy of systems combine to create a large collection of obscure and complex data sources.

[RAB/ALLAN: I added the above paragraph for context - can you rework this section as we discussed, with more specific descriptions of the different data sources which come into play in call detail, along with some orders of mag of size? AMA, SS7, lidb, metadata feeds, 500 pages of documentation, wireless, different data from different sources. point: dont know at the outset what the important variables are. Try and answer Ref 1 point 3.1. Another point I just thought of....statisticians need to understand the data to do their job - this data was so complex that it was near impossible to do the analysis without getting involved in the data management]

The most common source of data describing telephone calls is AMA (Automated Message Accounting) information, the call detail records used to create phone bills. Because it is designed for billing, AMA data is comprehensive (**?**) but may not be timely. In total, AT&T collects information on billions of local, long distance, cellular, and international calls each day, amounting to hundreds of gigabytes of data. Of course, for billing purposes, daily receipt of AMA data is sufficient. Originally AMA was sent from the network switches to the billing systems on magnetic tape, but is now transmitted electronically and can often be sent in real time. Note that even with real-time AMA data, there is still the drawback that the detail for a call is only available after the call is complete.

Data from cellular calls is also available, although typically not in a standardized form like AMA. Instead, the equipment vendors for cellular switches typically have their own proprietary call detail formats.

Another source of data is the SS7 signalling data that is used to set up calls (**?**; **?**). The Common Channel Signalling network uses the SS7 protocol. It is independent of the voice network and it was designed, at least in part, to prevent fraud by keeping the signalling away from the voice channel. Data collected on the signalling network is similar in volume to the AMA data and has the same advantage of real-time collection; it has the additional advantage that it can be sampled at call set-up time, avoiding the delay until call completion inherent in AMA.

Another data source comes from the Line Information Database (LIDB) that gives the status of other calls, such as bill-to-third, collect, and calling card calls. (**?**)

3.2 Storage

In a modern computing environment, it makes economic sense to store all of the call detail data for an extended period of time. Storing the call detail in a separate database simplifies the fraud detection system; when call detail is needed, it is a simple matter to query it from the database.

Database technology is well established, though the volume of data and need for real-time loading make selection of the database management system important. In our application, the database is managed by Daytona (**?**).

[RAB/ALLAN - re-work this paragraph to talk more about what are the desired properties of a DBMS: real-time updating, seconds from call to data, storage of vast amounts of historical data. minimal overhead. Daytona has these things and so we used it. Should sound less like an advertisement and more about the key features.]

Our strategy is to store all of the call detail for several years. In 2005, this database was recognized as the largest data warehouse, based on normalized data volume, in the Winter Corporation's survey of the world's largest databases (**?**). Because Daytona uses both field-level and row-level compression, the actual disk space for the database, including extensive indices, is smaller than the size of the raw data alone. The compression also speeds up retrieval time for queries, since fewer bytes are accessed from disk. A typical query that would retrieve the most recent 2000 international calls for a particular phone number would take less than a minute.

As the call detail arrives, each record is augmented with supporting information, e.g. the estimated retail cost of the call, the type of service (direct dial, collect, bill to third party) and whether the calling number is a business or a residence. In addition, it is important to ensure the quality of the data in the database by making as many consistency checks as possible.

Once the call detail database is established, it fills many needs beyond providing information for the fraud system. Having a database allows for retrospective analyses and provides the means of testing new algorithms on historical data. It also provides the historical view of individual calling patterns, so that deviations from the normal can be seen.

3.3 Detection

3.3.1 The need for personalized thresholds

A fraud detection system processes a large volume of data so that detection is a needle-in-a-haystack problem – the system must automatically filter the data looking for unusual usage patterns, and then algorithms or people can look at the unusual parts to zero in on the potential fraud.

Early fraud detection algorithms often used thresholds, delivering alerts when the number of messages and the number of minutes of calling within a specific time period exceeded a preset threshold. A drawback with this method is that if the time period is, say, an hour the fraud has, on average, 30 minutes without alerting. Another, more important drawback is that it treats customers identically; if a threshold is set so that it finds fraud for a small business line, a large business is likely to trip the threshold routinely.

In order to better identify fraud, thresholds moved to finer granularity. Separate thresholds were provided for hour of the day, day of the week, and country called. Ultimately, one of the early AT&T fraud systems had about 30,000 threshold values because of this desire to fine-tune alerting, and this resulted in difficulties managing all the thresholds while still failing to treat customers individually.

Even with such a fine-grained set of thresholds, there is still a wide variety of usage from customer to customer, so that thresholds set tightly enough to catch fraud often catch lots of legitimate use, too. In addition, as more and more orthogonal thresholding variables are introduced, eventually there will be more thresholds than customers (the curse of dimensionality).

As this dawned on us, it seemed like a good idea to specialize thresholds to individual customers. We would keep a signature for each customer phone number, containing a rough idea of that phone's current calling characteristics. This is how several of us at AT&T Bell Laboratories got involved in fraud detection in 1995 – working to devise a customer-specific signature based alerting system. Many details of the specifics of signature-based systems appear elsewhere (**?**; **?**; **?**). We give a brief description here.

3.3.2 Signature-based alerting

The basic idea of signature-based alerting is that as each call comes in, it is compared to the current signature for that customer and also to a generic fraud signature. If the call looks more like the customer usage pattern than fraud, the characteristics of the call are used to update the signature according to an adaptive exponentially-weighted moving average. On the other hand, if the call looks substantially more like fraud than like the customer signature, a fraud score will be increased for the phone number. Once that per-customer fraud score passes a threshold, an alert is generated.

Notice that the signatures are self-updating – as new data is seen, the signature becomes more representative of the customer's current calling behavior, and is able to track evolutionary changes in the customer's behavior.

Signatures are based on various univariate measures:

- calling rate (calls/hour)
- distribution of calls by day of week
- distribution of calls by hour of day (for both weekday and weekend)
- distribution of call duration
- regions of the world called
- most frequent countries called
- most frequent phone numbers called
- whether calls are charged and the billing numbers used

All of this information is stored in about 500 bytes per customer and contains simple statistical summaries of the properties of that particular user, as shown in figure **??**.

Figure 1: Statistical profile for a typical legitimate customer

Each parameter in the signature X is updated by a version of a exponential weighted moving average:

$$X_n = \theta * D_c + (1 - \theta) * X_p \tag{1}$$

where X_p and X_n are the previous and new values of the parameter and D_c is the information that comes from the current call.

These types of EWMA models were introduced in (**?**) and have been widely used in statistical forecasting and statistical process management. The properties of EWMA models have been studied extensively, and are covered in standard textboks such as (**?**). EWMAs update parameters in a smoother fashion than simple moving averages, and allow for streaming updating of parameters without the need to access data that has already been seen.

There are two important parameters to set in the EWMA - the parameter θ and the updating interval. Both of these together help determine how quickly new information washes out old information. If θ is large, the decay curve is steep and new information quickly washes out old information. If the updating interval is small, as in call-based updating, θ might have to be adjusted so that signal from infrequent callers does not get washed out too fast. In applications with time based updating, it is often sensible to set θ globally for all signatures. As an example, consider an application with daily updating. With $\theta = 0.85$, the parameter value will effictively discount all data (by reaching 0.1% of its value) after 30 days, while $theta = 0.5$ will reach the same value in just over a week. What type of fraud we are trying to catch might inform which of these is more reasonable. The most time-sensitive applications tend to use event-based updating, where Equation 1 is applied call-by-call. Here we set a different θ for each signature as a non-linear function of the calling rate to ensure that profiles are not diluted for those profiles we heavy activity. To take care of this, we make θ smaller as the calling rate, r, measured in calls per week, increases. Basically, we multiply θ by $1/log_2(r)$ to ensure that old information is not washed out of the estimate too quickly. Setting a different θ for each signature adds a significant amount of complexity to the modelling. A middle ground between global parameter setting and individually varying parameters is to segment the data into relevant categories (e.g. residences and businesses) and fit the parameter separately for each segment.

Another way of setting θ was proposed by (**?**). They optimized θ by looking at the signature as a predictive function. The best value of the parameter is the one that best predicts future behavior of the phone number. With this perspective, *theta* can be learned by segmenting the data into training and test sets and using machine learning methods to minimize the predictive error.

An interesting special case is how to initialize the signature for a phone number. When there is no prior signature, data values from the first two calls are used to select an initial signature from a set of empirically built signatures. Also, old signatures that are not reinforced by new calls within a month or so are dropped completely, since after that much time without calls, the phone number could well have been reassigned.

As mentioned earlier, the overall fraudiness for a particular phone number may increase call by call. To complement this growth, the fraudiness numbers are decreased every day, leading old fraudiness values to decay away unless they are reinforced by new indications of fraud. In this way, the fraudiness score is an indicator of recent fraud activity.

Signature-based alerting tends to do a good job of finding changes in calling patterns that indicate fraud. However, no one detection algorithm can find all types of fraud. In our system, we use many detection algorithms, each of them tending to be simple, rather than a single complex algorithm. There is power in many simple models as opposed to one complex one. Various detection algorithms may generate an alert on the same phone number, and these alerts are combined into a single case to be investigated. The case manager is a component of the system that can decide which cases should be investigated first, based on type of fraud and likelihood of fraud (as opposed to a false alarm).

Signatures extend naturally to help solve other problems in the telecom business. For example, a common question asked about a phone number is whether it belongs to a business or residence. Various characteristics of each call tend to mark its likelihood as a business or residence (hour of day, duration, etc). By keeping a signature that records how business-like a phone number is, we can determine empirically how each phone is behaving.

3.3.3 Moving to Graph-Based signatures

Statistics-based signatures such as those shown in Figure **??** proved to be very useful in the type of anomaly detection-based alerting that was needed for catching many types of fraud. However, statistical summaries do not fully capture the directional network of who called whom. We use $\mathcal{G}_t$ to denote the *callgraph network*, where telephone numbers are nodes and directed edges represent communication between those numbers. The size of this graph is on the order of hundreds of millions of nodes and billions of edges.

In order to better cope with the scale, we developed a framework called the *Community Of Interest (COI) signature* for each telephone number (**?**). This signature includes the top numbers (top-k) called by the target number and the top numbers that call it. This way we were able to look at the massive graph at a local scale, where each phone number has its own small graph, and the union of these graphs make up the bulk of $\mathcal{G}_t$.

This signature had the additional challenge of dealing with the fact that phone numbers are transient - many new phone numbers appear regularly as new lines are provisioned, and many old ones disappear as people move or transition to other services. A study in 2003 (**?**) showed that about 1% of all phone numbers disappear in a given week, with a similar amount of new numbers appearing. Over the course of several months there is the potential for a large turnover of the phone number space, which we account for in the graph signatures by a variant of the exponential smoothing in Equation **??**. Let $\hat{\mathcal{G}}_{t-1}$ denote the top-k approximation to $\mathcal{G}_{t-1}$ at time $t-1$ and let g_t denote the graph derived from the new transactions at time step t. The approximation to $\mathcal{G}_t$ is formed from $\hat{\mathcal{G}}_{t-1}$ and g_t, node by node, using a top-k approximation to Eq. **??**:

$$\hat{\mathcal{G}}_t = \text{top-}k\{\theta\hat{\mathcal{G}}_{t-1} \oplus (1-\theta)g_t\} \tag{2}$$

where the $\oplus$ operator is a graph sum operation which takes the union of the nodes and edges in the two graphs for the aggregate graph. The "top-k" part of the updating is a pruning function which only includes the neighbor nodes with the highest weight in the COI signature. Everything that is not included in the top-k edges gets aggregated into an overflow bin called *other* (See Figure **??**). Truncation of the signature in this fashion ensures that only the most relevant nodes will make it into the signature.

Once we have COI signatures for each phone number, automatically updated through the exponential weighting process, they serve as a surrogate for doing analysis on the entire graph. Conceptually, we have now taken our communications graph $\mathcal{G}_t$, which is of the order of hundreds of millions of nodes and billions of edges, and replaced it with a database of hundreds of millions of graphs, where each graph is stored in an indexed database and is easily retrieved. As such, we can extract more sophisticated graphs by applying

$$(1-\theta)\begin{array}{c}\text{Old top-}k\text{ edges}\\ \begin{array}{ll}\text{node–labels} & \text{wts}\end{array}\\ \left(\begin{array}{ll} 652\text{-}5467 & 5.2\\ 756\text{-}2656 & 5.0\\ 652\text{-}4132 & 4.5\\ 653\text{-}4231 & 2.3\\ 624\text{-}3142 & 1.9\\ 735\text{-}4212 & 1.8\\ 423\text{-}1423 & 0.8\\ 534\text{-}2312 & 0.5\\ 526\text{-}4532 & 0.2\\ \text{Other} & 0.1\end{array}\right)\end{array} + \theta \begin{array}{c}\text{Today's edges}\\ \begin{array}{ll}\text{node–labels} & \text{wts}\end{array}\\ \left(\begin{array}{ll} 543\text{-}6547 & 10.0\\ 756\text{-}2656 & 6.2\\ 652\text{-}5467 & 2.0\\ 652\text{-}4132 & 0.8\\ & \\ & \\ & \\ & \\ & \\ & \\ \text{Other} & 0.0\end{array}\right)\end{array} = \begin{array}{c}\text{New top-}k\text{ edges}\\ \begin{array}{ll}\text{node–labels} & \text{wts}\end{array}\\ \left(\begin{array}{ll} 756\text{-}2656 & 5.2\\ 652\text{-}5467 & 4.6\\ 652\text{-}4132 & 3.9\\ 653\text{-}4231 & 2.0\\ 624\text{-}3142 & 1.6\\ 735\text{-}4212 & 1.5\\ 543\text{-}6547 & 1.5\\ 423\text{-}1423 & 0.7\\ 534\text{-}2312 & 0.4\\ \text{Other} & 0.3\end{array}\right)\end{array}$$

Figure 2: *Computing a new top-k edge set from the old top-k edge set and today's edges. Note how a new edge enters the top-k edge set, forcing an old edge to be added to Other.*

this process recursively, and including nodes in the extended COI signature that are two, three or more hops away from the target node.

These graph-based signatures have proven to be extremely useful in detecting different types of fraud. In one example, owners of adult-content chat lines attempted to set up fraudulent toll-free phone numbers which terminated at their chat line. Typically, these customers would pretend to be a legitimate business such as a flower shop or candy store, perhaps guaranteeing the account with a stolen credit card. The service would be activated and the number disseminated to the consumers of the service, who would call into the service for free. After a while, the bill goes unpaid and the number is shut down, but since we have phony information, there is no one to prosecute, leaving the fraudy business owner free to start over again with a new chat line.

Although catching the bad guys was difficult, monitoring the COI signatures of the users of the service led us to the next fraudulent line set up. These customers were not necessarily fraudulent themselves, but they belonged to a group of people who used the services that were fraudulent. In addition, the fraudsters themselves tended to communicate, or at least they belonged to an affinity group of people who tried to scam the phone companies. Based on this, we established a "guilt by association" module which was quite successful in catching this type of fraud. Figure **??** shows a typical case (labelled "Suspect") whose guilt-by-association plot shows several known fraudulent accounts. This would increase the fraud score of the suspect, perhaps enough to put her on a fraud investigation list.

Figure 3: *A guilt by association plot for the node labeled "Suspect". Circular nodes correspond to toll-free accounts while rectangular nodes are conventional accounts. Shaded nodes have been labeled as fraudulent by network security associates.*

3.3.4 Catching Fraud via Graph Matching

The COI signatures serve as a communications fingerprint, a way to characterize the usage not of a phone number, but of the person behind the phone number. As such, it has been useful in tracking down miscreants who try to cover their tracks by changing their phone number, their name, or their address. One example of this was our *Repetitive Debtors Database* (RDD). RDD was designed to keep a running database of COI signatures of delinquent customers in an attempt to track them down if they tried to assume another identity. Consider the case where we disconnect a customer due to nonpayment. In some cases this customer might try to sign up for a new account under another identity, either by using a different name, a different billing address, or a stolen credit card. Sometimes just permuting letters in the name is enough to avoid detection by the sales representatives. However, the new account, even with different identifying information, belongs to an individual, who is likely to have the same communications fingerprint as the delinquent account. Our

intuition is that if we build COI signatures on all new accounts, and match them to the signatures of the delinquent accounts, we can catch these fraudsters.

This approach entails building a distance metric between graphs, namely the COI signatures that we have built. The distance function is primarily based on the overlap between the two graphs (phone numbers that appear in both signatures) and the proportion of the overall communication accounted for by those overlapping nodes. In this way, we account for both the quality and the quantity of the overlap. **?**) propose two metrics for graph distance, one using the Dice criterion, a well-known metric from information retrieval, and one using the Hellinger Distance, a distance function on probability measures. Using these metrics allows one to characterize a large part of the communication graph for an individual using as concise a representation as possible. This was implemented in the RDD application and drastically improved the amount of fraud detected.

3.4 The Role of Humans

We have witnessed several attempts over the years to build systems with intelligent "agents" which can learn the actions that a human would take in a given situation and apply them. These agents are usually proposed as part of a system which attempts to take the humans "out of the loop" and perhaps introduce some cost savings. It could be argued that detecting fraud and stopping it would be best done completely by computer controlled algorithms. However, in many cases, it is difficult to diagnose fraud correctly without a person investigating. The same calling pattern could be fraud for one customer and normal usage for another. Humans are sensitive and creative in assessing calling behavior – they can often tell when a change in calling pattern makes sense and when it doesn't. In our experience, this intuition for different scenarios is extremely difficult to be codified.

Another good reason for having people in the loop is that it means that the fraud detection software doesn't have to be perfect (and thus difficult and complex to produce). Instead, the detection algorithms can provide alerts that function as clues rather than fully-weighed conclusions. One way to think of this is that the algorithms are pointing to potential crime scenes and the person is an investigator who looks for additional clues. The false alarm rate for alerts doesn't have to be zero as it would be if algorithms alone were going to be used to shut down a business customer's telecommunications because of suspected fraud. (Usually only a human can strike the balance between letting fraud continue and potentially shutting down an essential business tool.) Personal verification (or disproof) of fraud gives a solid answer for each case, and those definitive answers can be used to help tune the detection algorithms for better performance.

The task of working a case involves many user-controlled actions. Cases are selected by analysts based on their areas of expertise: business, residential, credit card, etc. Initially, the analyst retrieves the call detail related to the key number and looks at it, either as a table for small sets of call detail or graphically (for larger data sets). The analyst can see if there are suspicious patterns in the call detail (either with or without looking at the alerts generated by the system). If fraud is suspected, the analyst deals with it appropriately. For example, in the case of a business customer where fraud involves a PBX or voice mail system, the analyst will contact the customer and advise them on securing their equipment and offering to block traffic until the system is secure. Through the entire process the analyst documents what has been done, (call customer, leave voice mail, receive callback, implement a block) by interacting with the case manager software. Note that the case manager, which contains data on all events involved in that case (analyst notes, all alerts, etc.), now becomes another source of data that must be stored appropriately to be accessed for future cases.

We find that the experienced case worker uses a significant amount of intuition in working a case. For instance, the worker can look at a case, and understand that there is a trade off between the amount of money that can be recovered by working a case, the amount of effort that they will have to use to investigate it, and the probability that this case is a false positive. We decided not to attempt to code this information into a single fraud "importance" score, but rather to provide the analyst as much of the relevant information as possible, with a powerful case manager that allows them to sort and filter on relevant variables. One of the biggest examples of this point is visualization. Visualizing the data associated with a given case is not always appropriate or efficient (for instance, if there are a very small number of relevant calls associated with an alert). Several factors go into whether or not a sophisticated interactive visualization tool is the best way to investigate the data, or whether a simple table might suffice. We do not mandate visualization of every case, instead we provide the analyst with all tools necessary to determine whether it is warranted in a given

case.

3.5 Visualization

In many instances, the data associated with a given case is large enough or contains enough variables that a visualization tool can be of great help. These tools enable them to view and understand perhaps thousands of calls. The goal is to display all the recent call detail, not just the part that looks "fraudy", to allow the user to see the potentially fraudy calls in the context of the normal calling pattern. For this we provide a tool based on the Yoix language (**?**) , which itself is built atop Java. The tool provides a plot with a time axis to show each call and also provides interactive histograms of various call characteristics that allow the user to display interesting subsets of the data.

In the plots that follow, the horizontal axis is time and the vertical axis is call duration. Each call is shown as a vertical spike. In these displays spike color is determined by country called. Axes below the main display show tick marks that indicate when calls overlap one another (a ptential sign of compromised use), when terminating phone numbers are on a list of known high-fraud numbers, and when the operator is involved. Shading (if present) shows normal business hours in black, out-of-hours in gray.

Here are some examples: Figure **??** shows a year of call detail for seven phone numbers associated with a single customer. Even though there are 655 calls shown, the two heavy spikes stand out, emphasizing two periods when fraudulent calling occurred.

[RAB: Add a paragraph somewhere that gives a 'walk-through' of how one might interact with SeeCalls]

Figure 4: *Shows calls from a set of 7 phone numbers associated with a single customer over a 1-year period. Two sets of spikes are quite high, associated with two fraud incidents in the past year.*

Figure **??** shows an egregious fraud event, where the normal calling pattern is overlaid by many calls to a foreign country which is historically associated with fraud. Even though there are 7000 calls displayed here, it's easy to see the fraudulent calls, particularly since many of them are of the same duration.

Figure 5: *Blocks of yellow show calls to a specific foreign country happening out of hours, starting Jan 15 and continuing to the 28th. The display shows 7000 calls. Notice the uniform height of calls at 45 minutes.*

Another instance of fraud to this same country is shown in figure **??**. In this case, the fraudulent calls happen out of normal business hours (which are shown by the shaded rectangles), contrasting with the normal calling pattern.

Figure 6: *A two-day set of fraudulent calls to a specific foreign country (yellow bars). Note how these calls are primarily out of business hours and the general pattern for this phone is to make calls only during business hours.*

The next example, figure **??**, shows how intense fraudulent activity can contrast with infrequent normal behavior. The fraud pattern here is particularly easy to distinguish.

Finally, figure **??**, shows a single social-engineering event superimposed over a year of regular calling. These calls are often very long duration.

4 Implementation

In 1998, AT&T implemented a new fraud detection system called GFMS (Global Fraud Management System), built mostly by statisticians. How did a group of statisicians get involved in writing a production

Figure 7: *One year of calling behavior for a single phone line. The pattern of intense activity at the center is fraud, contrasting with the infrequent call pattern the rest of the year.*

Figure 8: *One year of calling shows a spike in the middle due to social engineering, with long duration operator-assisted calls to Yemen.*

fraud detection system? It wasn't our plan. The initial concept was that we would provide a signature-based alerting algorithm to plug into a new fraud system that was to be built by other organizations within AT&T. However, the new system never got built through standard development channels and we were left with a stand-alone detection algorithm. At that time, it became apparent that we would have to provide the entire fraud system in order to deliver the signature alerting. This was also under a strict deadline because of the Y2K effort within the company – the older fraud detection system was not Y2K compliant.

We succeeded because we had a small team and used simple components, many of which we had developed in order to carry out full-volume testing of the signature alerting. In particular, we had already constructed the call detail database and various tools to help with the huge flow of data. We also used extreme programming teams (**?**) to produce code quickly and effectively.

[ALLAN : Include more detail about the piece parts, Streamer, Hose, Vetter, etc. One of the refs asked for "conceptual design" - perhaps discuss here about the Unix pipe model of taking small things and joining them together].

One of the most important tools used in our fraud system is the "streamer" – software that reads the call detail record by record, and for each record passes it though a set of plugins. Each plugin does three things: initialization (once per file of records), processing (once per record), and wrapup (once per file). The plugins are independent of one another; each is designed to look for a particular kind of call detail record and extract the relevant information from it. The initialization routine of each plugin often opens a file to hold the extracted information. The processing function makes a very fast decision on each record, rejecting it very quickly if it is not of interest. Otherwise it extracts a few fields from the call detail record and writes them to the file opened by the initialization function. Finally, the wrapup function closes any open files (and removes files that were never written to).

The files extracted from the call stream by each plugin are typically processed by a job scheduled to run at regular time intervals. Often, these jobs will apply thresholds of various sorts, producing alerts when appropriate. Each detection algorithm is comprised of a plugin and the code that processes the extracted data to generate alerts. Plugins can themselves produce alerts, if they detect from a single call that an alert is justified; an example is an alert based on a call of very long duration.

As new fraud patterns arise, detection algorithms are generally implemented by adding a new plugin and some associated logic, allowing a quick response to the changing face of fraud. We have been able to add new plugins, fully tested, in as little as 45 minutes.

Our system gets validated in a few different ways. The cases that we detect are handed off to an internal fraud investigation department that makes a final determination as to whether to take action on the particular case. We have access to the results of these investigations, and allow for regular assessment of our detection algorithms. Because the system is modular, it is easy to adjust a model and reimplement it quickly if needed. Another method of validation comes from the fact that the impact of fraud often shows up on the bills of innocent customers. Therefore, we have millions of customers acting as fraud detectors when they inspect their bills carefully on a monthly basis. Reports from customers about incorrect charges often lead us to determine that some fraud has occurred. Keeping track of such reports allows us to get a high level sense of the amount of false negatives of our system, the undetected cases of fraud. These measures ensure that our models are effectively addressing the fraud problem through time.

5 Conclusions

In our journey to develop world class fraud management tools from massive data streams, we have attacked problems which on the surface appear to be quite different: subscription fraud, intrusion detection, repetitive debtors, access management fraud and the like all have their own characteristics that make detection hard. Nonetheless, over the years we have found some common themes that propagate across these different classes of fraud, specifically in the cases where we are dealing with extremely large data sets.

- **Need to join data analysis and data management.** In large data analysis projects, there is no way to decouple the analysis of the data from the storage, management, and retrieval of the data. In our early days, we thought we were only going to be building an algorithm, but at massive scales of data, even simple exploratory data analysis methods are challenging. Questions like: how much data is there? what are the distributions? are there outliers? become difficult. We discovered that having a stake in the data management allowed us to answer these questions more easily and also led to new possibilities and learning.

  In addition, after observing how the fraud team used our tools, it was clear that one of the most useful tools that they had was the ability to take high level summaries of the data, hone in on interesting patterns through interactive visualization, and access the raw call data underneath. Being able to access the full data records underlying the patterns could result in identification of important attributes in the data that were not previously considered important. Tying the data management and the analysis aspects of the system allowed for this close interplay between the fraud detection and the underlying data that would not be possible had the two systems been developed by different teams.

  [RAB/ALLAN: We talked a lot about this point, but I don't know if I captured it in the paragraph above. Please take a look.]

- **Inadequacy of large models.** In the early days, it took all of our effort simply to be able to keep up with the data coming in and attempt to do real-time signature updating. We were not able to fit any sophisticated models since the computational time and space were just too costly. As a result, our fraud system was built on simple models: simple distributions fit to profiles, likelihood ratio tests and basic regressions. As new types of fraud blossomed, the emphasis was on fighting it quickly, and simple fixes would go into the system. We ended up with a modular system where many small models came together to create a complex system. As the years went on, we found that many of the more sophisticated techniques we tried were not worth the extra time and complexity in terms of their performance against the collection of smaller models. In addition, some of our best performing models were simple ones based on hot-lists of known bad numbers.

  Many new data mining tools promise a one-size-fits-all approach to fraud detection, where a single tool can analyze any type of data, without much thought from the analyst. We think the better approach is to solve little parts of the problem, one at a time, and put them all together to make a single robust, adaptable system. This approach has worked well in the world of spam detection, where there is a similar "arms race" going on between the perpetrators and the enforcers (**?**). Most world-class spam detection systems are huge rule-based systems where every technique that a spammer uses is counteracted by a single small module which looks for that specific type of spam, and nothing else. Hundreds upon hundreds of these rules weigh in on every email, looking for signs of spam. When the spammers figure out a way around the current rules, the detectors simply write a new piece of logic looking for that specific loophole. The interesting statistical problem comes in combining the output of all of these detectors, but with the abundance of training data in the spam world it is easy to learn optimal models to reach acceptable values of false positives and negatives.

- **Necessity of humans.** Seemingly every day a new data analysis tool comes on the market claiming to do more "automated data mining". Our goal in building systems is not to take humans out of the loop, but to make the work of the humans easier. It is still true that the best pattern recognizer is the human brain. Most new fraud schemes are discovered by people who have broad domain knowledge and experience noticing that something is "just not right" with the data. The best that a fraud detection system can do is point the experts towards cases that might be fraudulent, but usually,

the investigation into the fraud requires a sophisticated sequence of deduction, analysis, integration between organizations, social interaction, and decision making that can only be done by people. Fraud detection systems should be built with an eye toward helping the experts, not replacing them.

- **Need for fast feedback loops.** The success of many of our tools depend on the ability to get quick feedback from a fraud management team that is investigating cases. For instance in the repetitive debtors example (Section **??**), we would provide the fraud team with a sorted list of cases which they would investigate in depth by calling the consumer and doing other background and data checks that only a human could do. At the end of the day, we would be given the results of these investigations, providing us with a labelled data set. In this way, our models could be improved incrementally and get continually better over time.

- **Importance of flexibility.** One of our design philosophies from an early stage was to build a powerful system from small components, each one doing a specific small job well. Small tools are built using simple scripts for data quality checking, data distribution, data storage, and analysis tasks. These small components are gathered together using a Unix-style "pipe", where components get plugged into the flow of the system. This philosophy allows for the most flexible system possible, as components can get updated, switched around, or removed with little impact on the whole system. From an analysis standpoint, the system is very modular. Once new types of fraud are identified, a statistical model is built to detect it, and it is easily plugged into the system and applied to the full data stream to identify new cases.

  Over the ten years that this system has been operational, the flexibility built into this system has allowed our fraud team to work nimbly to catch new types of fraud with new types of data that did not exist when the system was developed. New types of fraud are always popping up and any system designed to catch fraud has got to be lightweight and flexible enough to keep up with the arms race between the fraudsters and the fraud detectors. Of course, we don't know what types of fraud will be emerging in the coming years, and new challenges in data management and analysis will certainly present themselves. The lesson we learned is that a flexible system built with lightweight components will provide the best opportunity to adjust quickly to whatever comes our way.

We believe that our experience in building scalable, robust, and effective fraud detection modules has relevance in fighting the challenges that will undoubtedly arise with new technologies. The methods we have described can be applied to fraud schemes such as online auction fraud, click-stream fraud, and hopefully to other types of fraud emerging in the 21st century, and we hope that our observations might be helpful to those attempting to stop the fraudsters from succeeding.

References

Angus, I. and G. Blackwell (1993). *Phone Pirates*. Telemanagement Press.

Bolton, R. and D. Hand (2002). Statistical fraud detection: A review. *Statistical Science 17*(3), 235–255.

Cahill, M. H., D. Lambert, J. C. Pinheiro, and D. X. Sun (2002). *Detecting fraud in the real world*. Norwell, MA, USA: Kluwer Academic Publishers.

Cortes, C. and D. Pregibon (2001). Signature-based methods for data streams. *Data Min. Knowl. Discov. 5*(3), 167–182.

Cortes, C., D. Pregibon, and C. Volinsky (2001). Communities of interest. In *Proceedings of Intelligent Data Analysis 2001*.

Cortes, C., D. Pregibon, and C. Volinsky (2003). Computational methods for dynamic graphs. *Journal of Computational and Graphical Statistics 12*, 950–970.

Drechsler, R. L. and J. M. Mocenigo (2006). The Yoix scripting language: a different way of writing Java applications. *Software: Practice and Experience*.

Fawcett, T. and F. Provost (1997). Adaptive fraud detection. *Data Min. Knowl. Discov. 1*(3), 291–316.

Goodman, J., G. V. Cormack, and D. Heckerman (2007). Spam and the ongoing battle for the inbox. *Commun. ACM 50*(2), 24–33.

Greer, R. (1999). Daytona and the fourth-generation language Cymbal. In *Proceedings of the Twelfth international conference on Information and Knowledge management*, pp. 525–526. ACM Press.

Hill, S., D. Agarwal, R. Bell, and C. Volinsky (2006). Building an effective representation for dynamic networks. *Journal of Computational and Graphical Statistics 15*(3), 584–608.

International Telecommunication Union (1988). Recommendation Q.761, Signalling System no. 7. Technical report, ITU.

International Telecommunication Union (1993). Recommendation Q.764, Signalling System no.7 - ISDN user part signalling procedures. Technical report, ITU-T.

Kaplan, D. A. (2006, September). "intrigue in high places: To catch a leaker, Hewlett-Packard's chairwoman spied on the home-phone records of its board of directors.". *Newsweek*.

Kellogg, M. K., P. W. Huber, and J. Thorne (1999). *Federal Telecommunications Law* (2 ed.). Aspen.

Lambert, D., J. C. Pinheiro, and D. X. Sun (2001). Estimating millions of dynamic timing patterns in real time. *Journal of the American Statistical Association 96*, 316–330.

Moreau, Y., B. Preneel, P. Burge, J. Shawe-Taylor, C. Stoermann, and C. Cooke (1997). Novel techniques for fraud detection in mobile telecommunication networks. In *ACTS Mobile Summit*, Grenada, Spain.

Nosek, J. T. (1998). The case for collaborative programming. *Commun. ACM 41*(3), 105–108.

Phua, C., V. Lee, K. Smith, and R. Gayler (2005). A comprehensive survey of data mining-based fraud detection research. *Artificial Intelligence Review*.

Rosenbaum, R. (1971). Secrets of the little blue box. *Esquire*.

Rosset, S., U. Murad, E. Neumann, Y. Idan, and G. Pinkas (1999). Discovery of fraud rules for telecommunications–challenges and solutions. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, pp. 409–413. ACM Press.

Telcordia Technologies (2000). Line information database (LIDB) enhanced expanded measurement (EEM) generic requirements. *GR-3104-CORE, Issue 1*.

Telcordia Technologies (2005). Billing automatic message accounting format (BAF) generic requirements. *GR-1100-CORE Issue 10*.

Winter Corporation (2005). 2005 TopTen award winners. http://www.wintercorp.com/.