Paper SAS3536-2019

A Real-time Solution for Application Fraud Prevention

Prathaban Mookiah, Ian Holmes, John Watkins, and Tom O' Connell
SAS Institute Inc., Cary, NC

ABSTRACT

Application fraud in the Finance industry is a form of fraud that is committed by obtaining various financial products via misinformation and deceit, malicious acts such as identity theft, synthetic IDs, and account takeovers. In this paper, we present a real-time solution for application fraud prevention that was developed within the Fraud and Security Intelligence division at SAS®. This solution leverages some of our past successes in solving real-time fraud in other spaces such as payments as well as deposit monitoring. In addition, application fraud detection requires that we form a network view of the entities involved in the application to assess their risk. To this end, we have engineered a solution that combines well used techniques such as entity resolution, network and graph theory, dynamic Signatures, models based on machine learning to produce a score that represents the risk of an application, and the application of business rules for decisioning in real time. This paper describes the highlights of this solution.

INTRODUCTION

Application fraud in the Finance industry is a form of fraud that is committed by obtaining various financial products via misinformation and deceit, malicious acts (such as identity theft), and fictitious or synthetic IDs. Incidents related to identity fraud are hitting record highs worldwide, and this trend is not expected to slow down soon. This increase is driven by a plethora of factors including large-scale data breaches, growing proliferation of online shopping and related activities, increased sophistication of fraudsters, and a lack of general awareness among the public about how to protect their identities. In a world where there is an ever-increasing demand for instant credit decisions, financial institutions are looking to rely on real-time fraud mitigation solutions to detect and block fraudulent applications to shield themselves. Real-time processing of application events for fraud detection poses a unique set of challenges including sub-second response times, processing disparate sources of information all in one message, and forming entity networks.

In this paper, we will present a real-time solution for application fraud prevention developed within the Fraud and Security Intelligence division at SAS. This solution leverages some of our past successes in solving real-time fraud in other spaces (such as payment and deposits monitoring) using the SAS® Fraud Management solution. In addition, application fraud detection requires that we form a network view of entities involved in the application to assess their risk. To this end, we have engineered a solution based on SAS Fraud Management that combines various techniques such as entity resolution, network and graph theory, real-time profiling through Signatures, scoring models based on machine learning to represent the risk of an application, and the application of business rules for decisioning in real time.

NEED FOR A REAL-TIME APPLICATION FRAUD SOLUTION

The demand for real-time application fraud prevention is mainly driven from a customer experience standpoint. In the contemporary digital world, customers have come to expect instant decisions, especially on credit-related applications. Competition among businesses to satisfy the transient banking relationships of today often means that they need to capitalize upon the application requests quickly and efficiently. This has now reached the point where

credit facilities need to be available almost immediately, for example, at the point of sale for high ticket items through instant credit cards, car loans at the dealership, revolving credit provided by wireless carriers for newer mobile devices, and so on. However, risks around such unsecured products have been difficult to monitor in real time due to the data and analytic complexity involved in assessing the risk. This risk becomes more pronounced when requests are being made outside of a typical banking relationship where there might be little or no information about the applicant making the request.

From a risk point-of-view, poor data security, incessant data breaches, the increasing sophistication of fraudsters, and the proliferation of the dark web have led to record losses and exposure related to application fraud. When fraudsters understand the limitations of an institution's capability in preventing fraud, these fraudsters can easily exploit an institution's weaknesses when decisions are made without a capable real-time application fraud prevention system in place.

In addition, from a regulatory point of view, in many countries, regulations such as Fair Credit Reporting Act pose certain challenges around the handling of accounts after they have been approved and opened. Therefore, it is critical that applications are automatically screened in real time, so that they can be queued for additional layers of review and verification if necessary.

In more recent times, these factors have become increasingly prominent so a real-time application fraud solution has become a true necessity. It is in this backdrop that we designed the solution described in this paper.

INTRODUCTION TO SAS FRAUD MANAGEMENT

The solution described in this paper is built as an extension to the capabilities in SAS Fraud Management. First we will present a very brief introduction of this solution. Interested readers can find a more comprehensive overview of SAS Fraud Management at https://www.sas.com/en_us/software/fraud-management.html

SAS Fraud Management is an enterprise-level fraud solution that enables financial institutions to monitor multiple channels and lines of business in real time to protect themselves against potential fraud from a single platform. It is designed for users who need real-time decisions with very high throughput (> 10K transactions / events per second) and with low latency (< 50 ms). Typical channels with such requirements include payment cards (credit and debit cards) and various mobile and online payments including emerging mobile payments as well as non-monetary events such as authenticating users across digital channels.

Here are the vital components of SAS Fraud Management:

- a highly flexible orchestration system that can seamlessly integrate an expanding array of new data sources
- a modular, flexible, and extensible message layout to transport any type of event and transactions
- a proprietary profiling technology supported through multi-entity Signatures
- advanced machine learning capabilities including analytics and self-learning
- the On-Demand Decision Engine (ODE) which supports execution of up to four models on a single transaction, Champion – Challenger processing, and rules execution
- rule authoring and back testing using the estimation capability

- a comprehensive alert management system
- a transactional data repository (TDR) that stores all data flowing through the system

HIGH-LEVEL SOLUTION ARCHITECTURE

Figure 1 shows the high-level architecture of our solution. This is based on the core architecture for SAS Fraud Management and includes additional processes to support the application fraud use case.

In addition to the components of SAS Fraud Management, the following features have been added to support application fraud processing:

- a mechanism for entity resolution.
- a network generator for generating networks among connected entities.
- a feedback mechanism to provide information about known frauds back into the system. (This mechanism already exists in SAS Fraud Management and was enhanced for this solution.)

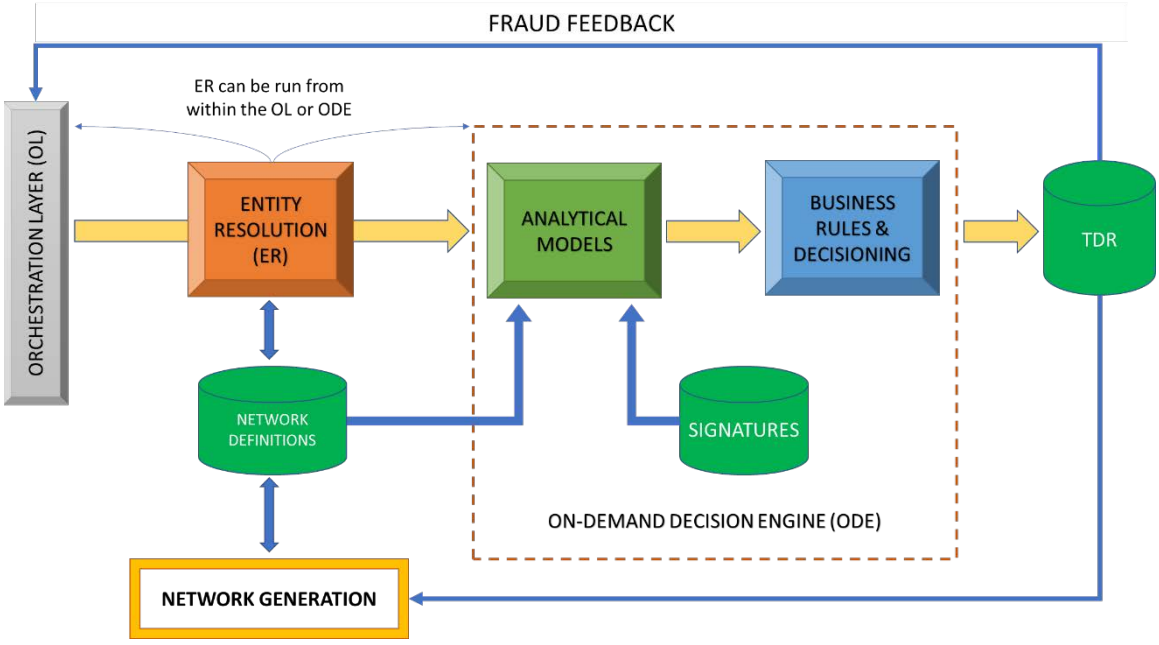

Figure 1. High-Level Solution Architecture

DATA ORCHESTRATION

Orchestration provides the opportunity to enrich new account information with contextual information about the applicant. In the modern threat environment, the correct external data can provide significant benefits. External data can include this type of information:

- channel-specific data through which the customer is applying, for example, device information and reputation
- biometrics and behavior, such as signals of unusual activity on the application page or the device
- public records that give a perspective of the customer's activities outside the bank and historical information around their identity

The addition of proper contextual data can provide a rich set of features critical for network generation, analytical models, and rules. It also enables the fraud managers to identify trusted customer behavior, provide differentiated service, and reduce friction.

Data orchestration within our solution is achieved by using SAS® Business Orchestration Services, which is a SAS Fraud Management component. SAS Business Orchestration Services provides the ability to enrich incoming application events with external data feeds in real time and translate that data into the SAS Fraud Management data structure. This service allows the system to enable multiple asynchronous API calls to a variety of third-party data providers as well as internal data stores. A metadata-driven API reduces the complexity of connecting new adapters, which reduces the speed to market for new data sources. SAS Business Orchestration Services also serves as an optional point from which processes such as entity resolution can be executed in real time and incoming transactions enriched with the results.

NETWORK GENERATION

Network analysis is a vital analytic component in solving application fraud. Network analysis provides a mechanism to holistically analyze communities that contain various correlated entities. It is particularly effective in solving application fraud due to some of the techniques used to perpetuate application fraud. These techniques include the use of compromised identities, synthetic IDs, and often the existence of coordinated fraud rings using common contact details.

A collection of disjointed networks can be formed by considering the linkages that exist between various entities such as persons, addresses, phone numbers, emails, and devices that are present among the applications. The different entities form the nodes of these networks. A direct link between two entities exists if the entities are present in the same application.

Generating networks is relatively a simple computational exercise. To illustrate, consider the five application events that occur on different days shown in Figure 2. The following entities are used to demonstrate our entity networks:

- person (disambiguated name)
- address
- phone number
- email address

App. No.	Date	Name	ID No.	Address	Phone Number	Email
1	1-Jan-19	Peppa Pig (N1)	11891	100 Sty Street, Pigton, PA 19100 (A1)	123-456-7890 (P1)	peppa@domain.net (E1)
2	1-Jan-19	Suzzy Sheep (N2)		8 Barn Drive, Sheeptown, SD 57105 (A2)	357-135-0865 (P2)	suzzy@domain.net (E2)
3	2-Jan-19	Rebecca Rabbit (N3)	10580	4A Warren Ct., Harewille, HI 96701 (A3)		rebr@domain.net (E3)
4	2-Jan-19	George Pig (N4)	23011	P.O. Box 1234, PA 19100 (A4)	111-111-1111 (P4)	george@domain.net (E4)
5	3-Jan-19	G. Pig (N4)		100 Sty Street, Pigton, PA 19100 (A1)	111-111-1111 (P4)	george@domain.net (E4)

Figure 2. Sample Applications

Application 1	Application 2	Application 3	Application 4	Application 5
N1 → A1	N2 → A2	N3 → A3	N4 → A4	N4 → A1
N1 → P1	N2 → P2	N3 → E3	N4 → P4	N4 → P4
N1 → E1	N2 → E2	A3 → E3	N4 → E4	N4 → E4
A1 → P1	A2 → P2		A4 → P4	A1 → P4
A1 → E1	A2 → E2		A4 → E4	A1 → E4
P1 → E1	P2 → E2		P4 → E4	P4 → E4

Figure 3: Links Between Entities

The links in Figure 3 are the direct links that exist between the various entities in these examples. Please note that the exact values have been replaced with a label for compactness. By considering these links, networks can be formed, which show the relationships that exist among the entities in these applications. Figure 4 shows the creation and evolution of these networks at the end of each day. It should be noted that in most cases networks used for application fraud are undirected, which means a direction is not associated with the links. In this example, we follow this convention.

After the networks are generated, various metrics can be computed at the network, link, and node levels. From an analytic perspective, such metrics can be directly used as features to the fraud model. Some examples of such metrics are listed in Table 1.

The network definitions (nodes and links) along with the various metrics associated with them are stored using a data structure that is optimized heavily for real-time lookups. The network definitions are retrieved in real time from this storage. The network generation process also extracts network definitions from the same storage and writes back to it.

The fraud feedback mechanism that exists in the solution (described in a later section) also provides this process with information about known fraud events and entities. This information is reflected within the network definitions.

The network generation process is designed to run at periodic intervals within the solution. This periodicity can be configured by the end user. The interval can be in the order of few minutes to several hours or days based on the business needs. It starts with extracting all new applications since the last run. By combining this data with the existing network definitions, the process updates existing networks based on the new application data. It also generates new networks that might have manifested in the recent applications. It should be noted that this process is fast and efficient and does not adversely impact real-time scoring performance. The updated network definitions are then made available immediately to the ODE.

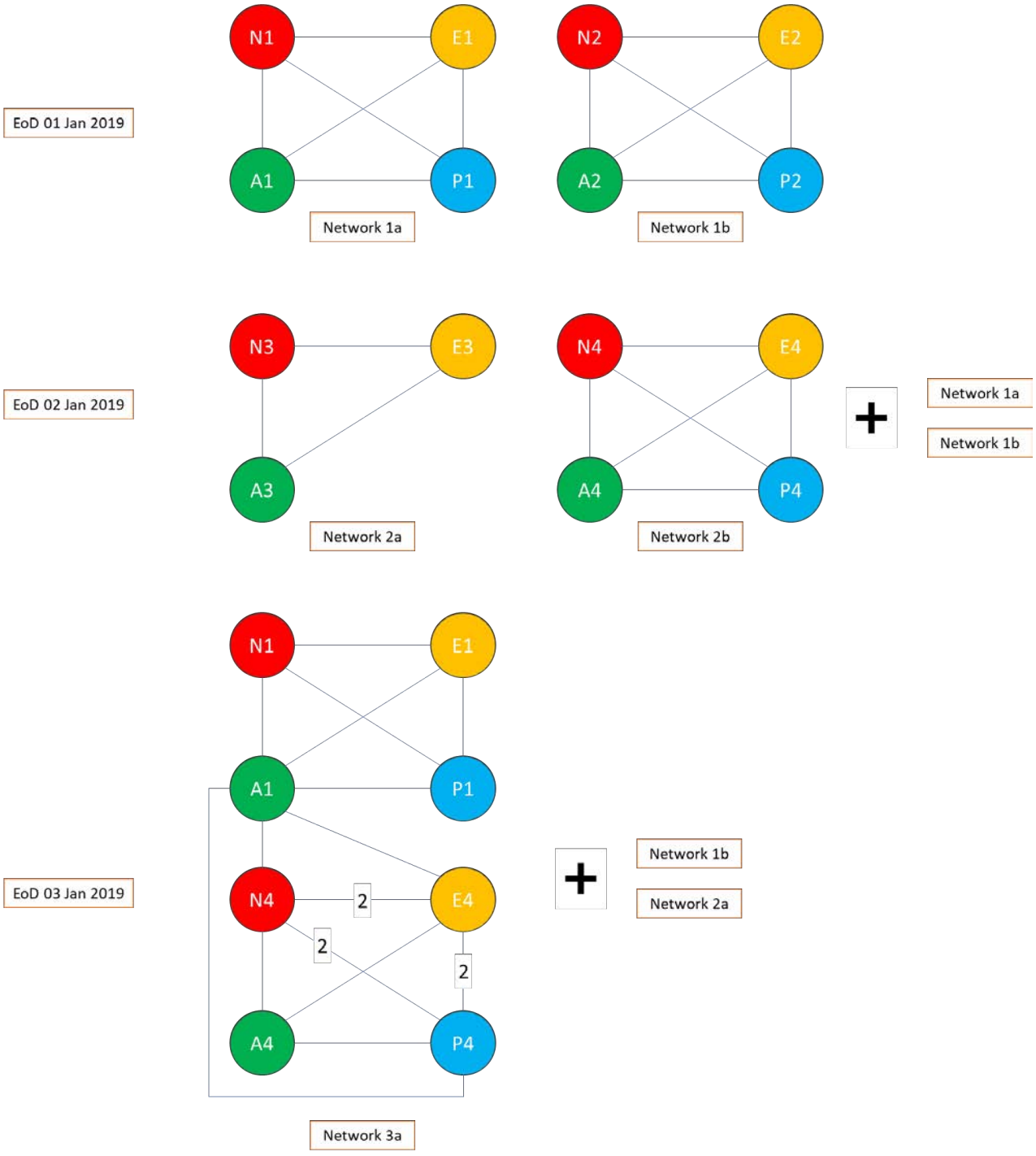

Figure 4: Entity Networks at the End of Each Day

Based on the applications received on day one, two distinct networks are observed. Similarly, two additional networks are observed for a total of four distinct networks after day two. Two of these networks collapse into one based on the fifth application, which results in three distinct networks after day three. Edge weights correspond to the number of unique applications in which the relationship between the two entities were observed. (The value is one when not indicated in Figure 4.)

SIGNATURES

Signatures are a SAS proprietary technology that provide a mechanism to convert entity behavior into inputs for the analytical model. It is a very flexible and extensible structure that the model developer can use to persist behavioral information about various entities. Signatures can be fetched and updated in real time during model processing. A model typically uses Signatures for multiple entities, and we use the same entities in our network definitions as well. Signatures and networks complement each other in a manner very suitable for the given problem. Signatures provide detailed information about individual entities whereas the networks provide information about the interconnection and relationships between the various entities. Together, they provide a holistic level of necessary information to assess an application's risk. Signatures can also be updated based on fraud feedback as deemed appropriate by the model developer.

ENTITY RESOLUTION

Entity resolution is the process of uniquely identifying entities when multiple and often ambiguous references are present for such entities. Entity resolution in application fraud primarily deals with entities such as persons, addresses, emails, phone numbers, devices and so on. However, given their nature, entities such as addresses, emails, and phone numbers are simply disambiguated by proper standardization and filtering. It is often entities such as persons and devices that require more in-depth resolution techniques.

While various tools exist to perform advanced entity resolution, the current form of our solution relies on the generated networks for entity resolution. Let us consider the problem of resolving the person entity. It is possible that multiple people might exist with the same name, which is what essentially makes it a tricky problem. When information such as government-issued identification numbers are available along with the name (see Figure 2), simple standardization of the name compounded with such unique identifiers can provide the necessary resolution. However, when such information is not available, you can look to see if the two identical names belong to the same network. In some cases (especially fraud cases), multiple applications with different names can be submitted but these applications share other common entity values such as phone numbers. We can resolve these different names to be the same person with an arbitrary level of accuracy by ensuring that the appropriate entities are included in the network definitions. Moreover, factors such as the distance between the same two persons in terms of hops within the network can be used to make fuzzy determinations if they are the same person or not.

For the purposes of this solution, entity resolution is required to ensure that the proper Signatures and networks are retrieved during scoring. When an application arrives for processing, the networks associated with entities that do not need to be resolved are retrieved based on the lookup table generated during the network generation process. These networks are then combined (if more than one network is related to the application) to determine if the person (or any other entity that needs to be resolved) belongs to these networks. If they are not part of the network, a second round of lookup is performed based on the entity value compounded with associated identification numbers, such as Social Security numbers, driver's license number, and so on. As the final resort, the entity value is directly looked up, but this is likely to yield erroneous results if the entity value is very common in the population (for example, some common names).

In principle, entity resolution can be run either from the On-Demand Decision Engine or the orchestration layer. In our current implementation, it is run within the On-Demand Decision Engine. However, in lieu of our entity resolution mechanism, more specialized tools such as the HPENG procedure can be used for this purpose. In such cases, the orchestration layer serves as an ideal point from which entity resolution can be performed.

ON-DEMAND DECISION ENGINE (ODE)

The On-Demand Decision Engine brings everything together to assess the risk of a given application. This engine performs various tasks:

- accepts the incoming application event
- completes entity resolution for the various entities being used (though this can be performed within the orchestration also)
- fetches the Signatures associated with entities present in the application event
- fetches the network definitions associated with entities present in the application event
- passes the Signature, network definition, and incoming transaction to the model as inputs for scoring and executes the model
- appends the model results to the transaction and executes the rules
- performs post-processing tasks, such as writing back the updated Signature and processing rule outcomes, to issue responses and generate alerts

ANALYTICAL MODELS

The model is the interchangeable component of the solution through which machine learning plays a vital role in detecting fraud. Signatures and networks, along with the incoming transaction, provide the input for model execution. All these inputs are essentially used to derive features that drive the model. The efficacy of the solution is highly dependent on the performance of the model. Therefore, the network and Signature definitions are driven by the requirements of the model. The solution allows for seamlessly upgrading and downgrading Signature and network definitions as required by the model developer.

The solution does not depend on the type of model being deployed if the feature engineering and model code are consistent with its packaging requirements. Both supervised methods and unsupervised methods can used based on the model objectives.

Feature engineering is the single most important aspect of effective models. It is in this aspect that the power of Signatures and networks come into play. As stated earlier, many of the network attributes can be directly treated as features or used to derive other features. Similarly, Signatures serve as a source from which various behavioral features can be derived. Signatures continuously track the behavior of the entities, so they provide a mechanism for model adaptation since the features will evolve to represent the changing behavior. Table 2 lists examples of various features that are derived from the networks and various Signatures.

Network Level Features • cardinality of network • has a known fraudulent entity associated with network • various ages (network, oldest link, recent link, and so on) • network density, number of cliques present in the network, and so on • highest centrality
Node Level Features • shortest path lengths to other nodes / known fraudulent nodes • centrality measures • equivalence measures • number of connected edges
Features Common to All Entity Signatures • time since last application • number of applications • types of applications made • known fraudulent entity / blacklisted or whitelisted entity
Features Related to Persons' Signatures • degree of variance against previously reported demographic data, such as name variants, occupations, salaries, addresses, age, and so on • propensity for various behavioral factors such as channel through which applications are made, time of the day when applications are made, and so on • frequency of applications at other financial institutions (if available via bureau data)
Transaction Level Features • risk associated with various categorical variables such as occupation, employment status, time of application, product type, channel type and so on • distance to home based on IP address

Table 2. Sample Features Derived from Networks and Signatures

The model development process using these features is beyond the scope of this paper. However, the final output of the model is a score between 1 and 999, which represents the risk of the application. Moreover, the model can also produce reason codes that provide a verbal reason that explains the model score.

BUSINESS RULES AND DECISIONING

Output from any analytical process must be translated into business decisions. For this solution, the analytics in the form of a score that ranks relative risk and associated reason codes are presented for use in business rules. In addition, all elements can be used in rules. To this end, SAS Fraud Management allows users to author business rules and execute them in real time after model execution. This provides the ability to drive real-time responses, to drive alerts for human review, or to enact a secondary form of authentication.

The rules apply general decisioning logic whilst offering the business the day-to-day flexibility to amend and tune. The coding and guided options for rule development are supported by a robust rule estimation process. By using historical data to offer a look back assessment, this rule estimation process supports operational confidence from both a decision perspective as well as an out-sort percentage for applications to review, which can ultimately impact operational resources.

As with the models, the rules can also leverage enriched external data (such as credit reference agency data or third-party verification data) to look for flags indicating prior fraud risks. Digital identification information is also typically evaluated in the rules. The rules can also consult internal bank collated hotlists as well as those hotlists offered by industry sharing. These rules can be refreshed with new insights and business data.

Rules allow for:

- straight through processing, such as direct approval or decline of the application
- responses back to the applicant to request additional information, such as previous employer details
- referral initiation by creating a work item to allow assessment through an alert
- control of processing such as requests to third parties for additional data

Although the options around the business outcome decisions can vary, they can be completed in combination and be amended based upon risk tolerance, product request type, and so on.

FRAUD FEEDBACK

When fraud applications are detected during the decisioning process or at any point after account opening, it becomes what is arguably the most important piece of information for the success of the solution. Therefore, the ability to provide fraud feedback to the system is vital. Fraud is fed back to the system by retransmitting the original application event with a flag to indicate that it was detected to be fraud. The message also contains flags to indicate which of the several entities present in the application are truly compromised and fraudulent. The network generation process uses this information to mark fraudulent nodes within a network. Similarly, the Signatures can also be updated based on this feedback. Having information about fraud entities and events can significantly improve the analytic performance of the models. Moreover, it also provides an inherent black-listing mechanism for known bad entities.

CONCLUSION

In this paper, we presented a high-level overview of a real-time solution for monitoring and preventing application fraud. It is built on the highly proven and successful SAS Fraud

Management solution by enhancing current functionality and adding several new functionalities including network generation and entity resolution.

This application fraud solution enables users to make real-time approval decisions on applications by combining both network and behavioral aspects associated with entities alongside advanced analytic models and business rules. With sub-second responses times and high throughput rates, institutions will be now able to significantly enhance customer experience by reducing friction while protecting against fraud attacks.

REFERENCES

United States Federal Trade Commission. Fair Credit Reporting Act. Accessed March 21, 2019. Available https://www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/fair-credit-reporting-act

SAS Institute Inc. SAS Fraud Management. Accessed March 21, 2019. Available https://www.sas.com/en_us/software/fraud-management.html.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Prathaban Mookiah
SAS Institute Inc.
SAS Campus Drive
Cary, NC 27513
919-677-8000
prathaban.mookiah@sas.com