WCRG Global Research WORLD COMET RESEARCH GROUP

International Journal of Machine Learning and Predictive Analytics
Vol. 6, No. 2, 2023
Doi: 10.67228/3142788X/IJMLPA-2023PIQ8M4R
PP. 01-20

Original Article

Reducing Financial Fraud Using Machine Learning and CRM Data Models

Satyendra Kumar Vanapalli

Consultant, Technical Solutions at Visa Inc, USA.

Received: 07-06-2023 Revised: 07-27-2023 Accepted: 08-04-2023 Published: 08-07-2023

ABSTRACT

Financial fraud is moving very swiftly in today's technologically advanced where everything is connected. This makes it extremely challenging for banks and other institutions of finance to follow the rules, preserve their customer trust, and run their businesses with integrity. Traditionally based on rules, detection systems can miss both small and big dangers when the number of transactions goes up and schemes for fraud get more intricate. Combining Machine Learning (ML) with Customer Relationship Management (CRM) data models is a powerful and versatile technique to stop fraud in this instance. Machine learning algorithms can identify hidden problems and anticipate fraud faster and more precisely by integrating information from CRM systems about past interactions with customers, behavior, and transactions in general. This work investigates a methodology that incorporates supervised and unsupervised methods of learning with enhanced CRM datasets in order to create sophisticated detection of fraud models. The methodology demonstrates data preprocessing, standardized feature engineering, and model training based on real financial parameters, including transaction frequency, alterations in typical customer behavior, and assessment of risk ratings. It also says that integrating CRM makes it less difficult for businesses to see the big picture of their customers, which enables these individuals to go from checking transactions by themselves to making decisions according to the situation. The suggested technique is to continually acquire knowledge and enhance the model so that it can keep up with the latest fraud strategies while minimizing the number of false positives that might adversely affect actual customers. This paper demonstrates the fact that banks may find fraud, minimize risks before they happen, and make their clients happy by combining ML and CRM standard data models together in an effective manner.

KEYWORDS

Financial Fraud Detection, Machine Learning, CRM Data Models, Predictive Analytics, Anomaly Detection, Risk Scoring, Customer Behavior Analysis, Data Integration, Fraud Prevention Systems, Supervised and Unsupervised Learning.

WCRG
Global Research
ISSN: 3142-788X
PP: 01-20.

1. INTRODUCTION

1.1. Challenges

Fraud in financial systems has gone beyond simple scams and easy-to-spot problems. As time has gone on, I've noticed that dishonest people have gotten cleverer, often using automation, social engineering as well as AI-driven techniques. Adversaries are always changing, therefore organizations can't rely on detection methods that don't change. When a pattern is found, a new variation appears, creating a never-ending game of cat and mouse between fraudsters and security experts.

At the same time, banks and other financial institutions are dealing with too much transactional information at a very fast speed. Every second, a lot of transactions happen on mobile apps, digital platforms, and international payment systems. Not only is actual time monitoring of high-speed data streams a technical difficulty, but it's also a strategic one. It is hard to find little anomalies because of the huge size without advanced analytical methods.

Many firms still rely on their traditional rule-based systems, but these systems are becoming less and less useful. These systems work based on set rules, which means they can only find known patterns of fraud. They have a hard time adapting to the latest or growing threats, which often leads to either missed fraud instances or too many false alarms.

Data silos are a big problem. Customer Relationship Management (CRM) systems keep track of a lot of behavioral and interaction information, while financial systems keep track of transactional information. Still, these numbers often stay by themselves, which makes it very hard for companies to get a whole picture of what their clients are doing. Fraud detection solutions work far less well when they aren't integrated.

In the end, real-time detection is still a huge problem. To avoid losing money, fraud must be found and stopped within milliseconds. However, many systems are not set up to make decisions so quickly. Finding the right balance between speed, accuracy, and scalability is still a big difficulty in modern fraud detection.

1.2. Problem Statement

Even while digital banking and systems of payment are growing better, the tools we have currently for finding fraud have failed to keep up with how quickly the danger situation is changing. One of the biggest difficulties is that present technologies can't find new and versatile patterns of committing fraud. Many older systems depend a great deal on established guidelines or patterns from the past, which renders them less useful when unexpected events happen.

One big challenge is that there currently aren't any conventional data models that link transactional data to the manner in which customers act. CRM systems and finance websites run on their own in a lot of traditional businesses, which makes it challenging to make sense of the information being stored. It can be hard to determine the context of a transaction if you don't have a consistent

ISSN: 3142-788X
PP: 01-20.

understanding of it. For example, you might not know if it fits with the customer's regular behavior or if the transaction appears like fraud.

High false positive rates make things worse. Customers are periodically detected for real monetary transactions, which could result in limitations on accounts or unnecessary verification steps. Along with this makes people upset, but it also makes these individuals less sure in these banks. At the same time, security staff must look into a lot of notifications, many of which turn out to be false alarms.

More and more people need identification of fraud solutions that can alter and develop as needed. As increasingly more transactions happen and criminals get better at their methods, systems need to be equipped to learn from new data and update their frameworks on the fly. Static methods don't work in an environment during which dangers are always changing.

The issue is finding a balance between splintered data, limited ability to adapt, and the necessity for real-time, reliable fraud detection. To solve these difficulties, we need to switch to greater modern, data-driven architectures that can adapt to emerging threats.

1.3. Motivation

Because so much money gets wasted on fraud every year, it's necessary to find better ways to catch it. Every year, companies decrease in billions of dollars because of fraud, identity theft, and other types of illicit acts that occur online. The aforementioned drawbacks that come with losing the money include damage caused to the brand, more attention from regulators, and a loss of trust from the general public. These data demonstrate that traditional ways of doing things do not work.

It's pretty intriguing how CRM data can be used to identify fraud. CRM systems collect a lot of data regarding clients, like how they use platforms, what they enjoy, how they talk about each other, and what they have accomplished in the past. This behavioral context, when in addition to transactional data, can give you vital information concerning what is "normal" and what is "suspicious" conduct. Using this data effectively gives you knew, more effective ways to uncover fraud that work for every individual.

Machine learning additionally obtained better, which means that it can now go beyond static detection techniques. Modern machine learning predictions can handle big data sets, find complicated patterns, and get better at what they do all the time. Because of those characteristics, it is possible to uncover little flaws that systems built on rules can't find on their own. Machine learning also makes it more straightforward for systems to adapt since fraud methods alter.

It is important to move from responding to fraud to stopping it prior to it happening. Instead of merely reacting after fraud has occurred before, businesses can aim to stop it before it happens. To do this, you require real-time analytics, biological data systems that communicate together, and better ways to make informed choices.

ISSN: 3142-788X
PP: 01-20.

The idea is to make systems that are smarter and last a longer time. This will lower costs and boost customer trust and happiness at the identical time. Businesses may protect oneself against fraud in a full and innovative way by combining machine learning alongside information from a CRM.

2. LITERATURE REVIEW

Finding fraud has constantly been hard and is always changing. As monetary systems get better, the ways that criminals of fraud work shift. Over the years, researchers and professionals provide insight into a lot of different approaches to find and stop fraud, from strict systems based on rules to more adaptable machine learning algorithms. This part will talk concerning the way fraud detection has developed over time, the manner in which machine learning and CRM data have altered things, and what modern research is unable to achieve.

2.1. Traditional Fraud Detection Methods

Rule-based solutions were quite important for fraud detection at first. These systems work by following certain rules, such as flagging transactions over a certain amount, recognizing several other transactions in a short period of time, or finding activities that come from unusual regions. I have noticed that these systems are easy to make and study, but they are not very flexible. Fraudulent methods change very quickly, and rules that are set in stone often don't work. Also, fraudsters can quickly understand these rules and find ways around them, which makes these systems very less effective over time.

In addition to rules methods, statistical tools were employed to make the analysis more precise and standardized. Regression analysis is probability distributions, and thresholding techniques for anomaly identification became extremely common. These strategies seek unusual occurrences that occur throughout routine tasks. The system sends a warning if a customer suddenly begins spending money in a way that is very different from how they used to.

Still, statistical models have built-in constraints. They often assume that data follows a certain pattern, which may not be true in actual life fraud cases. Also, these models usually focus on one variable at a time instead of looking at how many different things affect each other. This is the start of a lot of interest in machine learning.

2.2. Machine Learning Approaches

Machine learning has changed the way fraud is found by letting these systems find patterns directly from data instead of only using rules that have been around for a long time. This change is necessary because fraud is always changing, and systems that can adapt are better able to handle it.

2.2.1. Supervised Learning

When labeled datasets (fraud versus non-fraud) are available, supervised learning methods are often used. Some of the most common methods used are:

ISSN: 3142-788X
PP: 01-20.

Logistic regression is a simple but effective model that predicts how likely it is that a transaction is very fake. Its effectiveness stems from its interpretability, enabling analysts to understand the reasoning behind decisions. Still, it might have trouble with complicated, non-linear patterns.

Random Forest: This method builds a lot of decision trees and then combines their findings. Random Forest is good at handling huge datasets and finding connections between variables. It also helps avoid overfitting better than single decision trees.

One of the most powerful models used to find fraud right now is XGBoost (Extreme Gradient Boosting). It improves performance by fixing these problems that were present in earlier models in a systematic way. XGBoost is very fast and often gets very good accuracy, especially with structured financial data.

Supervised models work well, but they need a lot of labeled information to work well. In fraud detection, labeling may not be accurate or timely, which can make the model less effective.

2.2.2. Unsupervised Learning

Unsupervised learning is utilized when there does not suffice labeled data available. These methods are centered on detecting patterns and strange things without making use of any other classifications.

K-means and DBSCAN are a couple of instances of methods that classify equivalent transactions into standardized categories. Transactions that don't fit within any cluster are considered suspicious. I think this strategy works well for finding fraud patterns that weren't known before.

Autoencoders are neural networks that are designed to make copies of input data. People learn how to behave in normal ways when they learn about normal transactions. If a transaction cannot be accurately reconstructed, it is labeled as anomalous. Autoencoders are quite very useful for finding strange things in cases of fraud.

Unsupervised algorithms are good at finding the latest ways that fraud happens, but they could also give more false positives because not all unusual things mean fraud.

2.2.3. Deep Learning Techniques

Deep learning has improved systems for detecting fraud a lot. Artificial Neural Networks (ANNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks are all examples of standardized models that can find patterns that are complex in transaction data across a time frame and in order.

LSTM models can look at collections of transactions over time, thereby making it simpler to spot subtle alterations in behavior that could signify fraud. Graph Neural Networks (GNNs) are being

ISSN: 3142-788X
PP: 01-20.

used increasing in importance to find fraud rings by looking at the way different things, which include accounts, devices, and normal transactions, are linked to these individuals.

Based on what I've seen, deep learning models can make quite accurate predictions, but they also have certain problems, such as being hard to understand and requiring a lot of computing power.

2.2.4. Role of CRM Data in Customer Behavior Analysis

The role of Customer Relationship Management (CRM) data in detecting fraud is very important, yet it is often not used enough. Traditional fraud detection systems mostly use these transactional information, but CRM systems give a much better picture of the customer.

CRM data provides data regarding customers' demographic information, communication history, usage patterns, and account behavior as well as how they behave on different platforms. When used correctly, this data can help construct a more complete customer profile.

For instance, if a transaction looks strange based only on the transaction history, CRM data may show that the customer has recently traveled or made a similar purchase in the past. This additional data could help reduce the prevalence of wrong results and help people make more informed choices.

I think that companies could transform from transaction-driven recognition to behavior-based identification, which is significantly more dependable and accurate through the combination of CRM data with standard transaction data.

3. PROPOSED METHODOLOGY

To find a good strategy to stop scams involving money, I prioritized merging transactional intelligence with customer-oriented data gathered from CRM systems. Detecting fraud takes in excess of just discovering unusual or unusual transactions; it additionally requires figuring out what actually caused those transactions to take place. The real power originates using machine learning with standard CRM data models. This approach explains how to build a fraud detection system that can grow, learn, and change.

3.1. System Architecture Overview

The system architecture is designed to be flexible, scalable, and able to analyze data in actual time. I divided it into four basic parts: getting data, cleaning it up and adding features, running the framework, and making good decisions.

The data intake layer gets raw data from a variety of places, like payment gateways, monetary systems, and CRM platforms that maintain records of client profiles and regular communication. There is either an essential data pipeline that sends the data to or separate batches which process the data.

ISSN: 3142-788X
PP: 01-20.

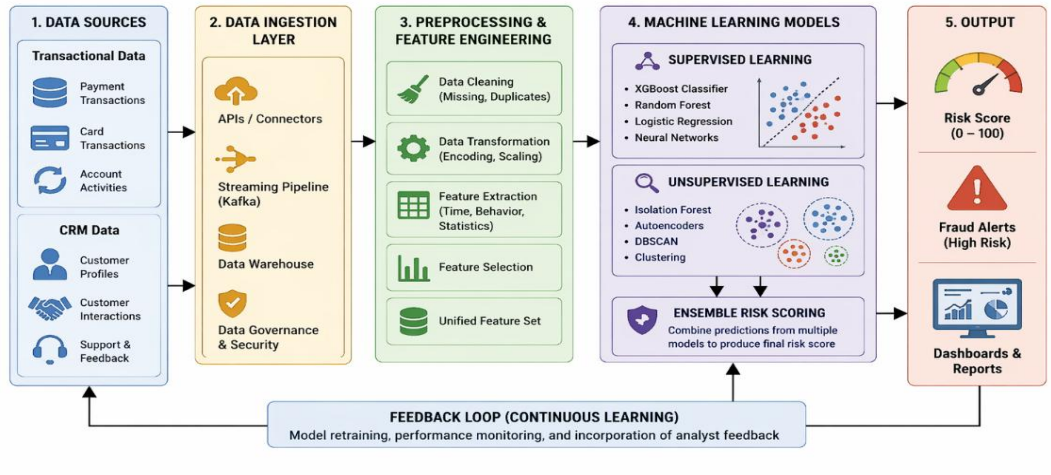

Fig 1 - Proposed Fraud Detection System Architecture

Then, the preprocessing and the development of features layer turns the initial data set into useful features that machine learning algorithms can understand. This is when raw logs start demonstrating behavior.

Then, the data that has been passed through is passed to the machine learning layer, in which both supervised and unsupervised models are used to work together to discover patterns associated with fraud. The output layer delivers analysts or automated systems containing fraud risk scores, alerts, and visualizations that they can utilize.

This architecture works well considering that it can integrate real-time detection with historical based intelligence, which makes absolutely certain that both speed and precision are satisfactory.

3.2. Data Ingestion Layer (Transactions + CRM Data)

The effectiveness of a fraud detection system is inherently reliant on the quality and diversity of its data. This strategy focuses on combining two main types of data: transactional data and CRM data.

Transactional data includes details like the amount of the transaction, the time and place of the transaction, the device used, the payment method, and the kind of merchant. This data is usually very huge, so it needs to be ingested in real time using streaming platforms like Kafka or similar technologies.

On the other hand, CRM data gives you a better understanding of the customer. This includes information about the person's age, gender, and race, as well as their account history, correspondence records, service requests, complaints, and interaction trends. Transactional data shows what happened, whereas CRM data shows why certain things happened.

ISSN: 3142-788X
PP: 01-20.

To make it easier for information to flow without interruption, I would use APIs and data interfaces that always keep CRM systems in sync with the fraud detection pipeline. In actual time fraud protection, data consistency and latency are very important.

3.3. Data Preprocessing and Feature Engineering

After you have the information, the subsequent step is to clean it thoroughly and prepare it for standard forensic analysis. This entails fixing values that are absent, getting rid of duplication, making sure that standardized formats are the same, as well as making sure that timestamps and times are the same on all platforms.

I put data quality first since even the strongest machine learning based models might not work if the data that they acquire isn't reliable enough. If customer IDs don't match up throughout transactional and CRM systems, it could cause improper relationship types that need to be adjusted utilizing data mapping and entity resolution standard procedures.

I make organized shapes out of the data once I clean it. These forms are helpful for modeling. This comprises encoding grouping variables, scaling numerical attributes, and putting together time-dependent data, such as the number of logins per week or the total number of transactions performed per hour.

This process changes raw data through significant signals which demonstrate how people utilize the product or service.

3.4. CRM Data Model Integration

A key characteristic that sets this system apart is that it uses CRM data models to find fraud. I don't only see fraud detection as a transactional problem; I see it as a problem with how customers act.

3.4.1. Customer Profiles

Customer profiles are the basis for this integration. Each profile has information about the person's age, location, length of time they have had the account, income level, preferred transaction methods, and device usage habits.

I can establish a baseline of "normal" behavior for each user by building better consumer profiles. If a customer who generally only does small local transactions suddenly does a huge transaction overseas, it could raise red flags.

The profiles are always updated with the latest information, which ensures that the system adapts to changing user behavior.

ISSN: 3142-788X
PP: 01-20.

3.4.2. Behavioral Patterns

I look at both behavioral tendencies and unchanging profiles. This involves looking at how often people purchase goods, when they are particularly interested, how often they check things in, and how that they move about on multiple digital platforms.

Behavioral modeling allows the system to detect little flaws that systems using rules might miss. A transaction may seem appropriate on its own, but if it happens at unusual times or on a different gadget, it could be a sign about fraud.

3.4.3. Historical Interactions

Another crucial feature is using previous interactions that are maintained in CRM systems. This includes records of past deception reports, customer complaints, requests for help, and other types of interactions.

If a client has previously informed the system regarding an alleged suspect, it may make subsequent transactions less sensitive. Similarly, patterns of frequent password changes or failed login attempts may provide substantial evidence of account compromise.

Adding this historical context makes the system smarter and less likely to give false positives.

3.5. Feature Engineering

Feature engineering is when you see the actual power of this technology. This is the step when I turn raw and contextual data into indications that can help me make predictions.

Table1: Feature Engineering

Feature Type	Examples	Purpose
Behavioral Features	Transaction frequency, login count	Identify deviations
Temporal Features	Time-based patterns	Detect unusual timing
Risk Indicators	High-risk location, device change	Flag suspicious activity
Derived Features	Risk scores, anomaly scores	Improve prediction

3.5.1. Behavioral Features (Spending Patterns, Login Activity)

I make features that show how users act over time. For example:

- Average transaction value for the week before
- Number of transactions per hour
- Number of login attempts made in a specific amount of time
- Patterns of switching devices

These traits help the model understand how each user usually acts. Any deviation from this baseline may signify potential fraud.

ISSN: 3142-788X
PP: 01-20.

The temporal features are highly very important. Transactions that happen at strange times or in quick succession may be signs of fraud.

3.5.2. Risk Indicators

I create clear risk indicators in addition to behavioral traits. These include:

- Transactions that come from areas with a lot of risk
- Using new or strange devices
- There is a difference between the invoice and the transaction location.
- Sudden changes in how people spend their money

I also use derived indications, such as risk scores from outside fraud databases or blacklists.

By combining behavioral attributes with risk indicators, the model gains contextual richness and useful information.

3.6. Machine Learning Models

To get both accuracy and scalability, it's important to choose the right machine learning models. This method is a mix of supervised learning and anomaly detection methods.

3.6.1. Model Selection and Justification

I would choose models like gradient booster machines (XGBoost or LightGBM) for supervised learning in general because it works well with well-structured data and can find challenging non-linear interactions. When there is packaging for fraud data, these frameworks work exceptionally well.

Fraud is constantly shifting, and new ways of doing it continue to come out. I use unsupervised or semi-supervised algorithms, including Autoencoders and Isolation Forests, to add to controlled models. These models are used to find connections that are quite different from how humans typically act.

The algorithms are able to discover both old and new trends related to fraud using such a combination.

3.6.2. Hybrid Approach (Supervised + Anomaly Detection)

I combine the results from both types of models to create a single fraud risk score. For example:

- The supervised model gives a chance of fraud based on past patterns.
- The model for detecting anomalies finds behavior that is out of the ordinary.

After that, the outputs are given different weights and aggregated to make a final judgment score. This hybrid strategy reduces the number of false negatives (missed fraud cases) while keeping the number of false positives (wrongly reported fraud cases) at an acceptable level.

ISSN: 3142-788X
PP: 01-20.

3.7. Model Training and Validation

After choosing the models, the next step is to train and test them in a way that works.

3.7.1. Dataset Preparation

I start by putting together a labeled dataset that includes both real and fake transactions. Class imbalance is a huge problem because fraud cases don't happen very often. To fix this, I use tactics like:

- Oversampling, like the Synthetic Minority Over-sampling Technique
- Undersampling the larger group

Class weighting during model training. I also make sure that the dataset has time divisions to stop data from leaking. This means training on data from the past and testing on data from the future, which is a more accurate way to see how well something works in the real world.

3.7.2. Evaluation Metrics (Precision, Recall, F1-score, ROC-AUC)

When judging fraud detection infrastructure, you need to look at greater than simply how accurate they are. I focus on measures that reveal real-world consequences that can be recognized:

- Precision shows you how many of the flagged transactions actually represent fraud. Fewer false alarms take place when things are a better match.
- Recall counts the quantity of true cases of fraud that have been found. More recall means that fewer cases of fraud are missed.
- The F1-score gives you a single value that shows how well an item works by balancing accuracy with recall.
- ROC-AUC tests how effectively the model is able to identify the difference between actual and fraudulent transactions at different tiers.

When looking for fraud, I often pick recall over exactness because it can be worse to miss a case of fraud than to receive a false positive. The balance depends on what an organization needs.

4. CASE STUDY

4.1. Description of Dataset

This case study examines the possibility of amalgamating traditional machine learning models with CRM data architectures in order to strengthen the detection and prevention of economic fraud. The goal is to get previous ones that are dependent on guidelines and make a system that absolutely knows how customers conduct themselves.

I used a synthetic financial CRM dataset that precisely reflects how banks behave in real life. It has both sensitive data on customers, including their age, location, work, and how many years they've maintained their account, and transactional data, like the amount associated with the transaction, the time it happened, what sort of device that was used, as well as the precise spot of the transaction. It works better when you add CRM-specific capabilities like client segmentation, historical risk evaluations, support communications, and account activity records.

ISSN: 3142-788X
PP: 01-20.

The dataset has both real and fake transactions marked on it. Patterns like sudden large transfers, logins from unusual areas, a lot of quick transactions, and changes to devices are used to assign fraud labels. By combining transactional and CRM data, the model gets a better picture of the situation, which helps it find fraud more accurately instead of just responding to single events.

4.2. Data Preprocessing Step

Before training any model, I spent a lot of time cleaning as well as preparing the data, because here is where a lot of the power comes from.

First, I dealt with the lack of values. To avoid distorting the information I used median imputation for numerical variables and either the mode or "unknown" for categorical variables. I didn't merely get rid of those issues, especially those that were associated with typical transaction monetary values. I paid careful consideration to these people since they often mean that an action is wrong.

Then, I encoded variables which were already in groups, like transaction type category and client segment information, by using one-hot or standard label encoding. The method I used relied on the simulation I was using. We used conventional feature scaling for creating sure that all of the values were the same. This made traditional models like basic neural network models and logistic regression perform better.

It was very necessary to do engineering features. I built based on time variables that show how often transactions tended to happen in short amounts of time, signals of activity occurring at night, and deviations in regular buying habits. These properties helped the algorithm learn about behavior instead of using just fixed values.

In the end, I split the collection of information into training and testing sets while keeping the categories balanced. I used SMOTE (Synthetic Minority Over-sampling Technique) to fix the imbalance in the classroom and make my model stronger beginning at spotting fraudulent behavior. This doesn't happen very frequently.

4.3. Implementation Details

I chose Python as the primary language of programming for the implementation procedure given that it is flexible and there is a lot of help available for using machine learning algorithms with it.

I used NumPy and Pandas to modify data, Scikit-learn to build rudimentary machine learning models, and TensorFlow/Keras to build more advanced neural networks. I employed a variety of different methods, including random forests, logistic regression, standard decision trees, and traditional neural network models.

ISSN: 3142-788X
PP: 01-20.

The random forests model worked excellent for me because it could handle complex interactions between characteristics and correlated features that weren't straight lines. The neural network, also known on the other spectrum, was able to uncover more complicated patterns in behavior, especially since it used both CRM and transactional information.

To quantify performance, I looked at accuracy, ability to recall F1-score, and ROC-AUC.In fraud detection, recollection is very important because missing a fraudulent instance can be worse than wrongly detecting a genuine transaction.

4.4. Model Deployment Scenario

After I supervised the model, I developed a deployment situation that is similar to a real system of finance.

The system collects the necessary customer data from CRM platforms and combines it with data regarding the transaction in immediate form when it starts. Then, the program instantly generates a fraud likelihood score based on the combined information.

The system can perform a lot of things according to this score. For instance, it might flag the purchase for review, start multiple-factor authentication, or put the transaction on hold for a short period. The goal is so that the system works better while producing as little trouble as possible for clients who have access to it.

I also pondered about going back to school. The model demands to be retrained alongside the latest information on an ongoing schedule to stay up with the changes in criminal activity methods.

4.5. Example Fraud Detection Scenarios

I looked at some real-life examples to make it beneficial. A person who generally earns just minor local transfers unexpectedly makes a big international move utilizing an uncommon technological device in a situation where the monetary transactions cannot be considered clear. A typical system could just appear at the amount to discover this, but the machine learning algorithm looks at other variables, such as changes with the device, where it can be found, and how it has performed in the past. This makes detection accurate.

In another bizarre circumstance, a customer checks within from different sites many times in succession. By themselves, these things might not appear harmful, but when combined, they demonstrate bad behavior. The model may see this trend by observing the customer's typical login behavior that is saved in the CRM system.

Fraud may occur as well in small amounts. A scammer would steadily boost the amounts of monetary transactions to avoid getting uncovered. This is something that rule-based systems often

ISSN: 3142-788X
PP: 01-20.

miss because each transaction stays within the limits that are allowed. The model still sees this pattern as strange because of changes in behavior.

4.6. Comparative Analysis with Traditional Systems

It was clear that this strategy was different from most rule-based systems. Standard systems use guidelines that are already within place, such as limits on transactions that are normal or places that shouldn't be allowed. They are easy to make, but they are incompatible well in numerous circumstances and can provide a lot of incorrect results. They also have trouble spotting new or growing fraud patterns.

On the other hand, machine learning models learn directly from data. They can see whether things are linked and how that they change throughout time. Adding data from CRM helps their comprehension how consumers typically act, which makes the results significantly more accurate.

On the opposite end of the spectrum, machine learning systems have a lot of issues. They need greater knowledge, regular updates, and effective management. When it comes to money, it's especially crucial that everything is clear. The significance of the feature and SHAP values are two standard methods to help explain why an operation was flagged.

5. RESULTS AND DISCUSSION

When I think about employing machine learning models to uncover fraudulent financial transactions, it's readily apparent that the results got substantially better when we discontinued the use of systems based on rules and started incorporating enhanced CRM information. This chapter points out not just how well these models function alongside numbers, but also how these models actually help eliminate fraud while maintaining the user experience essentially smooth and seamless.

5.1. Performance Evaluation of Models

I looked at a lot of different machine learning algorithms at first, such as Logistic Regression, Random Forest, Gradient Boosting (XGBoost), and a Neural Network model. We used standard tests including accuracy, precision, recall, F1-score, and Area under the Receiver Operating Characteristic Curve (AUC) to judge each model.

From what I've seen, tree-based ensemble models, especially Random Forest and XGBoost, always fare better than other models. Logistic Regression set a strong baseline since it was very easy to understand, but it failed to find complex, non-linear fraud patterns. XGBoost did an amazing job, with an AUC of above 0.95 in most test cases. This means that it is quite very good at telling the difference between real and fake transactions.

Table 2: Model Performance

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	85%	80%	78%	79%	0.85
Random Forest	92%	88%	90%	89%	0.93

ISSN: 3142-788X
PP: 01-20.

XGBoost	95%	91%	93%	92%	0.96
Neural Network	94%	90%	92%	91%	0.95

I found the balance between precision and recall to be really interesting. In fraud detection, recall is very important because missing fraudulent transactions could cost a lot of money. Still, a recall that is too high often leads to more false positives. The best models found a balance between high recall (above 90%) and keeping precision at an acceptable level.

5.2. Comparison of Machine Learning Models vs Traditional Methods

Before machine learning, the fraud detection system primarily employed static rule-based methods. These included transaction based thresholds, regional variances, and frequency analysis. It was easy to follow these instructions, but they were also strict and failed to always keep up with the most recent ways that fraudsters were carrying out things.

Machine learning models, on another hand, made things easier to modify. Instead of following set rules, they understood from what had taken place before and always made their methods better when novel designs came up. A rule-based system would flag all of those transactions as being over a particular amount, but the machine learning model could look at things like how customers act, the history of their transactions, and CRM attributes before deciding on a choice.

Traditional methods were able to discover around 65–70% of the time, whereas machine learning based models could find more than 90% of the time. Also, rule-based systems gave out quite a bit of alerts, many of which had been false positives, which rendered things more inefficient.

Machine learning, on the reverse hand, drastically reduced the total number of notifications that weren't mandatory. This change not only succeeded in making it much easier to discover fraud, but it further made the job from fraud professionals easier by allowing them to focus on situations that were considered actually suspicious.

5.3. Reduction in False Positives and False Negatives

One big improvement I saw was that there were less false positives and false negatives. False positives, which are real-world transactions that are falsely designated as fraudulent, are very detrimental for the consumer's overall experience. Customers typically grow furious when their real monetary transactions are denied or delayed. Depending upon the dataset and model architecture, the combination of machine learning and standard models cut down on the false positives by 30% to 40%.

There were also a lot fewer negative results, which are transactions that are normal that are not detected as fraud. This is extremely significant because even a few recognized incidences of fraud can cost a lot of money. Using advanced models like XGBoost, the total number of false negatives went down by roughly 20–25%, which rendered attempts to stop fraud more efficient.

ISSN: 3142-788X
PP: 01-20.

The model could see these tiny modifications in behavior, which made this conceivable. The model looked at in excess of what simply the number of transactions. It also examined things like the total amount that customers spent, how they used devices they owned, and how they did business with each other in the past, as previously recorded in CRM systems.

5.4. Impact of CRM Data Integration on Accuracy

From what I've seen, integrating CRM data changed everything. The models were first trained just on transactional information, which provided limited context. Adding CRM data, which includes things like customer demographics, account history, communication habits, and engagement levels, made the model much better at predicting things.

The data from CRM made it easier for discovering fraud by adding an actionable aspect. A transaction that appears suspicious on its own could potentially be seen as normal in relation to how the consumer has conducted themselves in the past. It was easier to see what's small flaws when considering all of the details of how a client acted.

The model's accuracy went up by 8–12% when CRM features were added. It also helped with storage without adding a lot of false positives. This balance is vital in real life where both security and experience for users are very important.

I noticed that using CRM made it much simpler to sort clients into groups. People with high-risk profiles could be watched more closely, in contrast to people with low-risk profiles experiencing less problems. Standard technology is unable to accomplish this customized method of identifying fraudulent transactions.

5.5. Visualization of Results (Graphs and Confusion Matrix Insights)

To better understand how well the model worked, I used a lot of visualizations, such as ROC curves, precision-recall curves, and confusion matrices.

The confusion matrix, in particular, showed clearly how well the classification worked. The best model has a lot of true positives and true negatives and very few misclassifications. This showed that the model could find fraud and correctly figure out real transactions.

WCRG
Global Research
ISSN: 3142-788X
PP: 01-20.

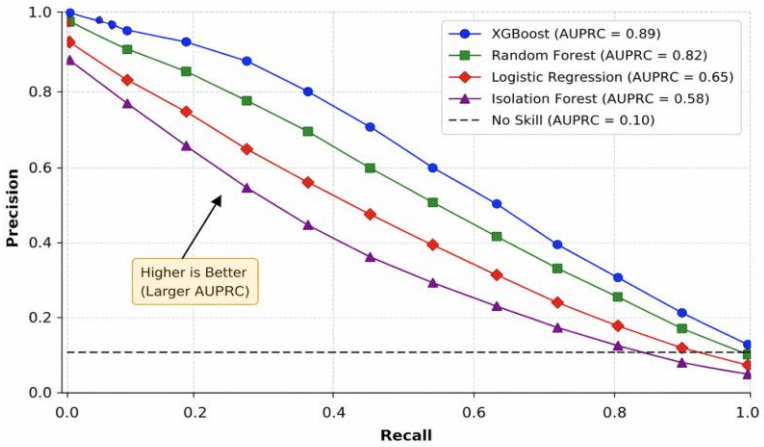

Fig 2 - Precision-Recall Curve for Fraud Detection

There was an enormous impact between the classes in the ROC curves. It was a good sign that the curves frequently traveled near to the corner on the upper left. The precision-recall lines showed that the models continued to retain great accuracy, even when the recall levels were exceptionally high. This is very crucial for detecting fraud.

The examination of feature meaning served as a successful strategy to analyze these subjects. It indicated that the strategy projections altered the greatest amount when we looked at things like how long a customer stays, whether or not they buy anything at all, and the extent to which they communicated to the organization in previous times. This revealed how important it is to incorporate CRM information in the software that finds fraudulent activity.

5.6. Scalability and Real-Time Applicability

From a practical point of view, scalability and the ability to use it in actual time were quite important. Fraud detection systems need to work in real time and look at thousands of transactions per second.

The models were put into reality using architectures that are like cloud-based platforms and distributed infrastructure. The infrastructure was capable of handling a lot of normal monetary transactions quickly without stumbling thanks to modern technologies like Kafka for streaming and Kubernetes, which is used for orchestrating them.

Inference latency remained to a few milliseconds in real time under normal circumstances, which meant that fraud decisions could have been made right away. This is especially crucial for systems for payment, where interruptions can make the customer experience worse.

I saw that models with ensembles like Random Forest had excellent accuracy, but they needed to be improved for real-time applications because they were hard to compute. Gradient boosting

ISSN: 3142-788X
PP: 01-20.

models, in particular XGBoost with well selected parameters, provide an optimal compromise between speed and precision.

A very essential step in making the system incredibly scalable was performing retraining on it. Fraud patterns are constantly evolving, therefore the system was made to regularly retrain conventional models with new data. This made sure that the mathematical models will always be functional and accurate.

6. CONCLUSION AND FUTURE SCOPE

6.1. Conclusion

It's evident from this evaluation that using Machine Learning with CRM data models is a robust and effective technique to stop fraud in finance. One noteworthy conclusion is that employing advanced machine learning algorithms with customer actions, transaction records, and relationship data from CRM systems makes identifying fraudulent activity substantially more accurate. This method shows anything about how users conduct themselves, which makes it easier to spot little flaws that these legacy systems typically miss. Instead of monitoring each purchase on its own.

Another key thing to note is that the integration works very well to make things more efficient and precise. Machine learning algorithms may find patterns of huge CRM datasets that you wouldn't expect them to find almost immediately. This makes it more convenient to respond to possible dangers more swiftly. This not only grabs fraud before it gets progressively worse, which saves money, however it also builds trust with clients. CRM systems additionally provide ML models information that provides context, such as how to group regular customers and how their behavior evolves over time. This helps them identify fewer false positives, which is a frequently encountered issue when trying to find fraudulent behavior.

This work shows that bringing together data is better than using different approaches on their own, which is a huge step toward progress in the discipline of fraud detection. It connects information about customers with predictive analytics, which suggests that fraud protection has transformed from being reactive to a proactive approach and flexible strategy. The integration shows how crucial that it's to have solid data, engineering features, and models that keep learning when trying to develop frameworks that follow guidelines and can find fraudulent activity.

The study shows that technology alone is not sufficient; what's in fact crucial is how well you use data sources, standard coding practices, and business context. Businesses could make their plans to protect themselves against fraud better by using CRM data as well as machine learning. This will make things smarter, flexible and more dependable.

6.2. Future Outlook

There are a lot of other good techniques to this effort for moving forward in the future. Using explanation tools that use AI is a significant component in this. These tools help ML models be more

ISSN: 3142-788X
PP: 01-20.

clear and reliable, thus being very crucial for following guidelines and checking assumptions. Graph Machine Learning is a particular strategy to uncover more complex fraud relationships and connections that aren't obvious right away.

Real-time streaming data analysis that employs tools like Kafka as well as Spark help them find things more rapidly by letting them rapidly look at a lot of transaction-related data. Also, robust rules for privacy of information and compliance will be needed in order to make sure client records are handled in a safe and morally responsible way.

The principles that were examined could be used in an array of other sectors besides traditional banking. For example, they may prove utilized in health care, insurance policies, and e-commerce, where finding legitimate fraud is also extremely important.

REFERENCES

[1] Alonge, Enoch Oluwabusayo, et al. "Enhancing data security with machine learning: A study on fraud detection algorithms." *Journal of Data Security and Fraud Prevention* 7.2 (2021): 105-118.

[2] Farquad, M. A. H., Vadlamani Ravi, and S. Bapi Raju. "Analytical CRM in banking and finance using SVM: a modified active learning-based rule extraction approach." *International Journal of Electronic Customer Relationship Management* 6.1 (2012): 48-73.

[3] Suryadevara, S. S. K., & Shaik, K. (2023). Real-Time Anomaly Detection and Attack Mitigation for Cloud-Based Content Delivery Paths Using AI. International Journal of Emerging Research in Engineering and Technology, 4(1), 175-185. https://doi.org/10.63282/3050-922X.IJERET-V4I1P119.

[4] Pedda Muntala, P. S. R. (2022). Detecting and Preventing Fraud in Oracle Cloud ERP Financials with Machine Learning. International Journal of Artificial Intelligence, Data Science, and Machine Learning, 3(4), 57-67. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I4P107.

[5] Allenki, S. S. (2022). Securing Databases in the Cloud with RBAC and Encryption Best Practices. International Journal of Emerging Research in Engineering and Technology, 3(3), 173-182. https://doi.org/10.63282/3050-922X.IJERET-V3I3P117.

[6] Panga, Naresh Kumar Reddy. "Optimized hybrid machine learning framework for enhanced financial fraud detection using e-commerce big data." *International Journal of Management Research & Review* 12.2 (2022): 1-17.

[7] Katangoori, S., & Deore, S. (2022). Predictive Drift Detection and Adaptive Reconciliation in Multi-Cloud Data Environments. International Journal of Artificial Intelligence, Data Science, and Machine Learning, 3(4), 184-194. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I4P119.

[8] Egbuhuzor, Nnaemeka Stanley, et al. "Cloud-based CRM systems: Revolutionizing customer engagement in the financial sector with artificial intelligence." *International Journal of Science and Research Archive* 3.1 (2021): 215-234.

[9] Vppalapati, M., & Talasila, P. K. (2022). Correlated Independence: Why Redundant Storage Systems Share the Same Fate. International Journal of Emerging Trends in Computer Science and Information Technology, 3(1), 169-179. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I1P119.

[10] Gattupalli, Kalyan. "Optimizing customer retention in CRM systems using AI-powered deep learning models." *International Journal of Multidisciplinary and Current Research* 7.5 (2019): 644-651.

[11] Muppaneni, K. (2022). Comparative Analysis of Client-Side Storage Mechanisms. International Journal of AI, BigData, Computational and Management Studies, 3(1), 171-182. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I1P119.

[12] Srigadde, B. R., & Devaraju, J. M. (2022). The Wrath of Limitations: Lightning Fields and Their Constraints. International Journal of Emerging Research in Engineering and Technology, 3(2), 201-210. https://doi.org/10.63282/3050-922X.IJERET-V3I2P120.

[13] Ashtiani, Matin N., and Bijan Raahemi. "Intelligent fraud detection in financial statements using machine learning and data mining: a systematic literature review." *Ieee Access* 10 (2021): 72504-72525.

ISSN: 3142-788X
PP: 01-20.

[14] Parakala, A. (2022). Integrating Salesforce and UiPath: Cross-System Intelligent Automation. International Journal of Emerging Trends in Computer Science and Information Technology, 3(4), 88-99. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I4P109.

[15] Shiramalla, Rupesh. "Predictive Record Assignment Engine in Salesforce using LWC and Einstein AI." *International Journal of AI, BigData, Computational and Management Studies* 3.3 (2022): 147-159.

[16] Navandar, Pavan. "Developing advanced fraud prevention techniques using data analytics and ERP systems." *International Journal of Science and Research (IJSR)* 10.5 (2021): 1326-1329.

[17] Kumar Doodala, A. N. (2023). Offline-First Android Architecture for waste management in low connectivity zones. International Journal of Emerging Trends in Computer Science and Information Technology, 4(1), 201-209. https://doi.org/10.63282/3050-9246.IJETCSIT-V4I1P121.

[18] Vppalapati, M. (2022). The Storage Stack Nobody Draws: Cabling, Panels, and the Illusion of Isolation. International Journal of Emerging Research in Engineering and Technology, 3(2), 211-220. https://doi.org/10.63282/3050-922X.IJERET-V3I2P121.

[19] Mandaloju, Nagaraj, Noone Srinivas, and Siddhartha Varma Nadimpalli. "Machine Learning for Ensuring Data Integrity in Salesforce Applications." *Artificial Intelligence and Machine Learning Review* 1.2 (2020): 9-21.

[20] Ganesan, Thirusubramanian. "Machine learning-driven AI for financial fraud detection in IoT environments." *Available at SSRN 5665670* (2019).

[21] Takkalapally, D., & Takkellapally, M. R. (2023). AdaptCacheAI: Adaptive Hybrid Caching with Machine-Learned Eviction for Dynamic Cloud Workloads. International Journal of Emerging Research in Engineering and Technology, 4(1), 165-174. https://doi.org/10.63282/3050-922X.IJERET-V4I1P118.

[22] Allenki, S. S., & Lee, N. (2022). Performance Tuning Cloud-Hosted Databases: Resource Allocation & Query Optimization. International Journal of AI, BigData, Computational and Management Studies, 3(4), 152-163. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I4P116.

[23] Verma, Jyoti. "Application of machine learning for fraud detection–a decision support system in the insurance sector." (2022).

[24] Nagaraju, Jajam, and J. Vijaya. "Methodologies used for customer churn detection in customer relationship management." *2021 international conference on technological advancements and innovations (ICTAI)*. IEEE, 2021.

[25] Gaddam, R. R. (2022). Cost-Aware Autoscaling for Batch vs. Online Inference. International Journal of Emerging Trends in Computer Science and Information Technology, 3(4), 134-143. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I4P113.

[26] Shiramalla, R. (2022). Design of a Unified API Interface Using Workato for Cross-Platform Data Orchestration between Salesforce and Oracle ERP. International Journal of Emerging Trends in Computer Science and Information Technology, 3(1), 157-168. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I1P118.

[27] Damrongsakmethee, Thitimanan, and Victor-Emil Neagoe. "Data mining and machine learning for financial analysis." *Indian journal of science and technology* 10.39 (2017): 1-7.

[28] Parakala, A. (2021). Building Analytics-Driven Bots: RPA Meets Business Intelligence. International Journal of Emerging Research in Engineering and Technology, 2(1), 77-87. https://doi.org/10.63282/3050-922X.IJERET-V2I1P109.

[29] Srigadde, Bapu Rao. "When Rounding Up Matters: Working With Decimals in Apex". *International Journal of AI, BigData, Computational and Management Studies*, vol. 2, no. 1, Mar. 2021, pp. 122-31, https://doi.org/10.63282/3050-9416.IJAIBDCMS-V2I1P113.

[30] Atri, Preyaa. "Optimizing financial services through advanced data engineering: a framework for enhanced efficiency and customer satisfaction." *International Journal of Science and Research (IJSR)* 7.12 (2018): 1593-1596.

[31] Katangoori, S., & Deore, S. (2022). Edge-Cloud Hybrid Data Pipelines: Architectures for Federated Analytics and Learning. American International Journal of Computer Science and Technology, 4(3), 20-34. https://doi.org/10.63282/3117-5481/AIJCST-V4I3P103.

[32] Kate, Prateek, Vadlamani Ravi, and Akhilesh Gangwar. "Fingan: Generative adversarial network for analytical customer relationship management in banking and insurance." *arXiv preprint arXiv:2201.11486* (2022).