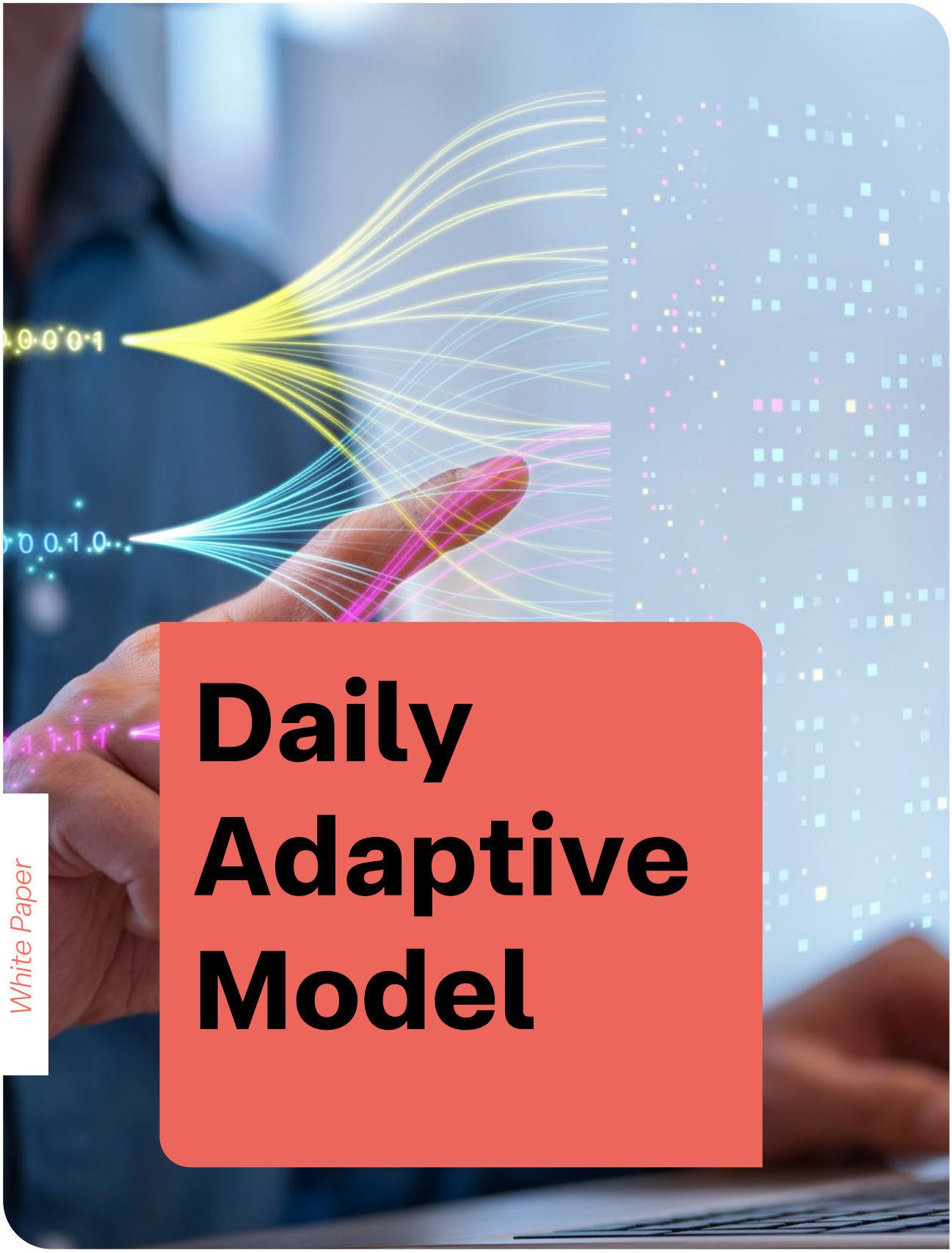
Lynx

AI Against Fraud & Financial Crime



Daily Adaptive Model

White Paper

www.lynxtech.com

Introduction

Digital transactions are constantly changing as companies develop new products and services, payment methods evolve, and users change their behaviors. At the same time, new fraud patterns continuously emerge as criminals adapt their attacks to bypass fraud prevention efforts.

Machine learning (ML) models used in fraud prevention must be updated frequently to capture these dynamic factors and catch the most fraud possible. They also must be able to ingest new data types to account for emerging products, technologies, and payment types relevant to fraud. Models that don't keep up with the shifting transaction and fraud environment drift and degrade over time.

Lynx's Daily Adaptive Models (DAMs) combine daily retraining and data extensibility to drive increased fraud savings and reduced friction for financial institutions. This white paper describes how Lynx updates its ML models daily to prevent more fraud than static ML models commonly used in fraud prevention. The benefits of incorporating dynamic payloads with **Lynx Flex** (which introduces the ability to configure API and intelligence feeds in a user interface to provide data extensibility that propagates to models, rules, and reports) are also discussed. First, however, essential concepts in machine learning and static models will be explored.

A Static ML Model Training Procedure

Many financial institutions and fraud prevention solutions rely upon static ML models to detect fraud. Static models are trained once and then used for a long period of time in production. This legacy approach is limited given the fast-paced changes in transaction fraud. To understand why, it is important to first review how static models are built, trained, and deployed.

Transaction Payloads

ML models are built (trained and tested) using historical transaction payloads from a financial institution's payment systems. ***Figure 1*** is a simplified example of a card transaction payload and shows the fields involved in a card operation such as the transaction type, the date, the time, the card number, the amount, and whether the transaction has been accepted. There is also a label* indicating if the transaction was fraudulent or genuine (red or green, respectively). Fraud and Genuine labels are derived from financial institutions' analysts who determined whether the transactions were fraudulent or genuine.

Sample Transaction Payload

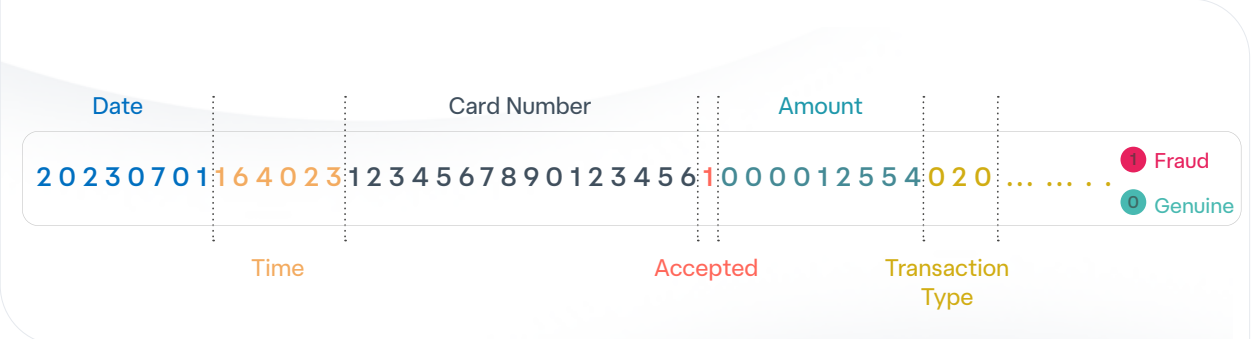


Figure 1

Learn More

* The use of labels in ML model training is referred to as Supervised learning, while training without labels is referred to as Unsupervised learning. For a deeper discussion of this distinction and the benefits of Supervised Learning in fraud detection, see the section titled 'Supervised vs. Unsupervised Learning' in Lynx's white paper Detecting Fraud in Payment Systems with Machine Learning.

Training Sets and Test Sets

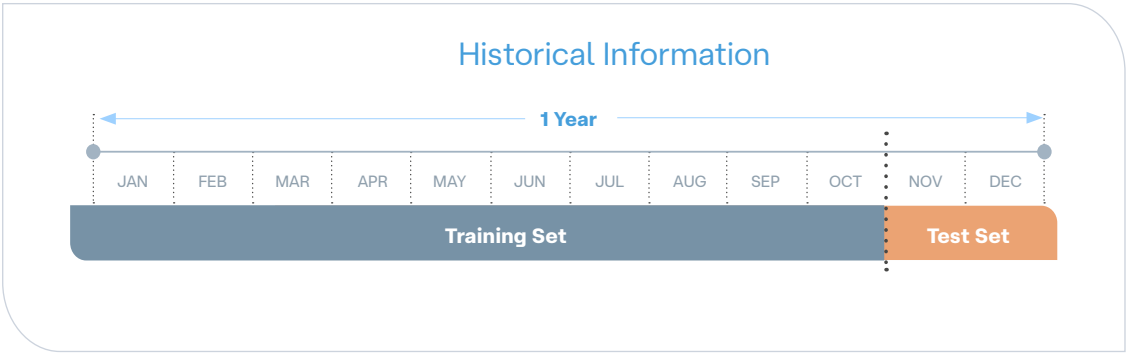


Figure 2

The available historical transactions are divided into sets of data for model training and testing. The transactions within this time period are divided into two sets of data: a training set which will be used to build the model and a test set which will be used to measure its performance.

Figure 2 shows a financial institution's historical transactions over a one-year period. In this case, the training set consists of all transactions from January through October while the test set consists of all transactions from November through December. The specific time periods and data selected for each set can vary, but model building always uses both a training set and a test set, and the training set will always contain older data than the test set. The training and test sets are deliberately separated to maintain objectivity: Once it is built, the model's performance will be measured against data it has not ingested during training.

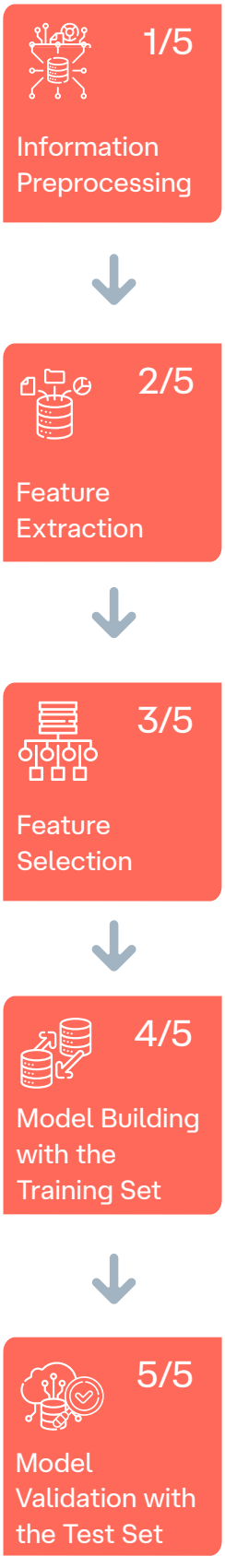


Figure 3

Now that transaction payloads and the development of training and test data sets have been described, the model building process can be explored. ***Figure 3*** provides a high-level overview of the typical process for building static ML models.*

Learn More

* For an in-depth discussion of ML model construction and validation, refer to Lynx's white paper Detecting Fraud in Payment Systems with Machine Learning.

Information Preprocessing

The preprocessing phase involves cleaning transaction data and parsing transactions into fields to prepare the data for ML model ingestion during training. This is an important opportunity to identify issues with the available data, as low quality or unreliable data (such as data fields that are incomplete or in the wrong format) can negatively affect model accuracy if used in training. Both the training and test data sets are preprocessed.

As previously shown, ***Figure 1*** displays a segment of a preprocessed transaction that has been transformed into numbers and separated into fields (date, time, card number, amount etc.).

Feature Extraction

Once transaction information has been preprocessed, feature extraction is performed to transform the original transactions into a set of features that characterize the transactions. Feature extraction can be likened to a game of "Guess Who" where a feature extraction algorithm asks many questions about the data to create new features.

For example, has the customer ever made a transfer to this beneficiary? What is the typical transaction value? How many transfers do they make per day, week, or month? How many devices do they use? Has the device changed? Thousands of these kinds of questions are asked to extract thousands or tens of thousands of new features from the transaction data. An example of an extracted feature might be "number of transactions in the last 2 hours."

After feature extraction, each transaction is represented by a new set of features along with its Fraud or Genuine label, as shown in ***Figure 4.*** Since the sets of features describe the transactions and whether they were fraudulent, they can be used to train the ML model to effectively detect fraud. Both new extracted features and transformed original features (like "date," "time," and "amount") are used in model training.

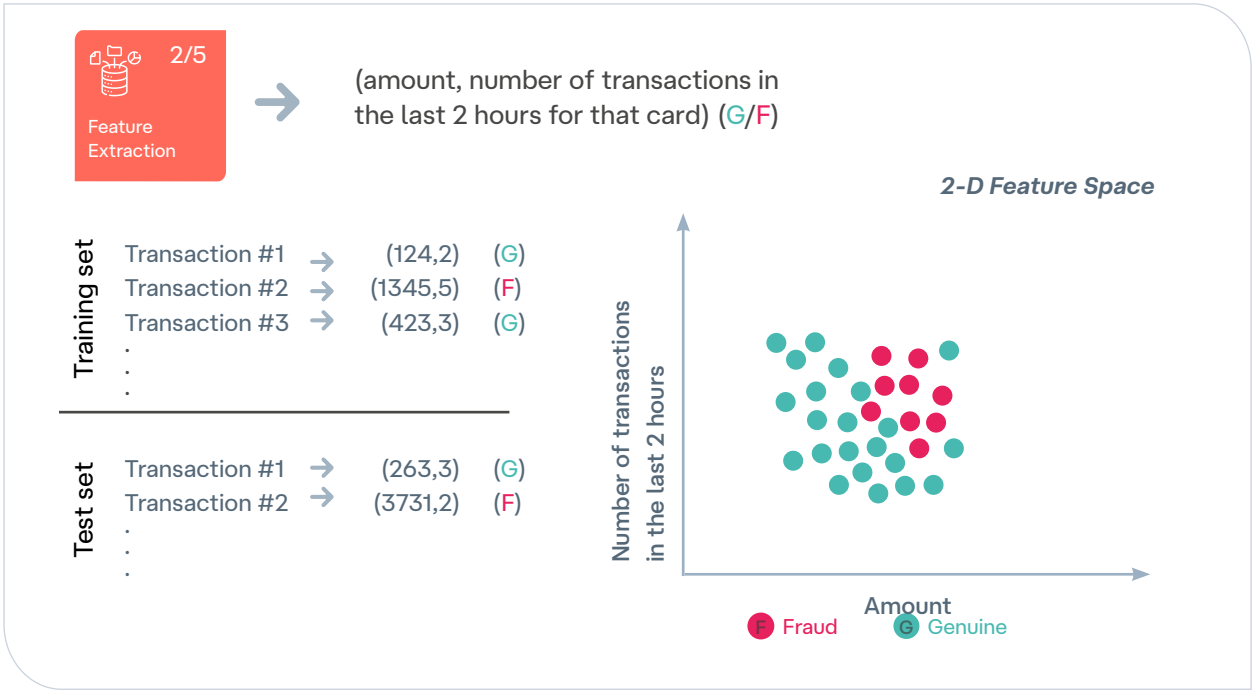


Figure 4

Feature Selection

In general, hundreds, thousands, or tens of thousands of features may be generated during feature extraction. A subsequent feature selection phase is typically performed with feature selection algorithms which choose features that are most relevant for identifying fraud and eliminate those that are redundant or irrelevant.

Feature selection processes, although costly upfront, can significantly reduce dimensionality, accelerating subsequent model training and reducing computational costs. However, these processes may also introduce unwanted bias into the model, as selecting and eliminating features prior to training influences which features the model trains on and which it doesn't.

An alternative approach is to allow the model's training algorithm itself to ingest all extracted and transformed features and select those that are relevant to the problem. Lynx has developed a fast and efficient training algorithm which chooses the most important features to minimize dimensionality, reduce training costs, and significantly reduce bias. This improves model performance over time and allows the model to adapt to changing fraud patterns. This is explained further below in the section titled "Daily Training with a Dynamic Payload".

Model Building

Once features have been extracted and selected, they are ingested by the model during training. The ML model essentially builds logic around the extracted and selected features to solve the problem at hand-detecting fraud.

The model's training algorithm is applied to the training data set (composed of the extracted and selected features) to calculate the model parameters (internal model variables learned from the training data that allow the model to make predictions). Various types of training algorithms are available for use, ranging from artificial neural networks to ensembles of decision trees.

Model Validation

Once the model has been trained, its performance is evaluated against the test data set which was not used during model training. Model performance is measured by how well the model identifies fraud via the risk scores it assigns to each transaction (indicating the likelihood of fraud) versus how many false positives it generates, as shown in ***Figure 5***. This is a crucial step which ensures that the model can accurately identify fraud in new data it hasn't seen before.

To determine if the model's performance is the best possible outcome, the cycle of feature extraction, feature selection, model training, and model validation is then repeated many times with different combinations of features and training algorithms. Finally, the best performing model is deployed in the live production environment where it scores incoming transactions for fraud over the next several months (typically 6 months).

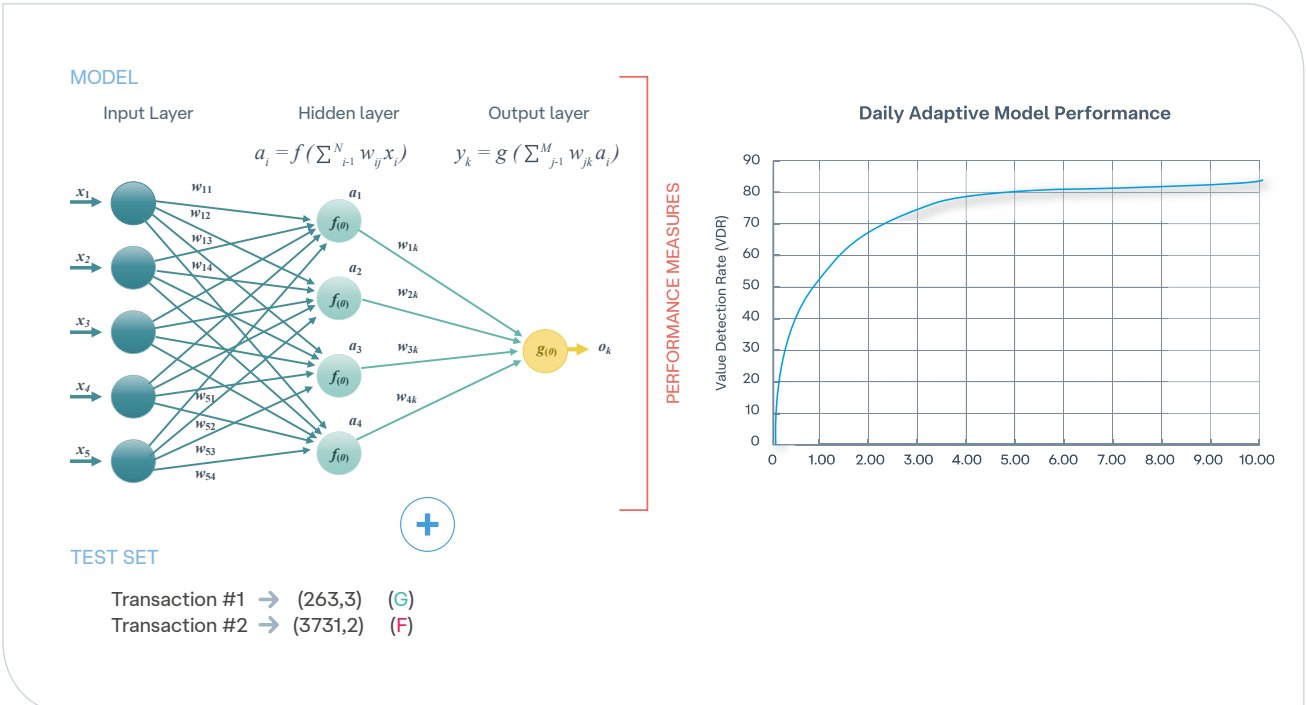


Figure 5

Retraining

After deployment, the static model building process begins again using data from new transactions that have occurred since the model was last trained. ***Figure 6*** shows a typical timeline for the static machine learning model retraining process.

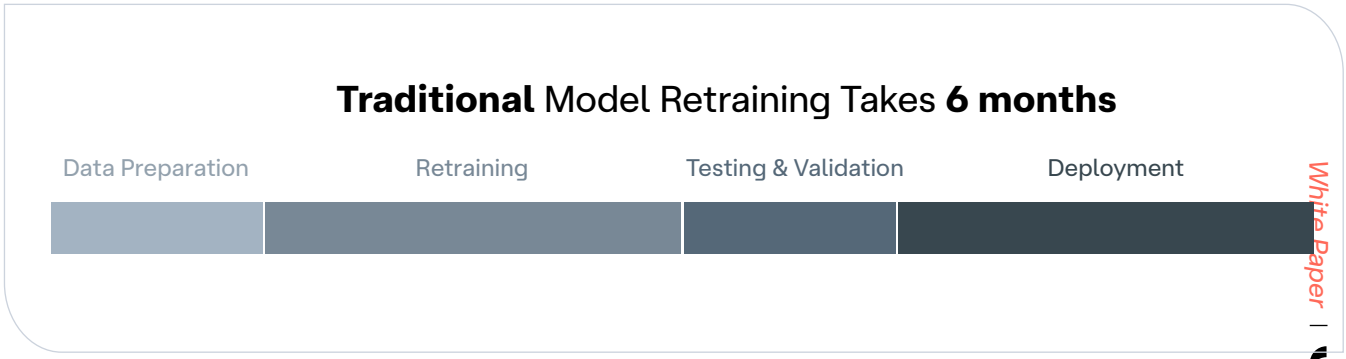


Figure 6

Limitations of Static Training in a Dynamic Fraud and Transaction Environment

Humans and technologies are fundamentally dynamic and customer transaction behaviors, technologies, and fraudster attack patterns constantly change. For the purposes of fraud detection, it is unreasonable to expect that transaction data from a specific time period will continue to be representative of all past and future transactions.

However, this is precisely what static ML models do. As these models remain in production unchanged for months at a time, the transaction data they evaluate becomes more and more differentiated from the

training data they were built around. Inevitably, the models drift and deteriorate as they await retraining, resulting in:

- Unidentified new and emergent fraud cases, which lead to losses that compound as criminals double down and exploit model weaknesses
- Lower detection accuracy, allowing more successful fraud and creating more false positives (genuine transactions scored as fraudulent)
- Increased operational costs to manage fraud impacts and false positive alerts
- Diminished user experience as false positives incorrectly block genuine activities
- Increased customer turnover as customers become frustrated with a poor user experience
- Brand and reputational damage due to more cases of fraud and more false positives

A Daily ML Model Training Procedure

Lynx has developed daily adaptive models that retrain every day on new data obtained from transactions and reported fraud.

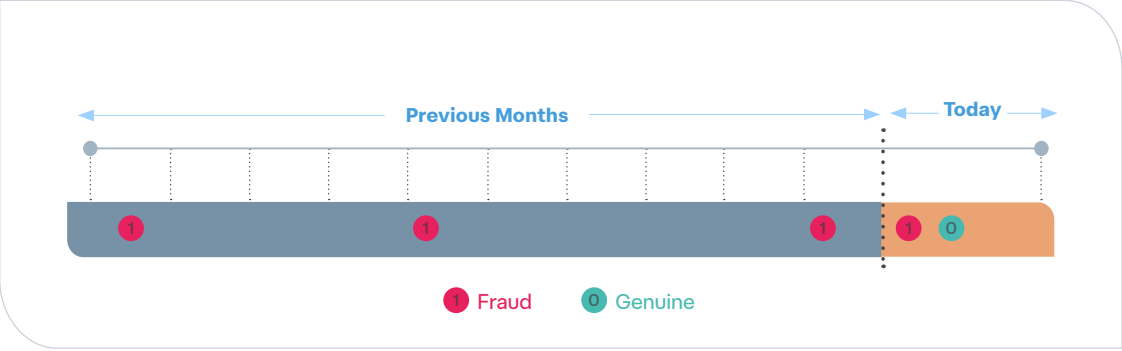


Figure 7

Figure 8

Figure 7 shows how Lynx's DAMs receive transactions and reported fraud cases on a daily basis. Notice that the new fraud labels may correspond to transactions made today (like the red fraud case on the far right of Figure 7), but also (and more commonly) correspond to transactions made days, weeks, or months ago. Additionally, Lynx adds intelligence through feeder data (customer onboarding, application, and identity verification data such as the customer's salary and travel frequency) to enrich transaction payload data, creating additional features and a more complete picture of each customer's financial behavior around current and prior transactions.

Figure 8 shows the retraining procedure that Lynx's DAMs follow each day.

After preprocessing, features are extracted from the new transaction and fraud data (feature extraction). Next, Lynx retrains the models using these new features. As discussed previously, Lynx's training algorithm selects features that are relevant for identifying fraud-contrasting with the legacy approach that uses a feature selection algorithm prior to training and adds undesired bias. Critically, this minimizes dimensionality, reduces training costs, and significantly reduces bias, while improving model performance over time and ensuring model adaptability as fraud patterns evolve. Lynx also utilizes dropout, a regularization procedure which increases model robustness and generalization to new unseen data.

Since the training algorithm itself selects features that are based upon new transaction data, greater importance may be assigned to features that were previously discarded if they are now more relevant to cases of fraud. The model then adjusts its parameters accordingly. This approach ensures the model selects features that are representative of both historic and recent transaction activities and captures all current and previously reported fraud. Lynx's daily retraining thus creates a virtuous cycle to maintain model performance using both new and old features and fraud labels.

As a final step, the updated model's fraud risk scores are assessed against the distribution of scores produced by the previous day´s model. This ensures that performance has not decreased and verifies that the model has been updated properly to combat drift, bias, improper score distribution, and other issues that can arise from new transaction and fraud data. The retrained model is then deployed to production. The entire retraining process takes just hours.

The benefits of daily training compared to static training include:

- Higher model detection accuracy and less drift, preventing more fraud and generating fewer false positives
- Lower operational costs to manage fraud impacts, incident recovery, and false positive alerts
- Reduced alert fatigue, which promotes greater talent retention, lower turnover, and lower recruitment and training costs
- Improved user experience due to fewer genuine transactions being blocked
- Lower customer turnover thanks to an improved user experience
- Improved brand image and reputation due to better fraud prevention with less unnecessary friction

Daily Training with a Dynamic Payload

Most static and daily ML models are limited to analyzing transactions that match the original transaction payload type they were trained on. These models are unable to adapt to new types of data and payment channels (and therefore generate associated features) which may be relevant to preventing fraud. This is an issue in the dynamic payment system environment, where new data fields and payment channels often emerge as technologies and products evolve.

Lynx solves these issues with Lynx Flex, which enables DAMs to train with dynamic payloads.

Figure 9 shows a transaction with a new field: Card-present / card-not-present (CP/CNP) Flag. In this example, assume that the CP/CNP Flag field has become available because the bank providing the transaction data recently switched to a new banking application, enabling more fields to be provided.

Lynx Flex allows the new data field (CP/CNP Flag) to be incorporated throughout the entire model construction and testing procedure, including the preprocessed transaction payload, the extracted features, model training, model validation, and retraining. Lynx Flex also propagates new relevant features throughout Lynx Fraud Prevention's workflows, dashboards, and reports so financial institutions gain greater visibility and are better equipped to prevent fraud.

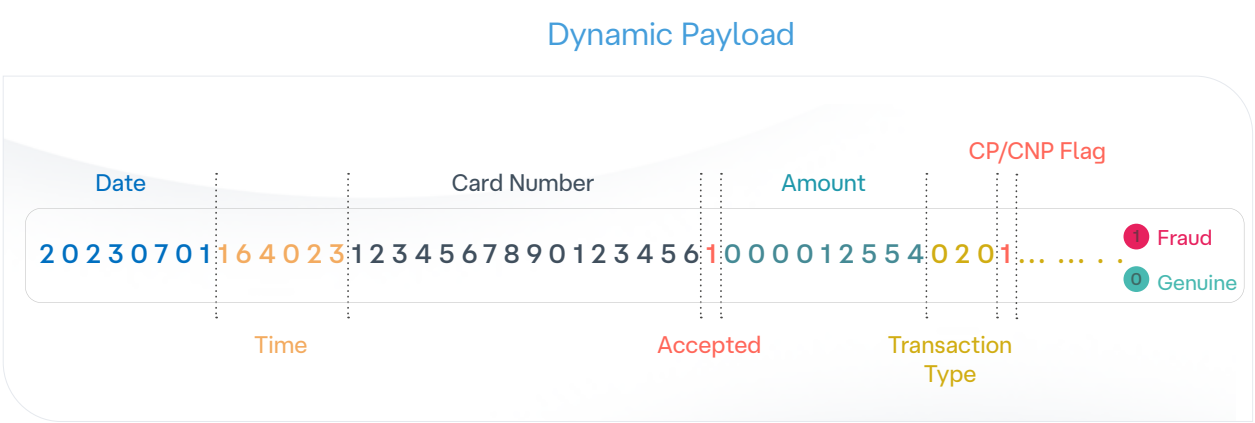


Figure 9

This extensibility ensures that Lynx's DAMs learn and improve from any new data field or type. As new data is added, new features are generated, and the DAMs retrain to enable optimal fraud detection. This significantly improves performance and prevents more fraud, reduces false positives, and enables more genuine and frictionless customer journeys compared to models relying on non-dynamic payloads.

Lynx's Daily Adaptive Models Outperform Static Models

Lynx's Daily Adaptive Models are a pillar-stone for any financial institution. Unlike static and inflexible machine learning (ML) models which drift, deteriorate, and can't learn from new payment methods or data fields, DAMs meet the changing transaction and fraud environment to remain relevant each day.

The proof is in the outcomes Lynx has driven for its customers, which include:

- The ability to identify and thwart new fraud attacks
- Automatic adaptations to significant data drift, such as during the COVID-19 pandemic
- Increased money mule account detection (Account Detection Rate) for the same False to Positive Ratio
- Increased fraud/scam detection (Account Detection Rate) for the same False to Positive Ratio
- Decreased False to Positive Ratio basis points for the same fraud/scam savings
- Increased monthly savings up to millions of GBP.
- Fraud savings for Tier 1 bank

Find out more about how Lynx's DAMs drive value for organizations and their customers below:

Lynx Daily Adaptive Model

Differentiators

Lynx's Daily Adaptive Models (DAMs) are built for **real-time** fraud prevention.

DAMs are architected to **efficiently calculate** tens of thousands of features and **evaluate** thousands of rules.

Lynx upholds strict **code discipline** and develops in a language optimized for production.

DAMs **automatically** generate **new features** and adapt to new rules.

Lynx has refined its techniques over two decades to train models that **handle highly imbalanced datasets.**

Lynx's algorithms and libraries are specifically designed to **address the problem of fraud.**

Retraining the Model **Daily**

Stop more fraud and increase savings.

Stay ahead with a proactive defense against evolving fraud tactics.

Enhance user experience, by reducing the likelihood of legitimate transactions being flagged as fraudulent.

Mitigate alert fatigue among analysts by providing more accurate alerts, allowing them to focus on genuine threats.

Reduce recovery costs associated with fraud incidents, improving overall operational efficiency.

Retain talent by providing higher job satisfaction through fewer false positives, thereby reducing recruitment and training costs.

Scale effortlessly to business growth and transaction volume changes without increasing operational costs.

About Lynx

Lynx utilizes advanced AI for fraud prevention, honed over 25 years. Originating from the Autonomous University of Madrid data science program, Lynx is trusted by leading financial institutions globally to significantly reduce fraud related losses. Processing over 69 billion transactions annually, Lynx's AI-driven approach illuminates real-time risks and empowers organizations to focus on crucial tasks.

Contact Us

Get in touch with us:

Website: lynxtech.com

Email: info@lynxtech.com

Chartis Financial Crime and Compliance50 2024

Chartis RiskTech Quadrant® Category Leader Enterprise Fraud Solutions, 2024

Chartis RiskTech Quadrant® Category Leader Payment Fraud Solutions, 2024

Chartis RiskTech Quadrant® Best of Breed Name and Transaction Screening Solution, 2024