Artificial Intelligence in Credit Card Fraud Transactions Detection

FRAUDSTER
DETECTIVE

843 million

*Number of credit card transactions in **Hong Kong** throughout 2020.*

25.6 billion

Expected monetary amount (USD) lost globally due to payment card fraud in 2020.

0.53 %

Average global credit card fraud rate in 2014.

Source: Hong Kong Monetary Authority, Intuit, MRC 2015 Global Survey

8-15%

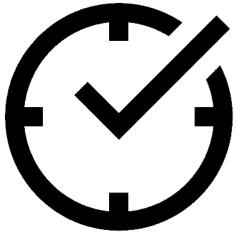
*Share of credit card transactions that were **manually screened** in 2014 in Top 10 EU countries.*

Source: MRC 2015 Global Survey

Why are we doing this?

People

The exhaustive human resources needed to conduct manual screening



Time

The enormous amount of time needed to screen credit card transactions

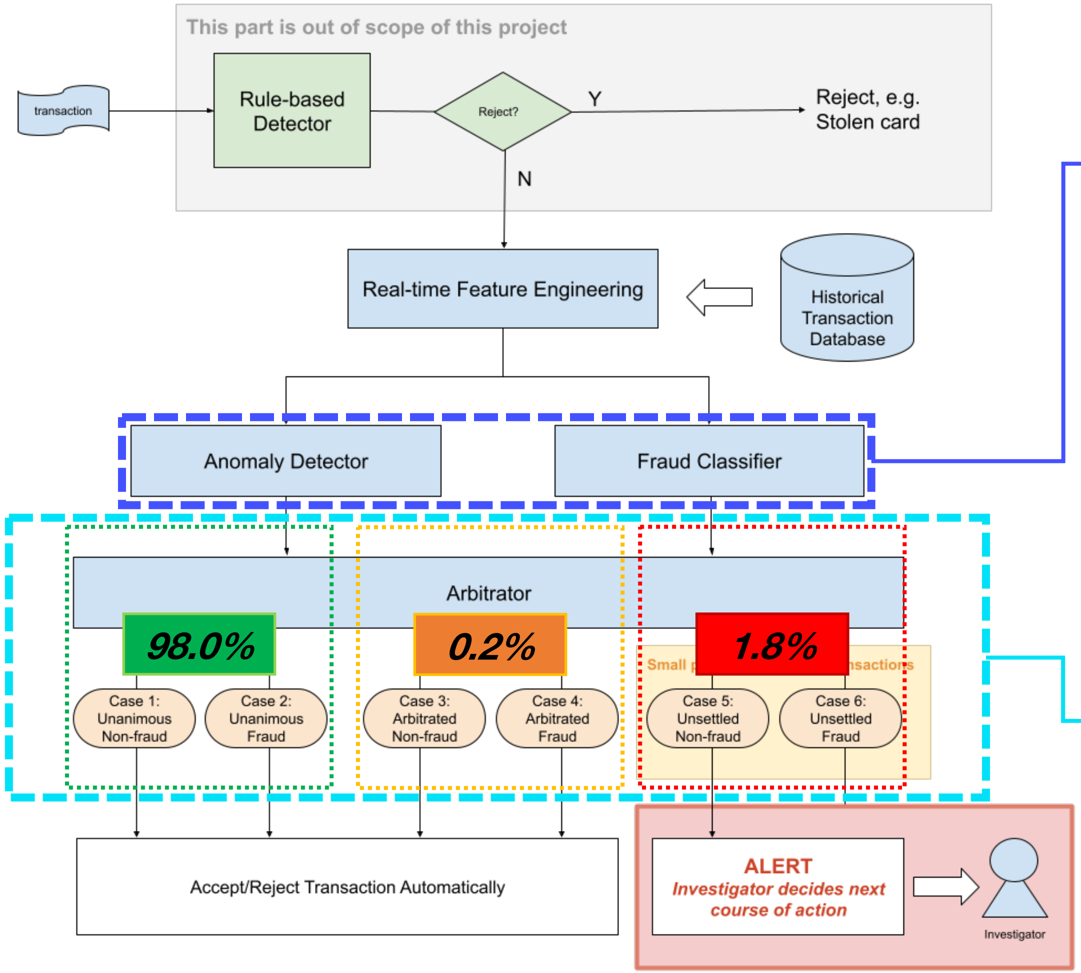
Cost

The monetary cost pressure on both merchant and financial institution sides

OPPORTUNITY COST

Dual-model detector system

How the dual system works



The dual-model system

1. The **Anomaly Detector** classifies a transaction into one of two (2) outcomes: ***0*** (genuine) or ***1*** (fraud)
2. The **Fraud Classifier** outputs a ***probability*** of a transaction being a fraud

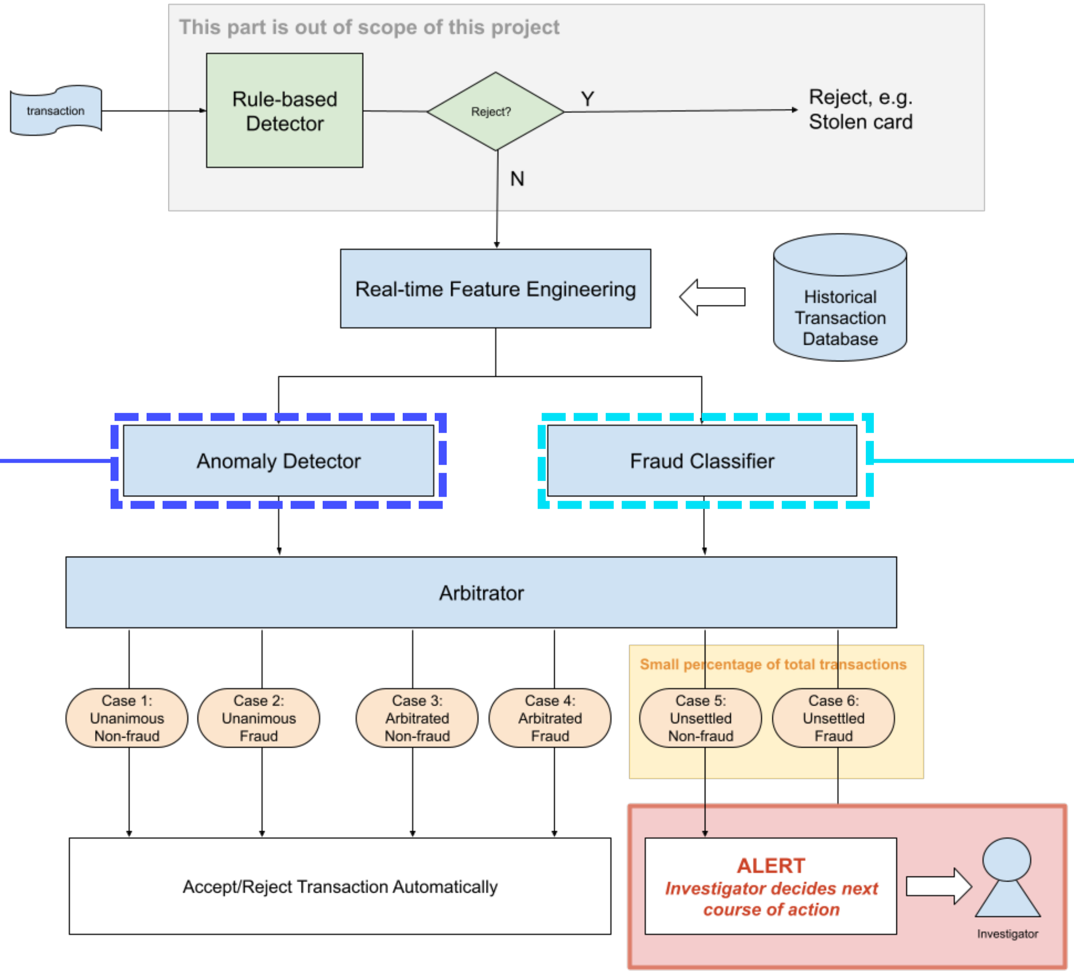
The arbitrator system

There are three (3) possible outcomes:

- **Unanimous** Both models classify a transaction as either genuine or fraud
- **Arbitrated** Both models output different results, but upon arbitration process, if the probability of a transaction being a fraud is still within preset threshold tolerance, the ***fraud classifier*** will agree with the ***anomaly detector***'s output
- **Unsettled** Both models output different results despite arbitration process, resulting in the transaction being forwarded for manual screening

Dual-model detector system

Anomaly Detector vs Fraud Classifier



One-class Support Vector Machines (SVM)

Precision 0.43
Recall 0.89

The dual-model system

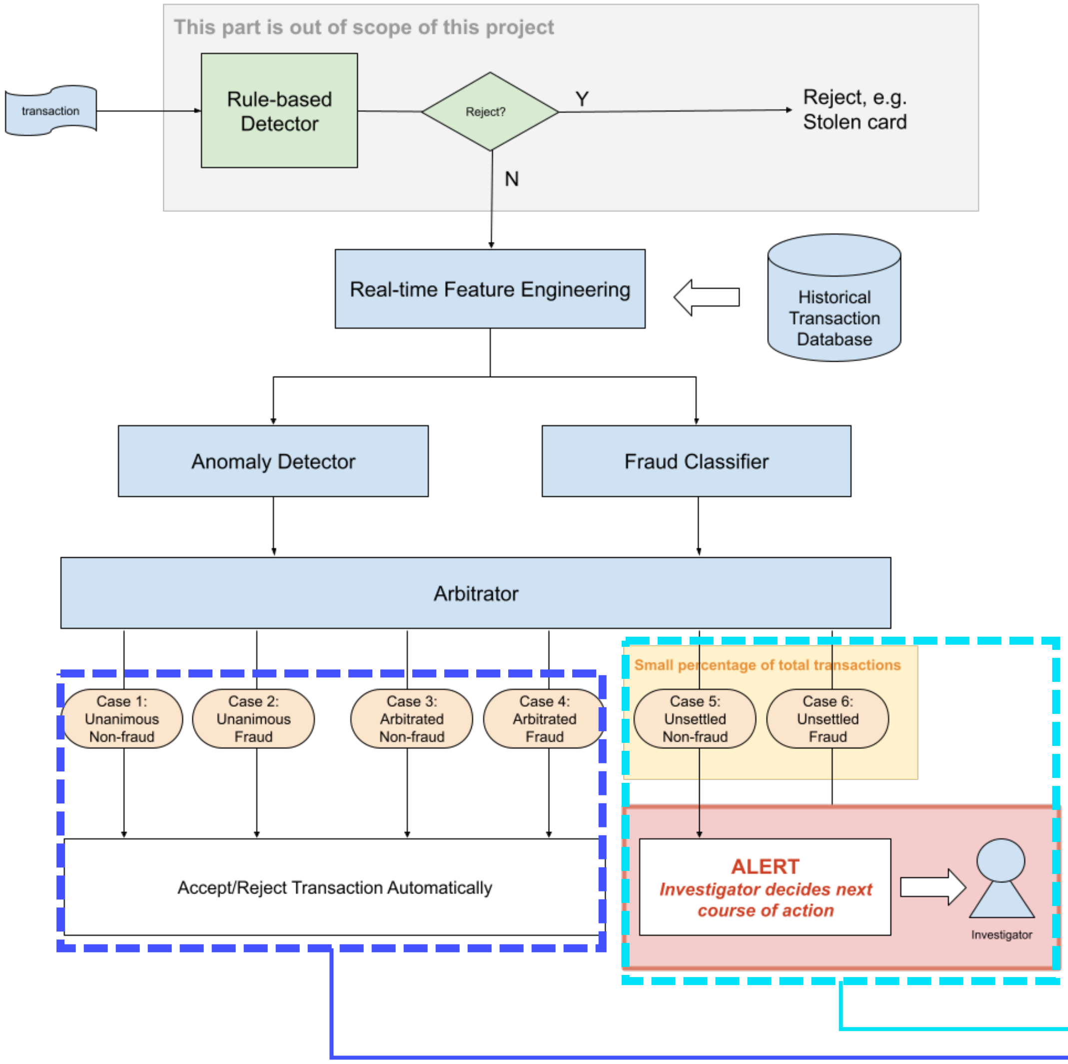
Precision 0.88
Recall 0.87

XGBoost Classifier

Precision 0.94
Recall 0.85

Dual-model detector system

Why adding value to an already-existing system?



To help companies allocate their ***human***, ***time***, *and* ***financial*** *resources more effectively and efficiently in aspects that truly matter by:*

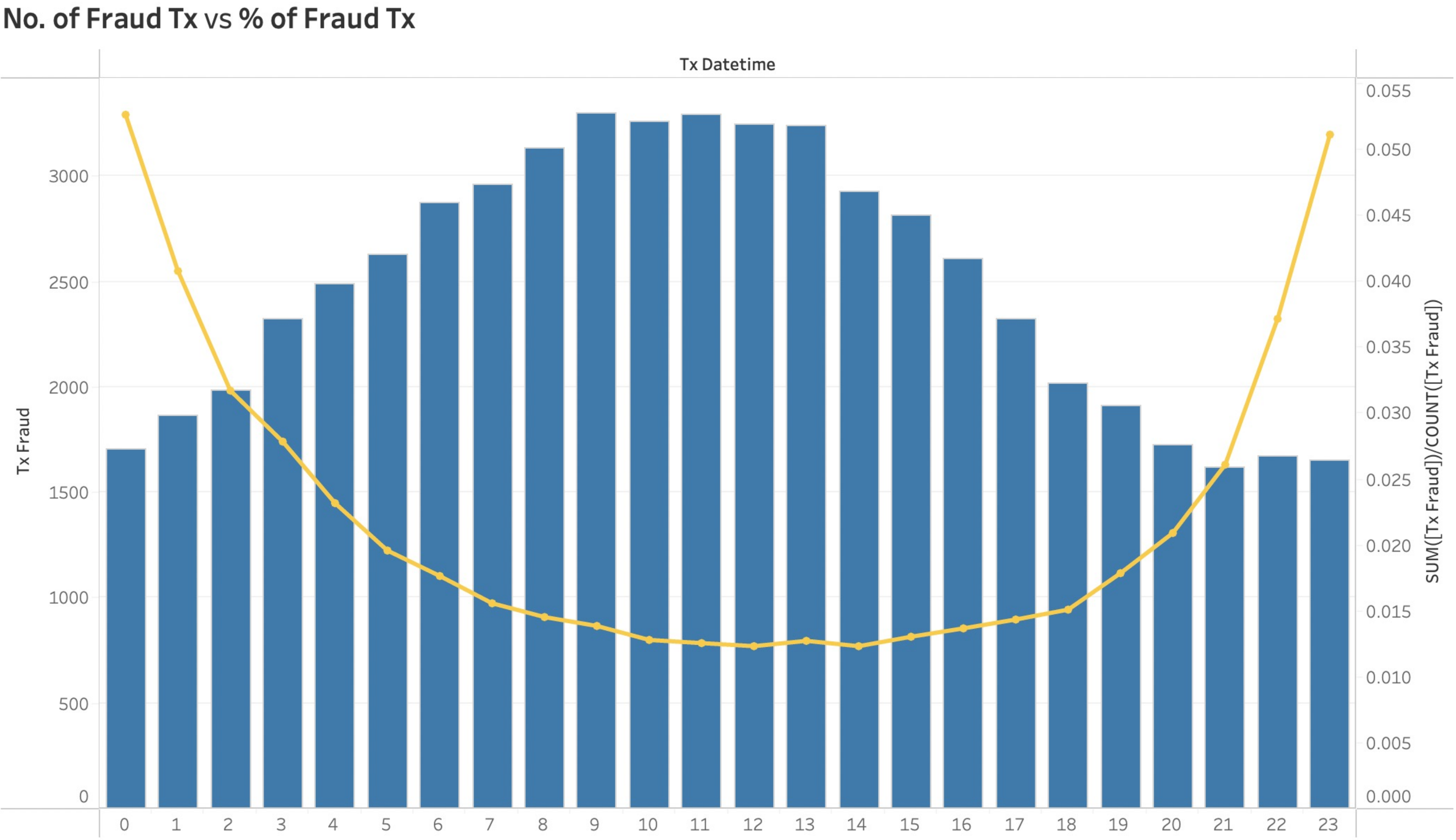
Minimising unsettled predictions

Maximising unanimous and arbitrated predictions

Overview
Value-Add
Selling Point
Demonstration
Evaluation

Why our dual-model detector system?

Realistic data



The number of transactions, both ***fraud*** and ***genuine***, peaks and valleys based on the time of the day.

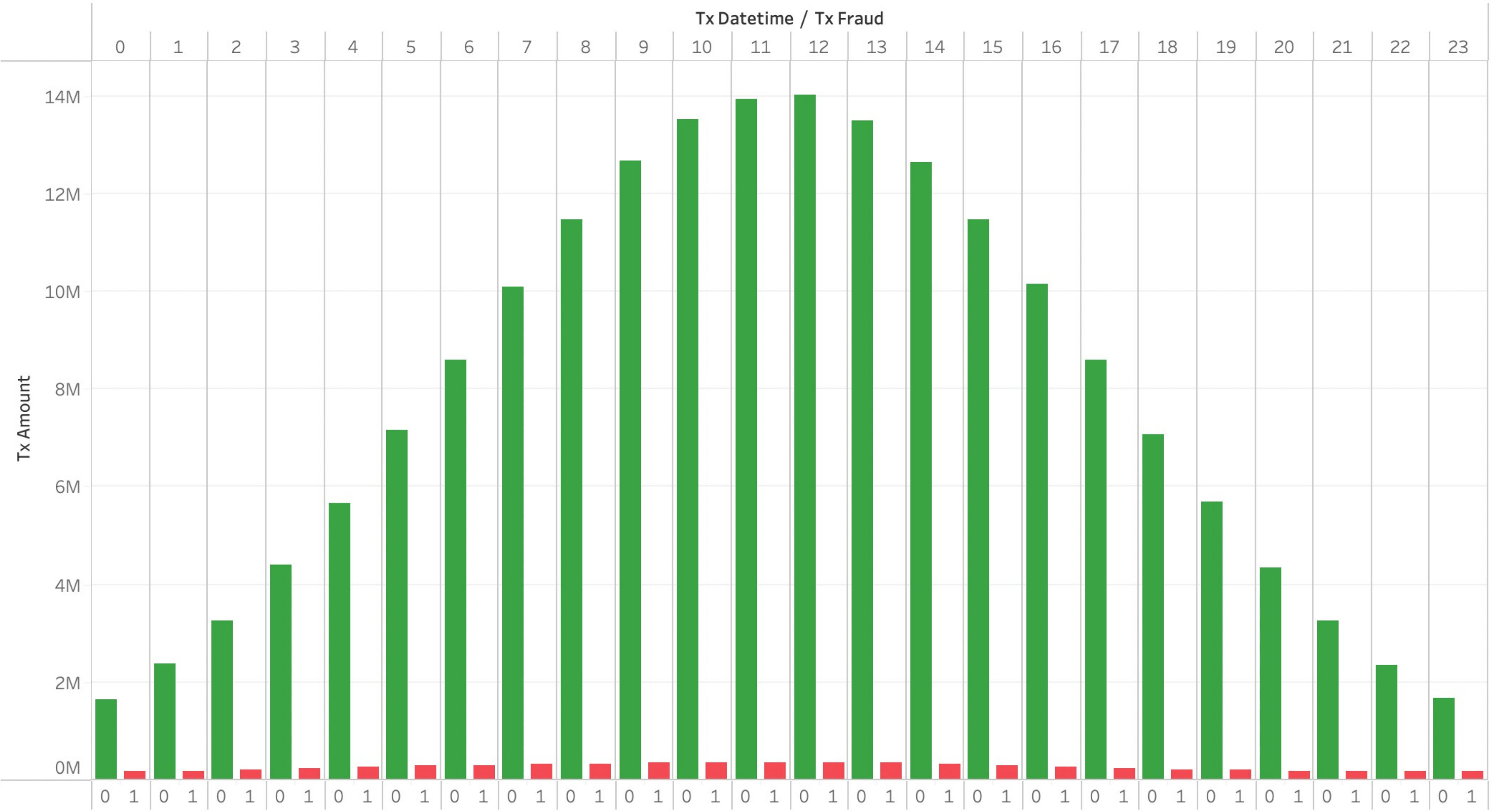
Our dataset simulates a scenario where ***fraud*** transactions are more likely to occur during nighttime, particularly from 21:00 to 03:00.

Our data simulates a realistic scenario in which credit card transactions occur when customers go about their days.

Why our dual-model detector system?

Realistic data

Hours vs Total Spending (USD)

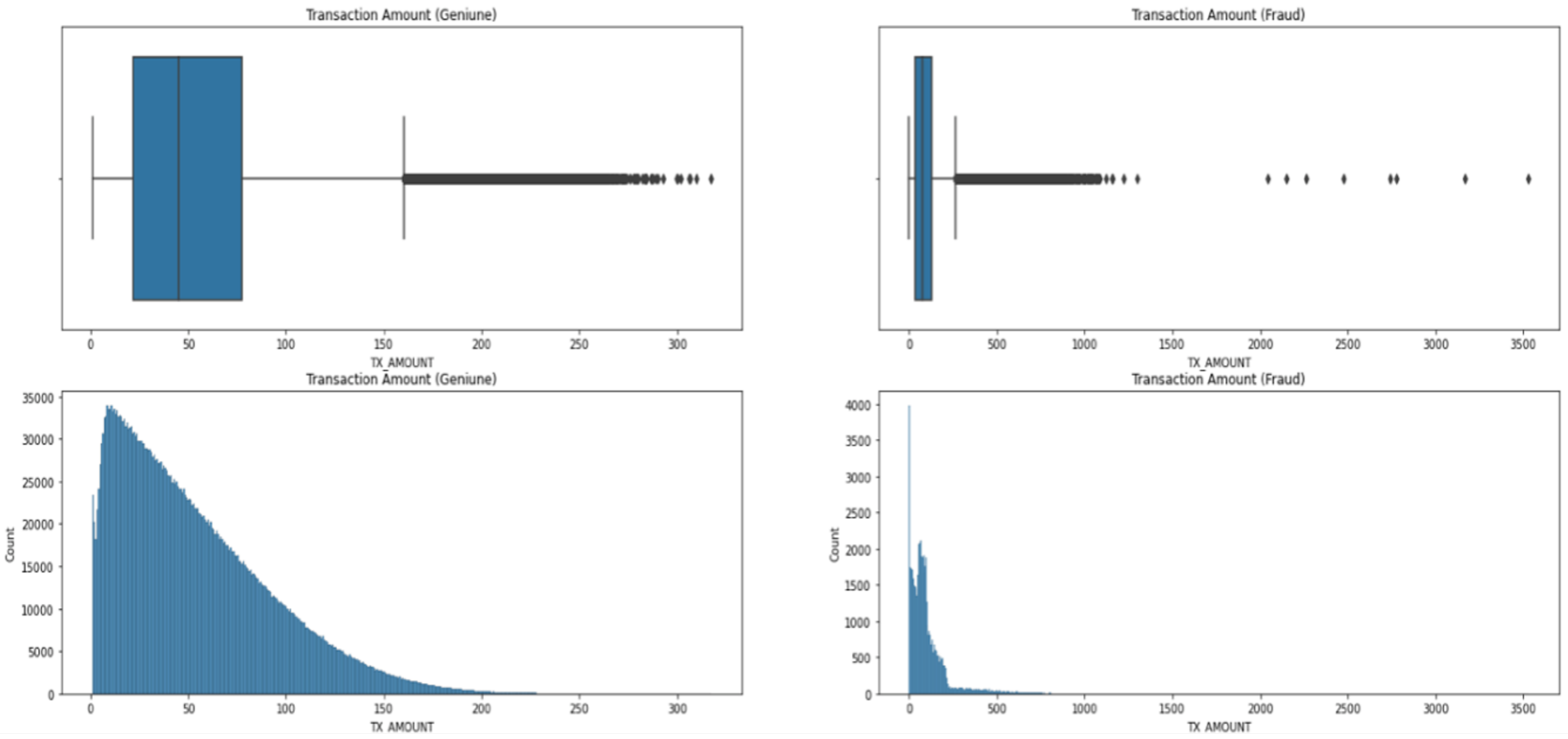


Total spending of all transactions, both ***fraud*** and ***genuine***, is realistically distributed across the day.

Total amount under ***fraud*** classification is notably higher during nighttime compared to that of during daytime.

Why our dual-model detector system?

Realistic data

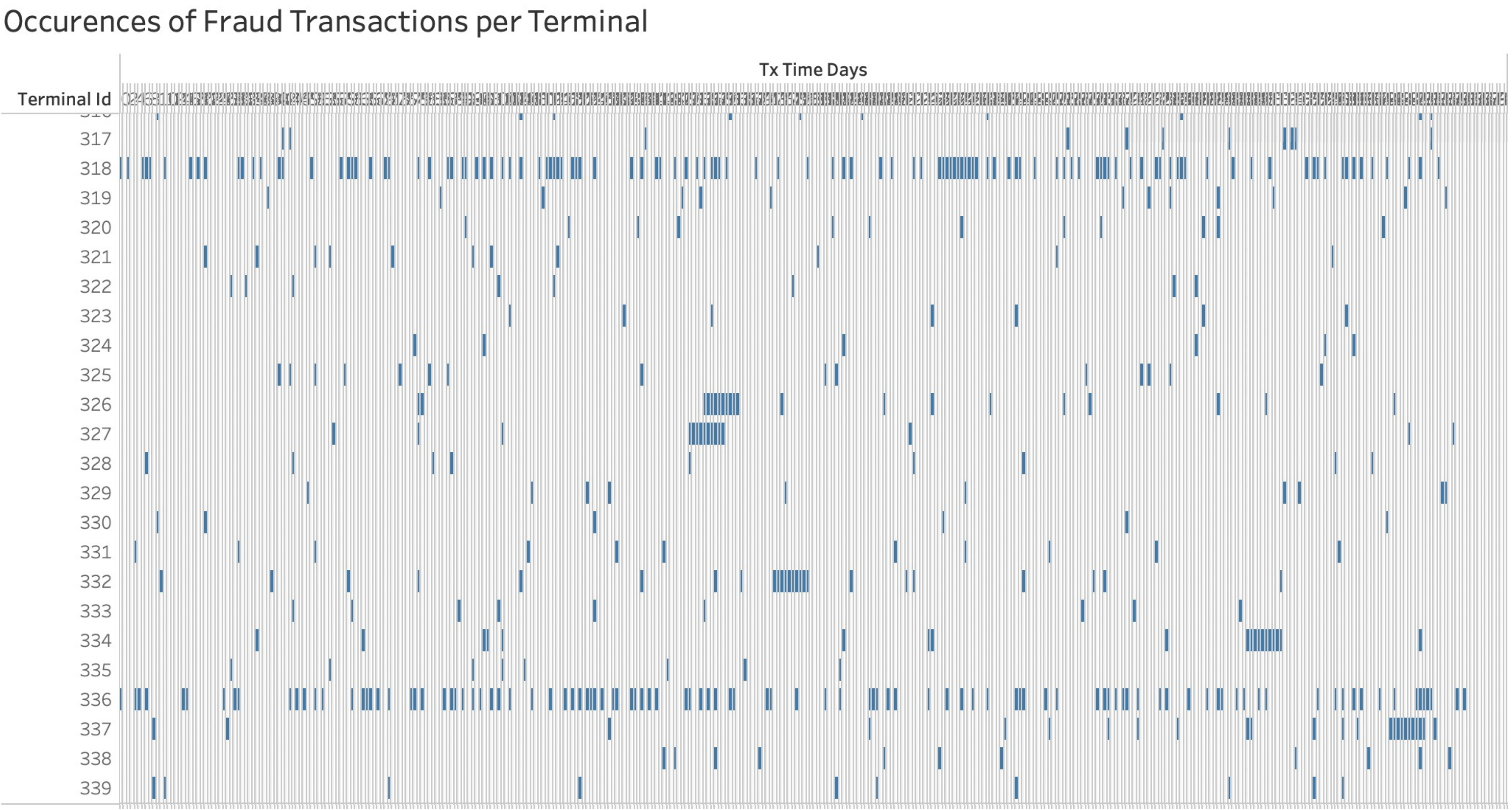


Amount of ***fraud*** transactions has larger range compared to that of ***genuine*** transactions.

Amount of ***fraud*** transactions range from $0 to $500 whereas that of ***genuine*** transactions range from $0 to $200.

Why our **dual-model** detector system?

Realistic data



Occurrences of ***fraud*** transactions are randomly distributed across all 2,500 terminals throughout the year.

There are three (3) realistic patterns of ***fraud*** when looked from *per terminal* basis:

- Sporadic
- Cluster
- Consistent

Why our dual-model detector system?

Feature Engineering

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3579099 entries, 0 to 3579098
Data columns (total 32 columns):
 #   Column                                      Dtype
---  ------                                      -----
 0   TRANSACTION_ID                              int64
 1   TX_DATETIME                                 datetime64[ns]
 2   CUSTOMER_ID                                 int64
 3   TERMINAL_ID                                 int64
 4   TX_AMOUNT                                   float64
 5   TX_TIME_SECONDS                             int64
 6   TX_TIME_DAYS                                int64
 7   TX_FRAUD                                    int64
 8   TX_FRAUD_SCENARIO                           int64
 9   TX_LAST_DATETIME                            object
 10  TX_LAST_SECONDS                             float64
 11  TX_LAST_HOURS                               float64
 12  TX_LAST_DAYS                                float64
 13  TX_TIME_HOUR_BIN_0                          int64
 14  TX_TIME_HOUR_BIN_1                          int64
 15  TX_TIME_HOUR_BIN_2                          int64
 16  TX_TIME_HOUR_BIN_3                          int64
 17  TX_TIME_HOUR_BIN_4                          int64
 18  TX_TIME_HOUR_BIN_5                          int64
 19  CUSTOMER_ID_AVG_AMOUNT_1DAY_WINDOW          float64
 20  CUSTOMER_ID_AVG_AMOUNT_7DAY_WINDOW          float64
 21  CUSTOMER_ID_AVG_AMOUNT_30DAY_WINDOW         float64
 22  CUSTOMER_ID_AVG_AMOUNT_2REC                 float64
 23  CUSTOMER_ID_AVG_AMOUNT_10REC                float64
 24  TERMINAL_ID_RISK_2DAY_WINDOW                float64
 25  TERMINAL_ID_RISK_7DAY_WINDOW                float64
 26  SUM_TX_AMOUNT_CUSOMTER_ID_SAME_TERMINAL_SAME_DAY  float64
 27  nb_TX_CUSOMTER_ID_SAME_TERMINAL_SAME_DAY    int64
 28  ONE_DOLLAR                                  int64
 29  CUSTOMER_TERMINAL_DISTANCE                  float64
 30  AMOUNT_Z_SCORE                              float64
 31  CUSTOMER_TERMINAL_DISTANCE_Z_SCORE          float64
dtypes: datetime64[ns](1), float64(15), int64(15), object(1)
memory usage: 873.8+ MB
```

Simulated Data

Data extracted directly from the generated simulation data.

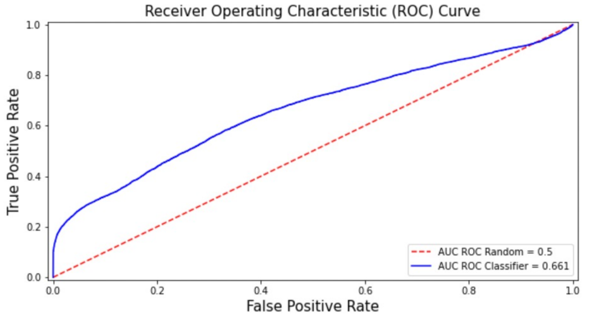
Feature engineering

Data specifically generated to help the model understand better the spending behaviour of each customer, allowing the system to differentiate transactions that ***deviate*** *from* ***normal*** or ***abnormal*** from the ***genuine*** ones despite possibly similar patterns.

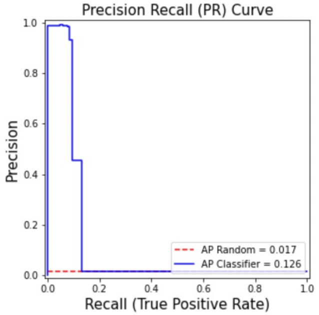
Why our dual-model detector system?

Feature Engineering

WITHOUT feature engineering



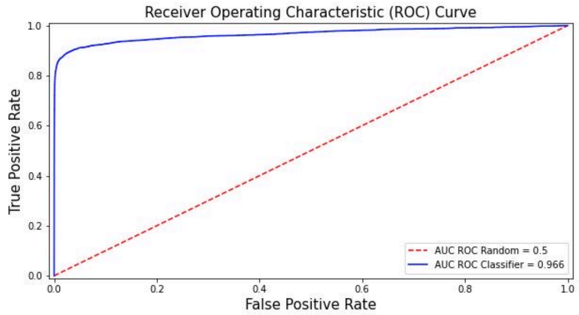

```
<ipython-input-6-454dce6eda95>:136: UserWarning: color is redundantly defi
  ax.step(recalls, precisions, 'b', color='blue', label = 'AP Classifier =
```

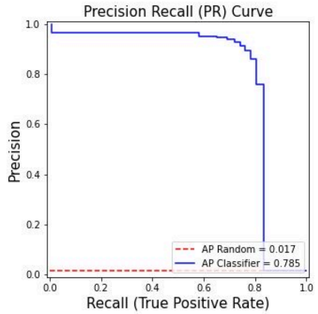


Classification report for threshold = 0.5 (for reference only)

	precision	recall	f1-score	support
0	0.98	1.00	0.99	298780
1	0.99	0.07	0.13	5094
accuracy			0.98	303874
macro avg	0.99	0.53	0.56	303874
weighted avg	0.98	0.98	0.98	303874

WITH feature engineering





Classification report for threshold = 0.5 (for reference only)

	precision	recall	f1-score	support
0	1.00	1.00	1.00	298780
1	0.93	0.74	0.82	5094
accuracy			0.99	303874
macro avg	0.96	0.87	0.91	303874
weighted avg	0.99	0.99	0.99	303874

With **feature engineering**, while *precision* for ***fraud*** transactions falls from 0.99 to 0.93, *recall* for the same transactions increases drastically from 0.07 to 0.74.

Feature engineering allows our model to identify many ***fraud*** transactions by applying the correctly predicted labels. The high *recall* rate means the model is able to differentiate between the real ***fraud*** transactions from the ***false negative*** ones.

Why our dual-model detector system?

Result

Probable Existing System

(as of 2014)

8-15%

Dual-model System

```
Classifier threshold 0.4        Arbitrate threshold 0.6        Unsettled default cl
* Classifier, threshold: 0.4
[4312, 269]
[782, 298511]
Precision 0.94          Recall 0.85
* Anomaly Detector
[4538, 6018]
[556, 292762]
Precision 0.43          Recall 0.89
** Dual Model Detector - Using Classifier for unsettled cases
[4420, 626]
[674, 298154]
Precision 0.88          Recall 0.87
                  unanimous  arbitrated  unsettled
DUAL_ARBITRATION     297711         623       5540
Percentage            97.97        0.21       1.82
```

~ 2.4 million transactions
1.7% fraud rate

1.8-2.0%

How does the system work?

Advantages

Extra level of confidence

Transactions are predicted ***unanimously*** or through ***arbitration*** process by the dual system.

Less wasted resources

Manual screeners (investigators) can focus solely on identifying ***unsettled*** transactions, hence allowing institutions to allocate resources more efficiently.

Double references

Both ***anomaly*** detector and fraud ***classifier*** provide double anchors for investigators when screening ***unsettled*** transactions and dealing with those particular customers.

Disadvantage

Customer behaviour

The model remains having difficulty in capturing the spending ***characteristics*** and/or ***patterns*** over time.

Addressing disadvantages of our system

Concept Drift

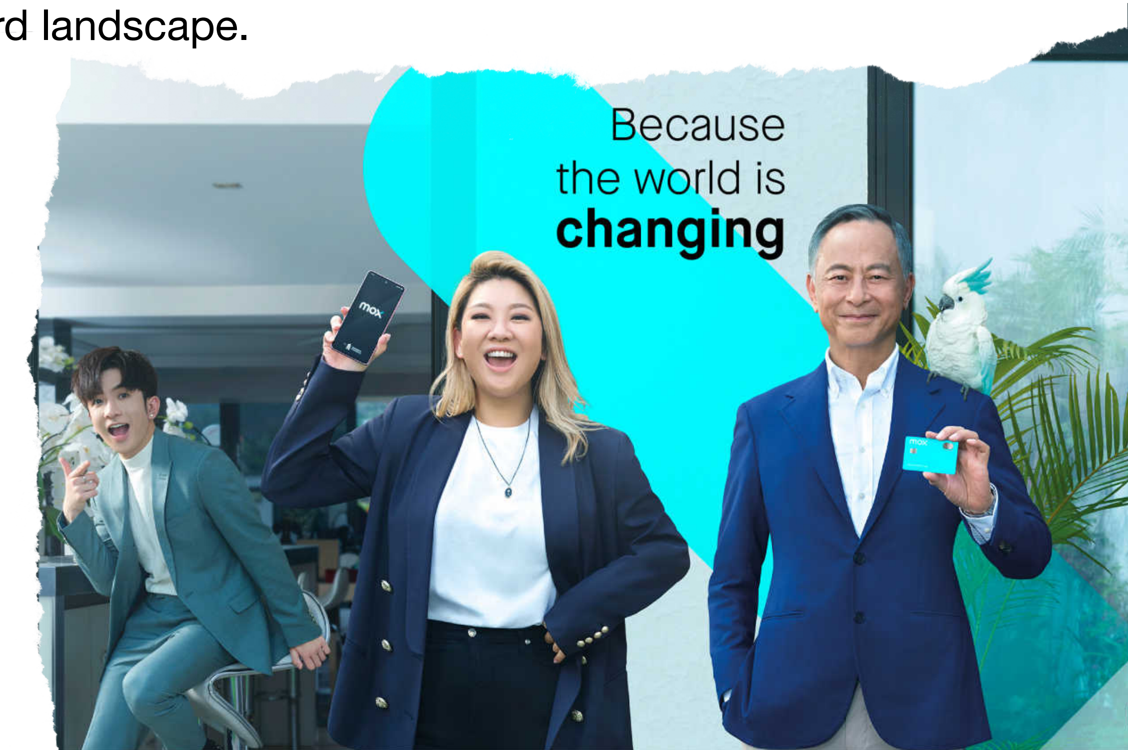
Concept Drift

Transaction data changes over time. It's undeniable that eventually the system will be met with degrading predictive performance of the ***dual model system*** as – at current time – our system assumes static relationship between input and output variables.

In our case, transactions may vary based on seasonality, evolving technology, macroeconomic growth, and other external factors that may affect the credit card landscape.

Possible Solutions

- Constantly update the system by ***exploiting only*** the recently supervised samples and ***removing*** the outdated information that's deemed no longer significant.
- Implement a system that trains itself regularly.



Addressing disadvantages of our system

Delayed Reporting

Delayed Reporting

There are delays involved between the time the ***transaction occurs*** and the fraud is ***reported***. For example, a fraud transaction might not be classified as fraud instantly, but rather at a later point of time (e.g. a week later, a month later, etc.).

Possible Solutions

- Regularly update the model on a periodic basis (e.g. weekly, monthly, quarterly, etc.) using only data from certain sets of historical data, such as only the previous 4 quarters.
- Apply ***sliding-window approach*** where classifier is retrained every day on the most recent supervised samples.
- Apply ***ensemble approach*** where a new component replaces the oldest one in the ensemble every single day.

Addressing disadvantages of our system

Verification Latency

Verification Latency

A delay in obtaining accurate labels of transactions (***fraud*** or ***genuine***) caused by factors such as limited human resources. Before the manual screener even manages to touch the transaction or put the correct labels, the customers might have already reported the transaction as ***fraud***.

Accurate transaction label is obtained only after the manual screeners contact the cardholders.

Possible Solutions

- Apply ***sliding-window approach*** where classifier is retrained every day on the most recent supervised samples.
- Apply ***ensemble approach*** where a new component replaces the oldest one in the ensemble every single day.

Looking **further**

Into *Cybersecurity*

In general

Address the ***technical*** and ***operational*** aspects of the retraining model

Human-guided machine learning system to accelerate learning process

For our dual-model system

Separate feature engineering for the anomaly detector model

Try ***different pairings*** of supervised and unsupervised learning algorithms

Try different ***arbitration logic*** and **parameters**

THANKS!

TIME FOR Q&A NOW