Fraud Detection in Mobile Payment System

Ashish Jain
M.Tech Computer Science and Engineering
IIIT Delhi
ashish18052@iiitd.ac.in

Shubham Gupta
M.Tech Computer Science and Engineering
IIIT Delhi
shubham18055@iiitd.ac.in

I. Problem Statement

The project aims to detect fraud detection in synthetic datasets generated by the PaySim mobile money simulator. It is a binary classification problem. The two classes are fraud and not fraud. The labels are present in the used dataset. Therefore we treated this problem as a supervised learning classification problem.

Fraud in digital banking is increasing rapidly with the era of modern technology. To catch fraudsters, methodologies for the detection of fraud is essential to prevent risk of loss. There are sufficient proofs which shows that Machine learning models can be taken into account to tackle this kind of problem.

II. Literature Review

The domain of fraud detection problem mostly has skewed datasets. Skewness results to make our model biased. [4] propose a novel data balancing technique called SMOTE. This technique generates synthetic minority observations by using k-nearest neighbors. [3] detect fraud in the payment transaction by applying various ML techniques like LinearSVM logistic Regression, etc. [7]shows how deep learning models can help to identify fraudulent transactions with better results. They had used autoencoders.

Different evaluation metrics had been proposed for evaluating the model's performance on unbalanced data. Accuracy is not such a useful metric when data is uneven. [6]describe various evaluation metrics that can help in assessing the results of our models like Matthew Correlation Coefficient (MCC) [8] uses the metric AUPRC rather than AUROC as the latter is more sensitive towards the parameter settings of different algorithms . [5] reviews various techniques for handling data level and algorithm level methods for handling imbalance like oversampling and undersampling.

III. Database details

Money transactions details contain various confidential information. Since there is a scarcity of available public datasets on money transactions primarily in the domain of mobile payment system. Therefore we are using synthetic dataset generated by a mobile money simulator PaySim [2]. The dataset is obtained from Kaggle. The dataset contains more than 6 million transactions each having 11 attributes. The dataset is highly skewed as shown in Fig.1 The details of the attributes are mentioned in Table I on Page 1.

TABLE I
Dataset Features Description

Features	Description
Step	It maps a unit of time in the real world.In this case 1 step is 1 hour of time. Total steps 744 (30 days simulation)
Type	CASH-IN, CASH-OUT, DEBIT, PAYMENT and TRANSFER
Amount	amount of the transaction in local currency
NameOrig	customer who started the transaction
OldBalanceOrg	initial balance before the transaction
NewBalanceOrg	new balance after the transaction
NameDest	customer who is the recipient of the transaction
OldBalanceDest	initial balance recipient before the transaction.
NewBalanceDest	new balance recipient after the transaction.
IsFraud	Fraud, Not Fraud (1, 0)
IsFlaggedFraud	Marked if transaction attempt to transfer more than 200,000 in a single transaction

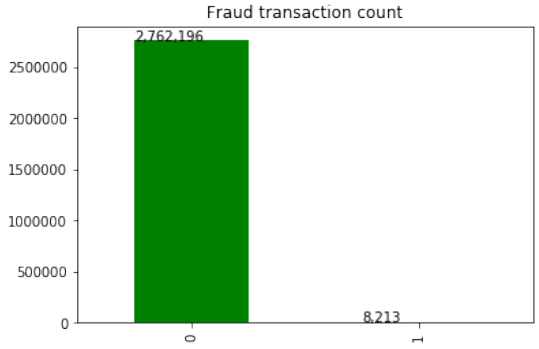

Fig. 1. Visualization of Unbalanced Data

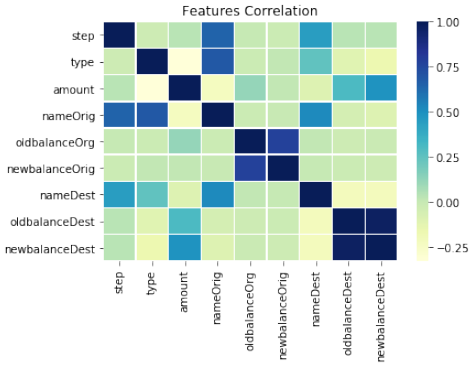

Fig. 2. Correlation among features

IV. Proposed Architecture

A. Data Preprocessing

The original data from Kaggle contains more than six million samples. We preprocessed the data and reduces the number of samples to 2.7 million according to the observations we made. The observations which lead to data preprocessing and obtaining relevant features are mentioned below:

- The attribute isFlaggedFraud is not useful:
  - It is set only 16 times in 6 million transactions.
  - Whenever isFlaggedFraud is set the attribute isFraud is always set.
  - It is always set in payment Type Transfer. Moreover, the features oldbalanceDest and newbalanceDest are always 0.0 when isFlaggedFraud is set.

  So we decided to discard isFlaggedFraud attribute without loss of any useful information.
- We have five types of payment in the original dataset CASH-IN, CASH-OUT, DEBIT, PAYMENT and TRANSFER. Only two payment types CASH-OUT and TRANSFER out of these five have fraud transactions. The rest three payment types have always isFraud set to 0. So we discarded the samples which have payment type other than CASH-OUT and TRANSFER. Fig.7 tells the ratio of fraud transactions with their payment types.

B. Feature Engineering

- Feature reduction by correlation
  - Some features have high correlation i.e they have some linear relationship between them.
  - From Pair of such features a feature can be dropped to reduce the computation cost and such features do not provide any new information.
  - example age and date of birth.
- Feature expansion by auto encoders
  - As the dataset has very less features, we expand our features by using autoencoders.
  - The encoding dimension in which we transform our feature vector of eight is thirty two. We are taking the output of the dense layer as the expanded feature vector. The scatter plot of our new latent representation is shown in Fig 4

C. Imbalance Data Handling

- Random Under-Sampling
  - This technique tries to balance the data distribution by randomly removing data points from majority class datapoints.
  - This is done repeatedly until both the majority, and minority class instances are balanced.
- SMOTE
  - Synthetic Minority Oversampling (SMOTE) works by generating synthetic samples based upon the actual minority observations.

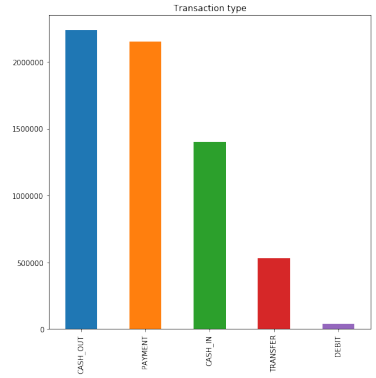

Fig. 3. Visualization of different type of transactions

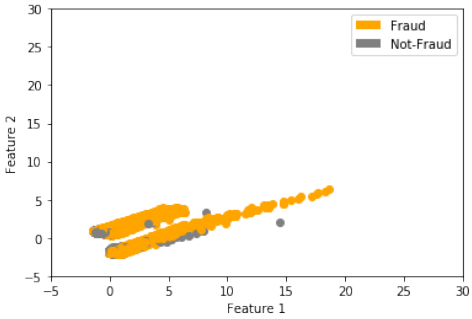

Fig. 4. Scatter Plot autoencoder features.

  - SMOTE generally works better than undersampling or oversampling.
  - The literature shows that SMOTE produces effective results in fraud detection.
  - It calculates K-nearest neighbors .

D. Classification

Different classification models are selected based on specific charateristic of the model.

Classifiers used are:

- Gaussian Naive Bayes
- Logistic Regression
- K nearest Neighbour
- Random Forest

E. Evaluation Metric

Various evaluation metrics are used to calculate the performance of the model. Accuracy is not a good measure for data which is highly unbalanced.

Evaluation metrics used are:

- Confusion Matrix
- Precision-Recall
- Area Under ROC
- Precision Recall Graph
- MCC (Matthews Correlation Coefficient)

MCC and Precision-Recall curve are good evaluation metric for data which is unbalanced [6].

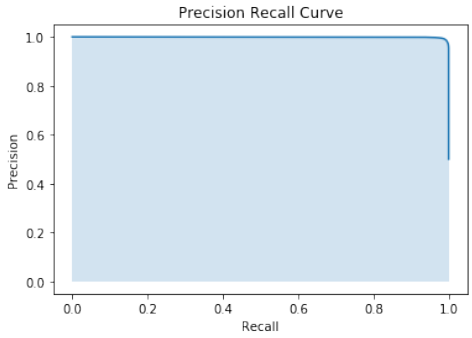

Fig. 5. Precision Recall curve with KNN classifier with SMOTE data balancing technique and feature selection using correlation among features

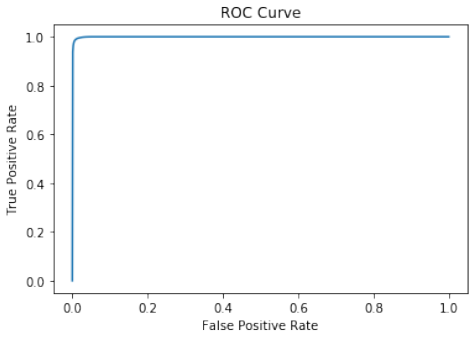

Fig. 6. ROC curve with KNN classifier with SMOTE data balancing technique and feature selection using correlation among features

V. Results

Results using various classifiers are generated on both random under sampling and SMOTE with different feature engineering techniques as discussed in approach. The results are shown in Fig. 8

VI. Results Analysis

- Models on unbalanced data are highly biased.
- Predicting all samples as not fraud results in accuracy of 99.70% because of less number of fraud samlpes.
- SMOTE techniques with KNN produces better MCC and recall of fraud class.

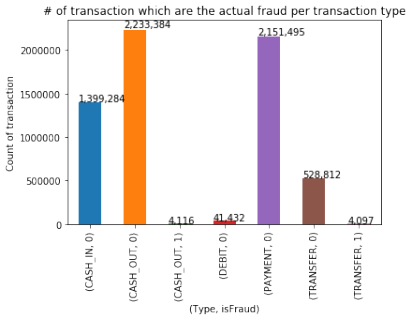

Fig. 7. Visualization of Data across various Transaction type

Data Balancing	Feature Engineering	Evaluation Metrics					
		Classifiers	Precision	Recall	MCC	AUC	Accuracy
Random	Original features	GNB	0.96	0.42	0.49	0.87	0.703
		KNN	0.93	0.87	0.80	0.96	0.901
		LR	0.95	0.85	0.81	0.96	0.903
		RF	0.98	0.99	0.97	0.99	0.983
Under	Correlation Features	GNB	0.93	0.44	0.48	0.86	0.706
		KNN	0.93	0.93	0.85	0.98	0.926
		LR	0.91	0.80	0.72	0.95	0.857
		RF	0.97	1.00	0.96	0.99	0.982
Sampling	Auto Encoder Features	GNB	0.93	0.44	0.48	0.86	0.70
		KNN	0.93	0.93	0.85	0.98	0.92
		LR	0.91	0.80	0.72	0.95	0.85
		RF	0.97	1.00	0.96	0.99	0.982
SMOTE	Original Features	GNB	0.95	0.46	0.51	0.87	0.721
		KNN	0.98	1.00	0.97	0.99	0.988
	Correlation Features	GNB	0.95	0.51	0.544	0.86	0.74
		KNN	0.98	1.00	0.97	0.99	0.98

Fig. 8. Results with Different Evaluation Metrics and Classifiers

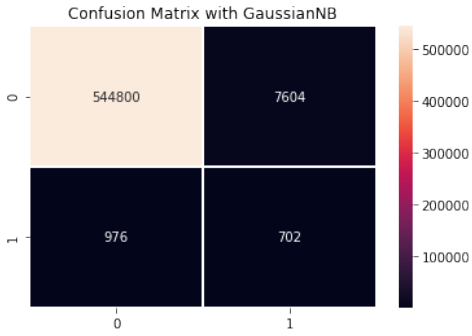

Fig. 9. Confusion Matrix Using Gaussian Naive Bayes (80:20 ratio)

- Since both methods based on nearest neighbors. Other data balancing techniques like ADASYN can be explored in future.

VII. Individual contributions of each group partner

- Ashish Jain
  - Evaluation Metric's implementation
  - Random Under Sampling
  - Data Preprocessing
- Shubham Gupta
  - SMOTE
  - Feature Engineering
  - Classification Model

References

[1] https://www.kaggle.com/ntnu-testimon/paysim1
[2] E. A. Lopez-Rojas , A. Elmir, and S. Axelsson. "PaySim: A financial mobile money simulator for fraud detection". In: The 28th European Modeling and Simulation Symposium-EMSS, Larnaca, Cyprus. 2016.
[3] http://cs229.stanford.edu/proj2018/report/261.pdf

[4] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.
[5] Kotsiantis, Sotiris, Dimitris Kanellopoulos, and Panayiotis Pintelas. "Handling imbalanced datasets: A review." GESTS International Transactions on Computer Science and Engineering 30.1 (2006): 25-36.
[6] Bekkar, Molhamed, Hassiba Kheliouane Djemaa, and Taklit Akrouf Alitouche. "Evaluation measures for models assessment over imbalanced data sets." J Inf Eng Appl 3.10 (2013).
[7] Mubalaike, Aji Mubarek, and Esref Adali. "Deep Learning Approach for Intelligent Financial Fraud Detection System." 2018 3rd International Conference on Computer Science and Engineering (UBMK). IEEE, 2018.
[8] Davis, J., & Goadrich, M. (2006, June). The relationship between Precision-Recall and ROC curves. In Proceedings of the 23rd international conference on Machine learning (pp. 233-240). ACM.