INTEGRATION OF MACHINE LEARNING AND BLOCKCHAIN IN THE BANKING SECTOR

A project report,
submitted in fulfillment of the requirements for B. Tech project

by

Mehul Jain (2018IMT-051)

Under the Supervison of

Dr. Aditya Trivedi

and

Dr. W Wilfred Godfrey

ABV INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT
GWALIOR-474 015

2021

CANDIDATES DECLARATION

I hereby certify that the work, which is being presented in the report, entitled **Integration of Machine Learning and Blockchain in the Banking Sector**, in fulfillment of the requirement for the award of the Degree of **Bachelor of Technology** and submitted to the institution is an authentic record of my own work carried out during the period *June 2021* to *October 2021* under the supervision of **Dr. Aditya Trivedi** and **Dr. W Wilfred Godfrey**. I have also cited the reference about the text(s)/figure(s)/table(s) from where they have been taken.

Mohul Jain

Date: 28.10.2021 Signatures of the Candidate

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

31/10/21

Date: Signatures of the Research Supervisors

ABSTRACT

With the advancement in technologies and introduction to various new domains, we see everyday improvements in the technical realm in all segments of society. The financial sector is also no behind others and has accepted the revolutionizing changes by bringing most of the banking solutions to the customer's doorstep through the online services provided by them.

Although shifting to this new and ever-growing digital space opens up novel opportunities for the banking system, it has also created a rich environment for fraudsters to cheat upon. Financial fraud is an ever-growing menace with severe consequences in the financial industry. Deceitful banking services can cause huge losses to the banking sector, provide discomfort to the users, and further negatively affect the economy.

The integration of Machine Learning and Blockchain can help the financial sector emerge from this trench. This work extends the fraud detection technique and proposes an Light Gradient Boosting based Machine Learning algorithm. Further, blockchain technology is incorporated so as to revert the transaction marked as fraudulent by the algorithm at the source itself, ensuring that the transaction never took place. This paper aims to look upon this problem and suggest a possible solution to it.

Keywords: Financial Fraud, Machine Learning, Blockchain, Light Gradient Boosting

ACKNOWLEDGEMENTS

I am highly indebted to **Dr. Aditya Trivedi** and **Dr. W Wilfred Godfrey**, and obliged for giving me the autonomy of functioning and experimenting with ideas. I would like to take this opportunity to express my profound gratitude to them not only for their academic guidance but also for their personal interest in my project and constant support coupled with confidence boosting and motivating sessions which proved very fruitful and were instrumental in infusing self-assurance and trust within me. The nurturing and blossoming of the present work is mainly due to their valuable guidance, suggestions, astute judgment, constructive criticism and an eye for perfection. My mentors always answered myriad of my doubts with smiling graciousness and prodigious patience, never letting me feel that I am novices by always lending an ear to my views, appreciating and improving them and by giving me a free hand in my project. It's only because of their overwhelming interest and helpful attitude, the present work has attained the stage it has.

Finally, I am grateful to our Institution and colleagues whose constant encouragement served to renew my spirit, refocus my attention and energy and helped my in carrying out this work.

Mehul Jain

(Mehul Jain)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AUC	Area under curve
CIS	Computational Intelligence Society
EFB	Exclusive Feature Bundling
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GBDT	Gradient Boosting Decision Tree
GOSS	Gradient-based One Side Sampling
IEEE	Institute of Electrical and Electronics Engineers
Light GBM	Light Gradient Boosting Machine
ML	Machine Learning
RFECV	Recursive Feature Elimination with Cross Validation
ROC	Receiver Operator Characteristic
TN	True Negative
TP	True Positive
TPR	True Positive Rate

CHAPTER 1

INTRODUCTION

This chapter includes introduction to the growth of fraudulent transaction and the solution proposed in this paper and the motivation behind choosing the field for research.

1.1 INTRODUCTION

With the development of online transactions, modern financial service providers participate in and process billions of transactions every day. These transactions continue to grow exponentially in the financial industry across various platforms worldwide [2]. In recent years, internet purchases have grown drastically, and unfortunately, so has fraud. As a result, the financial service industry and their customer have to bear huge fraud-related losses and damages.

Day by day, the cases of fraudulent transactions in the digital area of the banking sector are increasing. Hence an infrastructure is needed which is capable of detecting suspicious transactions and stop them from being completed.

Several data mining-based approaches for detecting transaction fraud have been developed in recent years [6],[13]. The majority of these studies utilised traditional models like logistic regression [1] and support vector machines [6] to predict the end result. However, the problem persists owing to a lack of feature engineering, such as a scarcity of aggregated or integrated information, and traditional models are unable to handle today's challenges.

Machine learning, a well-known technology for extracting knowledge from massive datasets, is a popular tool for detecting and avoiding financial fraud. Several models have already been developed to solve the problem in hand but these models lack in performance as the accuracy by which these model predict the nature of transaction is not very high. In this paper, we are proposing a LightGBM [7] based fraud detection algorithm. The dataset used for training is from IEEE-CIS competition which contains more than a million samples and greater than 400 feature variables.

Moreover, a Private Permissioned Blockchain would be created which would serve the purpose of reverting the transactions if they are classified as fraudulent, else, it would store the encrypted hash value of the transactional information(which we can use to verify the integrity of the transaction data). Storing the hashed value of the transaction on the blockchain would help us identify the data integrity at any time later by comparing the stored hashed value with the newly generated hash.

1.2 MOTIVATION

Online Transactions have become the most prevalent method of exchange due to new breakthroughs in digital commerce systems and communication technology; as a result, there is significantly increasing fraud connected with such transactions. Fraudulent transactions cost consumers huge financial losses every year. TransUnion found that the percentage of suspected fraudulent digital transaction attempts against businesses originating from India increased 28.32% when between 11 March 2019 and 10 March 2020 compared with 11 March 2020 to 10 March 2021 [5]. With this motivation, we are trying to create a system capable of detecting such transactions that help in protecting the consumer's interest and also of the financial service providers by detecting a transaction to be of fraudulent nature or not. And if a transaction is found to be of fraudulent nature(rather than legitimate), then preventing the completion of such a transaction, by stopping it at the source using blockchain.

1.3 REPORT LAYOUT

In **Chapter 2: Literature Survey and Objectives**, the past work related to detection of fraudulent transactions have been discussed along with the objectives which this paper aims to accomplish.

In **Chapter 3: Design Details and Implementations**, the working and system architecture used in making the project is stated. It also describes the methods,steps and processes that would be followed in achieving the task that we have proposed in this report.

In **Chapter 4: Results and Discussion**, discussion about the tasks completed so far and the results obtained by completing those tasks are stated. The future work that can be done extending the current research is also discussed here.

CHAPTER 2

LITERATURE SURVEY AND OBJECTIVES

In this chapter, work related to Fraudulent Transaction Detection mechanism is discussed and the objectives which are aimed to be accomplished through this paper.

2.1 LITERATURE REVIEW

There have been several work done to detect the nature of online transactions. There are various data mining based approaches that have been already implemented to solve the task of classifying the transaction as fraudulent or legitimate. Data mining is the process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistic and database systems. Using data mining, the objective is to detect and extract the patterns in data that may lead towards a fraudulent transaction. Then these fraud patterns can be fitted in binary classifiers further. In [3] Wenkai Deng et al. used a data mining based approach to classify the transactions.

In [11], Quah et al. focus on four main fraud occasions in real-world transactions and provide a comprehensive strategy to select an optimal algorithm with respect to the type of fraud occasion. They evaluate four popular machine learning algorithms, which are Support Vector Machine, Naive Bayes, K-Nearest Neighbor and Logistic Regression.

Minastireanu[10] presented a Light GBM based fraud detection algorithm to detect fraud click from millions of click actions. In this paper, the predictions were made on the dataset using the features given in the dataset, that is, no attempt was made to find the correlations between different features, by processing features to form an aggregated or combined feature, and improve the prediction accuracy of the model. Processing the features before generating our model can also improve the prediction speed and memory used as there might be some features that are somehow co-related.

Hence, one of the features can be dropped, hence reducing the memory consumed (like having cost of a house in both, Indian Rupee and Dollar is useless).

There are also some recent studies who managed to do some feature engineering on the dataset before using it to generate the prediction model. In [8] Xiong Kewai et al. proposed a deep-learning-based method for fraud detection. Feature engineering, memory compression, mixed precision, and hybrid loss are also used to boost model performance and increase prediction speed.

In [12] Du Shaohui et al. used tree-based Random Forest classifier, which is a classifier with many decision trees, for prediction purposes. In this paper, the first step followed is to eliminate some outliers and excessive missing data. In the further feature engineering cycle, the data is transformed and the statistical data such as maximum, mean and standard deviation is extracted. Then, Recursive feature elimination (RFECV) is used to eliminate some unimportant features. Finally, a binary classifier based on random forest is implemented according to the data and features. However, the metric score achieved by this model was also not at par.

2.2 RESEARCH GAPS AND OBJECTIVES

Many studies have tried to contribute towards the field of Fraud Transaction Detection, but there are some common limitations. Since the information pertaining a transaction is sensitive and difficult to collect, many models have been trained on a small dataset. Due to the small size of the dataset, the model may not be able to distinguish between huge gap in count of transaction belonging to each task. The real world dataset is often skewed and that's why the results from the previous models may not be accurate. Further, many of the works discussed lack Data Preprocessing and Feature Engineering being performed on the dataset. This leads to greater memory consumption, greater prediction time and lesser accuracy.

So, in this paper we are proposing to use techniques of Data Preprocessing (like Memory Compression and Feature Elimination) and Feature Engineering to reduce the memory consumed by such large real world dataset, remove some unwanted or not so important features and to derive some new features from the dataset that could prove to be quite useful in predicting the nature of the transactions. Using these techniques can improve the prediction time as well as accuracy of the proposed model. Further, in this paper, a LightGBM based ML model is proposed which is capable to tackle the skewed nature of classes in the dataset and efficiently classify the transactions.

Further, this paper also proposes the use of a Private Permissioned Blockchain Network to store the hash value of transaction's information. Storing the hashed value of the metadata on an immutable ledger like blockchain ensures the integrity of the details of a transaction are maintained as one may check the value of hash stored in the

blockchain with the newly computed hash value of the metadata of transaction. If the value comes out to be same, then the data have not been tampered with in the database, otherwise, it indicates that the integrity of the data have been compromised.

CHAPTER 3

DESIGN DETAILS AND IMPLEMENTATION

In this chapter we are going to discuss about the overall architecture of our project and about the methods and processes that needs to be followed and the techniques utilized for achieving the objective of our report.

3.1 SYSTEM ARCHITECTURE

This section discuses the overall system architecture of the project.

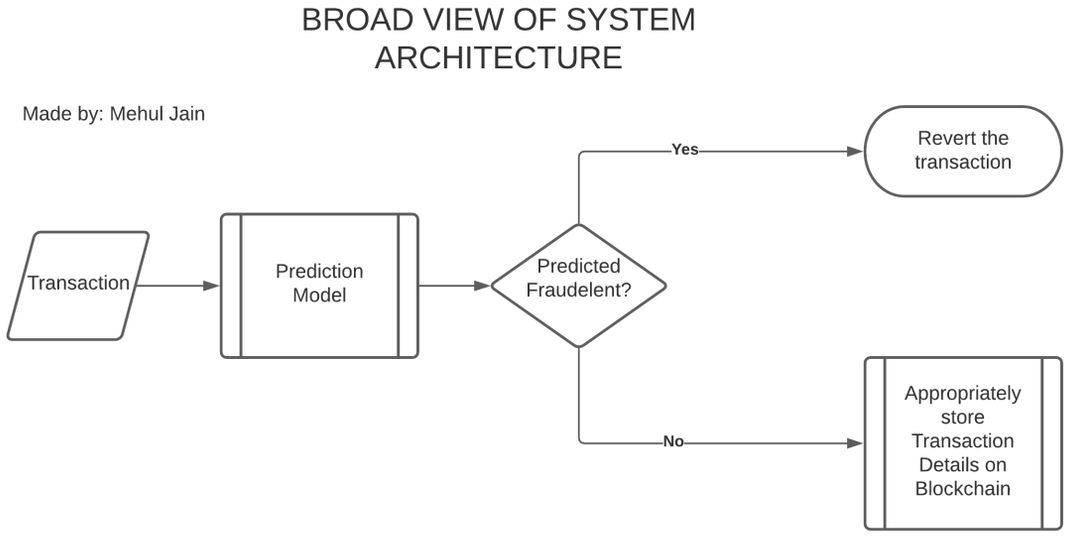

Figure 3.1: System Architecture

In Figure 3.1, the broad view of the overall workflow of the project is depicted. As soon as a transaction takes place, the metadata produced during the transaction would be fed to the proposed ML model which is built on LightGBM based approach. The ML

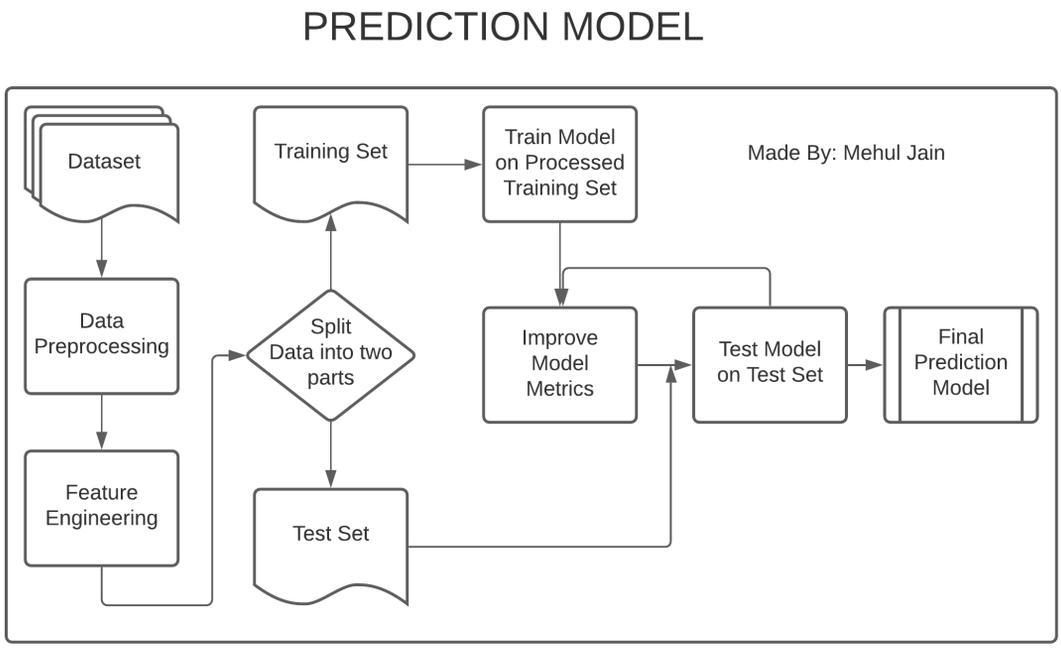

Figure 3.2: Prediction Model

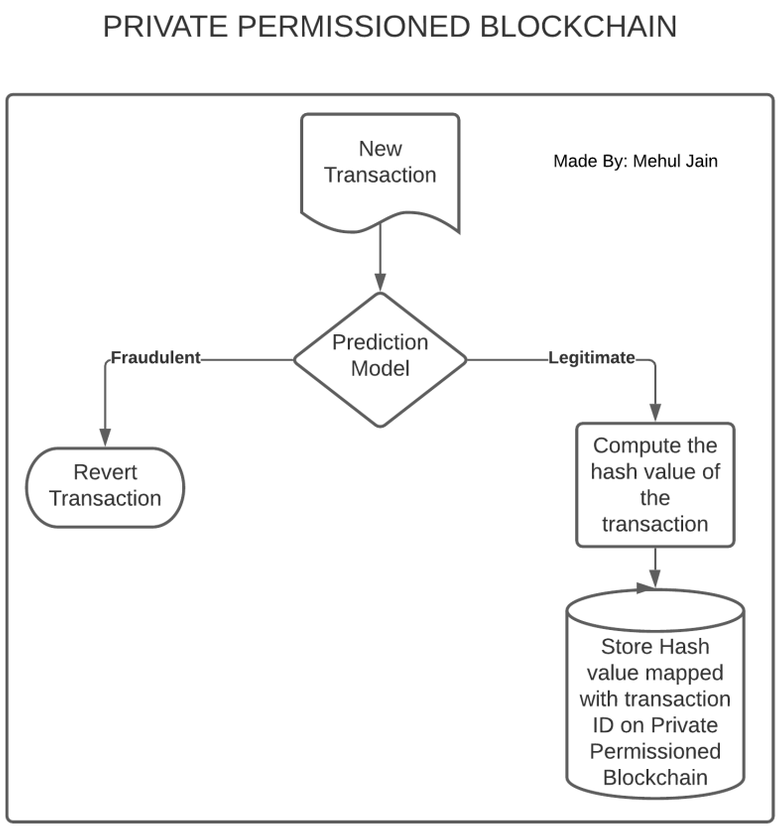

Figure 3.3: Private Permissioned Blockchain

model would then make its prediction on this transaction according to the pre-trained algorithm as one of the following, fraudulent or genuine transaction. If the current transaction is being labelled as fraudulent, the transaction would not be completed and its hash value would not be stored on the blockchain. But, if the transaction is labelled as a legitimate transaction, then its hash value of the transactional information would be computed and stored on the distributed ledger, which is the blockchain created, to check for the integrity of transactional informartion in future.

Figure 3.2, represents the workflow in designing and training our LightGBM based Ml Model and in Figure 3.3, the workflow of our tasks including Private Permissioned Blockchain are represented.

3.1.1 LightGBM based ML model

LightGBM is a gradient boosting framework that uses tree based learning algorithms. It uses two novel techniques :

- **GOSS Technique for LightGBM:** Different data instances have varied roles in the computation of information gain. The instances with larger gradients(i.e., under-trained instances) will contribute more to the information gain. GOSS keeps those instances with large gradients (e.g., larger than a predefined threshold, or among the top percentiles), and only randomly drop those instances with small gradients to retain the accuracy of information gain estimation. This treatment can lead to a more accurate gain estimation than uniformly random sampling, with the same target sampling rate, especially when the value of information gain has a large range.

- **EFB Technique for LightGBM:** High-dimensional data are usually very sparse which provides us a possibility of designing a nearly lossless approach to reduce the number of features. Specifically, in a sparse feature space, many features are mutually exclusive, i.e., they never take nonzero values simultaneously. The exclusive features can be safely bundled into a single feature (called an Exclusive Feature Bundle). Hence, the complexity of histogram building changes from O(#data $\times$ #feature) to O(#data $\times$ #bundle), while #bundle « #feature . Hence, the speed for training framework is improved without hurting accuracy.

The two techniques of GOSS and EFB together makes the model work efficiently.

Its Architecture is such that it splits the tree leaf-wise as opposed to other boosting algorithms that grow tree level-wise. It chooses the leaf with maximum delta loss to grow. Since the leaf is fixed, the leaf-wise algorithm has lower loss compared to the level-wise algorithm. Leaf-wise tree growth might increase the complexity of the model and may lead to over-fitting in small datasets[cite here], but as our dataset is extensive,

hence we don't need to worry about that. Light GBM based model have Faster training speed and higher efficiency, Lower memory usage, Better accuracy and the capabilility of handling large-scale data.

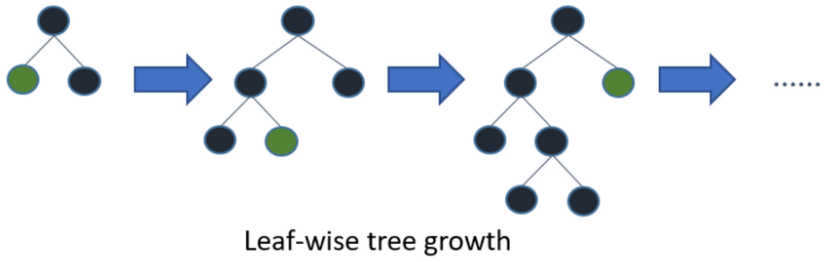

Figure 3.4: Leaf-wise tree growth [9]

Figure 3.5: Level-wise tree growth [9]

3.1.2 Private Permissioned Blockchain

A Private Blockchain is a type of Blockchain where the network authority is controlled by a single organization. To be a part of this type of blockchain, you must be validated by the central organization or the rules setup by them. The organizations that has created this network would generally only allow the permissioned members/nodes to participate in the network.

Participants need to obtain an invitation or permission to join. The way in which access is being provided for new members can vary: existing participants could decide the new members; a regulatory intermediary can decide it; or a consortium could make the decisions instead. Once a member is approves and joins the network, it would be responsible for maintenance of the chain as in a public blockchain.

Some basic features of the Private Permissioned Blockchain that we are going to implement:

- SHA-256 Encryption is done to secure the newly created block.

- All members are added to the network by invitation only.
- Every member that is given permission to be a part of the network is visible to other members. This way any unauthorised members of the network can be detected and reported.
- The verification is done by a a group rather than a single individual, thus ensuring decentralization of point of authority.

3.2 METHODOLOGY

3.2.1 Data Preprocessing & Feature engineering

The Dataset chosen is IEEE-CIS Dataset who have taken the data from Vesta Corporation's real-world e-commerce transactions and contains a wide range of features from device type to product features. The dataset consists of four tables, the training and test tables for Transactions and Identity. The dataset consists of over 400 features and over a million transactions.

The description of tables in the dataset is as followed:

- **Transaction Table:**
  Transaction tables have 394 features including 22 categorical features and 372 numeric features.Features in this table include the transaction related details like the transaction amount, payment card information, such as card type, card category, issue bank, country, etc., purchaser and recipient email domain, Vesta engineered rich features, including ranking, counting, and other entity relations etc. To know some insights about the features, refer Table 3.1.
- **Identity Table**:
  It consists of 41 features including ID, numerical and categorical features. Features in this table are identity information – network connection information (IP, ISP, Proxy, etc) and digital signature (UA/browser/os/version, etc) associated with transactions. Refer table 3.2 to get some insights about the features.

The two tables in our dataset, *Transaction Table* (Table 3.1) and *Identity Table* (Table 3.2) will be merged using the common feature present in both the tables, which is the *TransactionID*. All the rows in the *Transaction Table* will be mapped to their corresponding row in the *Identity Table* using the *Transaction ID*.

Table 3.1: TRANSACTION TABLE

Features	Features Description	Type
Transaction ID	ID of transaction	ID
isFraud	Binary classifier	categorical
Transaction DT	Transaction Date	time
ProductCD	Product Code	categorical
TransactionAmt	Transaction Amount	numerical
card1-card6	card	categorical
addr1-addr2	address	categorical
M1-M9	anonymous feature	categorical
P_ email domain	Purchaser Email Domain	categorical
R_ email domain	Receiver Email Domain	categorical
dist1-dist2	distance	numerical
C1-C14	counting	numerical
D1-D15	timedelta	numerical
V1-V339	Vesta Engineered Features	numerical

Table 3.2: IDENTITY TABLE

Features	Features Description	Type
Transaction ID	ID of Transaction	ID
DeviceType	Device Type	categorical
DeviceInfo	Device Information	categorical
id01-id38	Masked data	numerical

But before merging the tables, ***Data Preprocessing*** should be performed on the data to filter and clean data for further processing.

First of all we remove all the features having *percentage of null values greater than 90 percent* and having *less than 2 unique values*. We did so because due to lack of information in features that fall into this category, they aren't going to contribute in our prediction model.

For further preprocessing, we would be handling the task differently based on the type of data present in the column(Numerical or Categorical).

- **Numerical Features:**
  We can reduce the overall size of our dataset by applying *Memory Compression Technique* on our numerical data. The basic idea of the technique to change the data type of a feature based on the maximum and minimum value present for that feature. Based on this, we attempt to get max and min value of a column, which decides the most suited data type for the whole column, and substitutes the default data type with that. For example, if the maximum and minimum value in a feature(originally with data type *int64*) are 102 and -40 respectively, we

can reduce memory consumption by changing the data type of feature to *int8*(as data range for it is -128 to 127, which can easily incorporate the maximum and minimum value in the column).

- **Categorical Data:**
  Meanwhile, for categorical columns we convert them from string to int using *Ordinal Encoding*. For example, convert ['A', 'B', 'C'] to [1,2,3]

After that, we perform the ***feature selection technique***, specifically the Recursive Feature Elimination Technique that is implemented using the *"RFECV(Recursive Feature Elimination with Cross Validation)"* class of the *"feature_selection"* module of *"sklearn"* library. After running it, *159 features* were selected out of total 434 features from merged dataset. Some of those features are given in Figure 3.6.

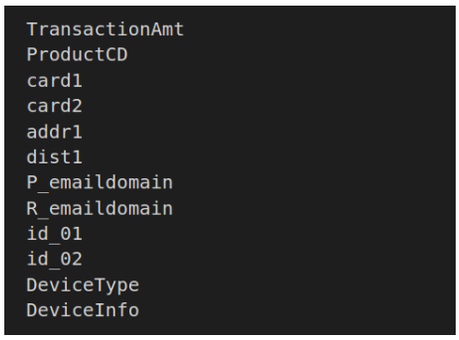

Figure 3.6: Some of the selected features after RFECV

This technique is basically a backward selection of the predictors. In the start, we choose the entire set of predictors to build out model and calculate the importance score of each of the predictors. According to those values, the feature(s) with least importance are removed and the scores are calculated again for further computations. The process is repeated till we reach an optimum set of predictors/features based on our *stopping condition*. The optimal subset is then used to train the final model. In RFECV, CV is for cross validation. Cross-Validation is a technique for evaluating ML models by training several ML models on subsets of the available input data and evaluating them on the complementary subset of the data. Use cross-validation to detect overfitting, ie, failing to generalize a pattern. RFECV uses feature ranking with recursive feature elimination and cross-validated selection of the best number of features.

After preprocessing, ***Feature Engineering*** is performed to extract some more information regarding our dataset. New features like :

- **Frequency counts :** Count the frequency of important categorical variables related to card, address, emaildomain and product code (demonstarted only three of them). Credit card which are used frequently have lesser chance of fraud (Ref

Figure 3.7). Information on transactions from a specific address (Ref Figure 3.8) or transactions taking place on a specific email domain (Refer Figure 3.9) can also be used to classify our transactions.

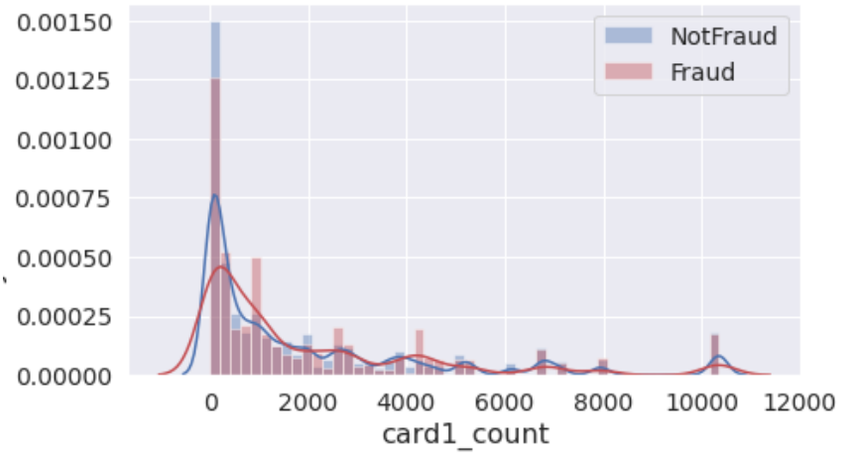

Figure 3.7: Distribution for card1_count

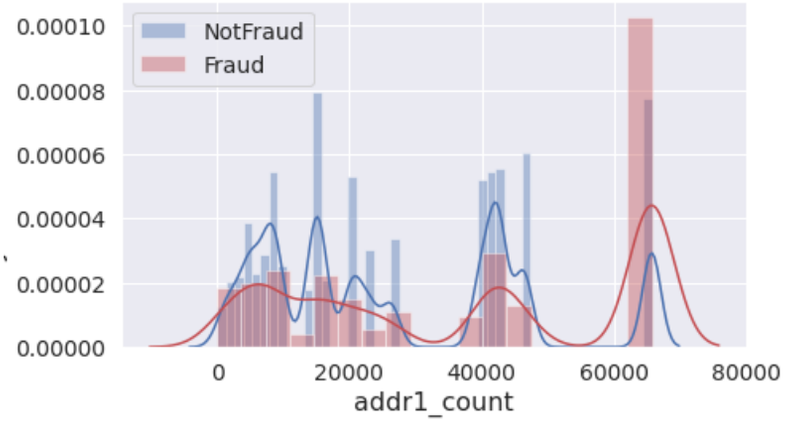

Figure 3.8: Distribution for addr1_count

- **Hour of day :** From TransactionDT extract the hour of day of the transaction time, encoded as 0-23 in the column. TransactionDT field indicates the timestamp of a transaction and we can easily obtain data related to time from it. Refer Figure 3.10 and note that more frauds are committed between 11PM and 1 AM. This is probably because fraud originated in different time zone.

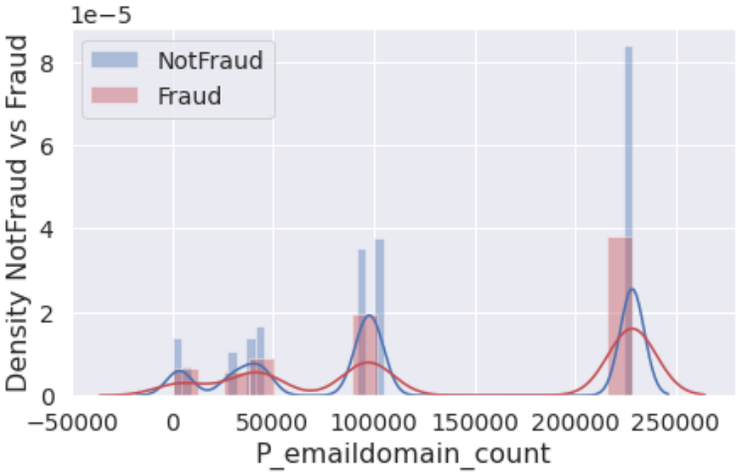

Figure 3.9: Distribution for p_email_domain_count

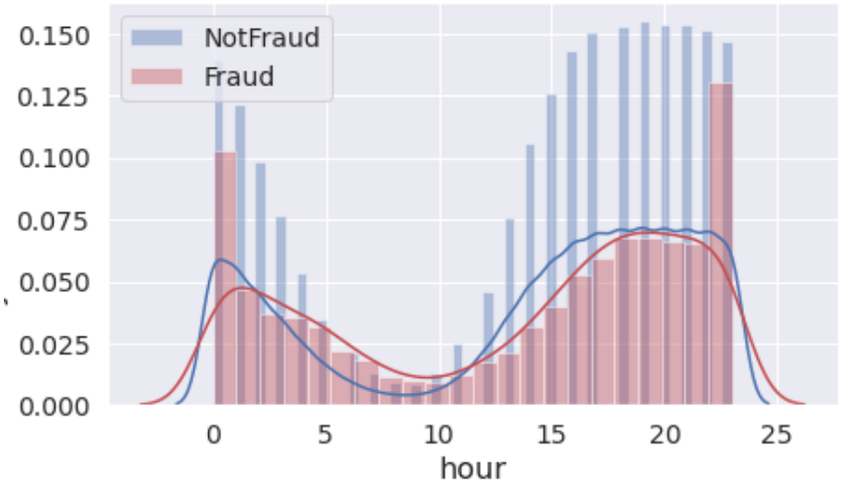

Figure 3.10: Distribution for Time(in Hr)

- **Transaction Amount decimal part :** It is evident from Figure 3.11 that the transactions having decimal part in the amount for transactions can also be used to determine the nature of the transactions as the amount here is in dollars and unusual decimals in the amount may account for the currency change factor (as fraudulent transactions may be initiated from outside the country).

- **Number of null :** The number of null values for a transaction can also prove to be an important identifier in classifying the transactions. (Refer Figure 3.12 for its distribution).

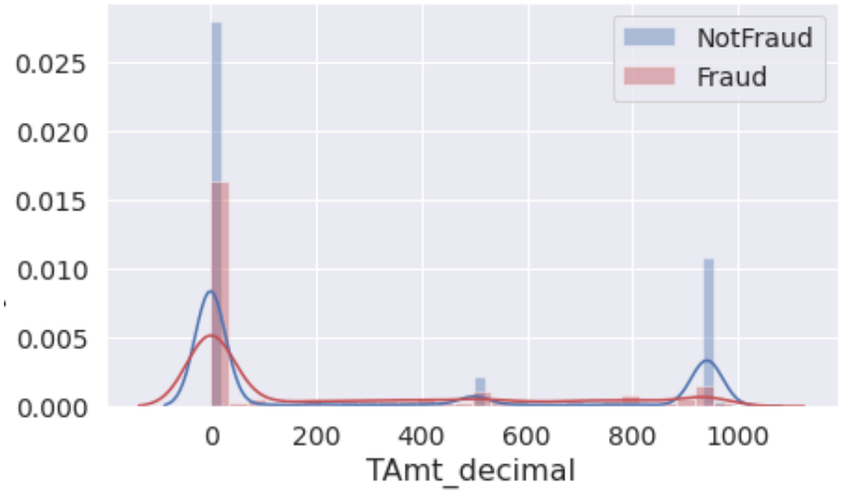

Figure 3.11: Distribution for decimal in Transaction Amount

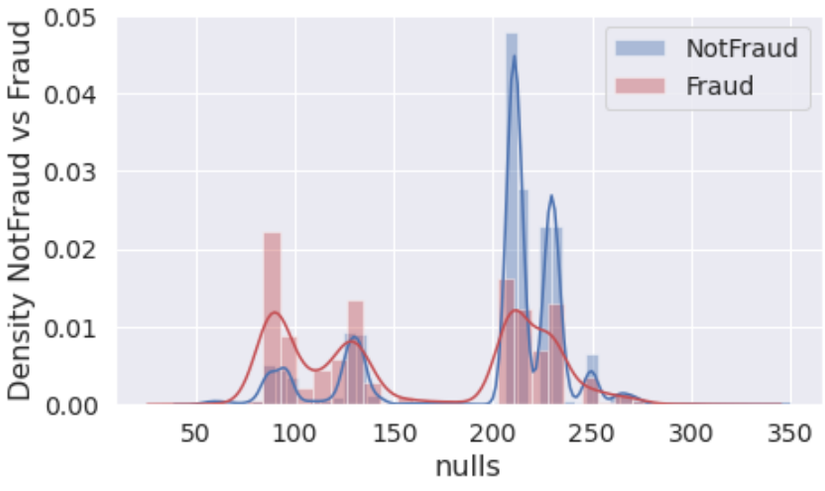

Figure 3.12: Distribution for Number of Null Values in Transaction

3.2.2 Training LightGBM Model

We would be using the *lightgbm* library for implementation of our LightGBM based Model. The environment for the experiment is Linux Ubuntu 20.04 and it is carried out on an Intel's Hexa-Core $I7^{th}$ Generation Processor.

Compare with Xgboost and other classical models, LightGBM adopts many optimizations like Gradient-based One-Side Sampling(GOSS) and Exclusive Feature Bundling-(EFB). By using GOSS, the model can exclude a significant proportion of data instances with small gradients, and only use the rest to estimate the information gain. With EFB, the model exclusive features to reduce the number of features and thus prediction speed is improved.

Table 3.3: PARAMETERS OF LightGBM MODEL

Parameter	Parameter Description	Value
n_estimators	number of estimators	10000
learning_rate	learning rate	0.0068832
bagging_fraction	sample rate of rows	0.41811
max_depth	max depth of each tree	-1
feature_fraction	sample rate of columns	0.37974
boosting_type	type of boosting	gbdt
num_leaves	number of leaves	49
bagging_fraction	bagging fraction	0.41811
bagging_seed	bagging seed	11

We trained our model with the processed dataset and choose the hyperparameters as given in Table 3.3 which have been calculated using the *Baysian Optimization* technique. In this technique, we build a probability model of the objective function and use it to select the most promising hyperparameters to evaluate in the true objective function.

3.2.3 Creating the Blockchain Network

The blockchain network is used to store the banking activity of a user in a distributed ledger. But, as a single transaction have many metadata associated with it, storing all of them on the blockchain network would become quite costly. So here we use the *Hashing* to solve our situation.

Hashing is the process of transforming any given string value of any length to another value of fixed length which is usually shorter then the original string. The key or the hash value generated makes it easier to find or employ the original string. For same input the hash output should be same and for different inputs the hash value should be different (reduce the number of collisions)

So instead of storing the whole transaction related information on the blockchain itself, the data is first hashed using the hash function and only the computed hash value is stored on the blockchain. In comparison to the original banking activity data, the computed hash value is very small, so the cost of a transaction is relatively low. The raw data can be stored in a distributed database or just a conventional file system. But we need to make sure, that we assign the key (computed hash value) stored on blockchain with its corresponding original data in database. In the distributed database an additional column is created to store the key mapped to the original data, hence allowing us to access the information by just using the computed key (hash value).

This helps us to look out for any sort of tampering with our data by just using the computed hash value. We just need to hash the original data stored in the database and compare the value with the hash value that is stored on blockchain corresponding to this transaction. If the value was being tampered with, we would know it as in that case the stored hash value would not match with the newly computed hash value.

CHAPTER 4

RESULTS AND DISCUSSION

In this chapter, we are going to discuss about the tasks completed by us and results obtained while implementing it. This chapter also concludes the paper by giving some future works that can be done to extend the project.

4.1 RESULTS

We deployed our ML model on a large publicly available IEEE-CIS fraud dataset detailed above. We apply AUC-ROC score, known as "Area under the ROC curve", and accuracy to evaluate the performance of our model. ROC curve plots TPR versus FPR.

$$TPR = \frac{TP}{TP + FN}, \tag{1}$$

$$FPR = \frac{FP}{FP + TN}, \tag{2}$$

Intuitively, AUC-ROC score of 0.0 means TPR of model is very low and FPR becomes much large, that's to say the model predicts extremely terrible. On the contrary, AUC-ROC of 1.0 indicates a model predicts 100% correct as FPR tends to 0.

Another metric for classification is accuracy, which indicates the proportion of correct predictions (both true positives and true negatives) among the total number of cases examined. So, we can state that higher the accuracy, better the performance of out proposed Model.

$$\text{accuracy(ACC)} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

In addition to AUC-ROC score, we also provided the Accuracy score of different models. In Table 4.1, we have compared the score of our model with two other models.

Table 4.1: PERFORMANCE METRICS

Model	**AUC-ROC Score**	**Accuracy**
NN(ensemble)	0.910	0.958
Random Forest	0.927	0.974
LightGBM	0.973	0.987

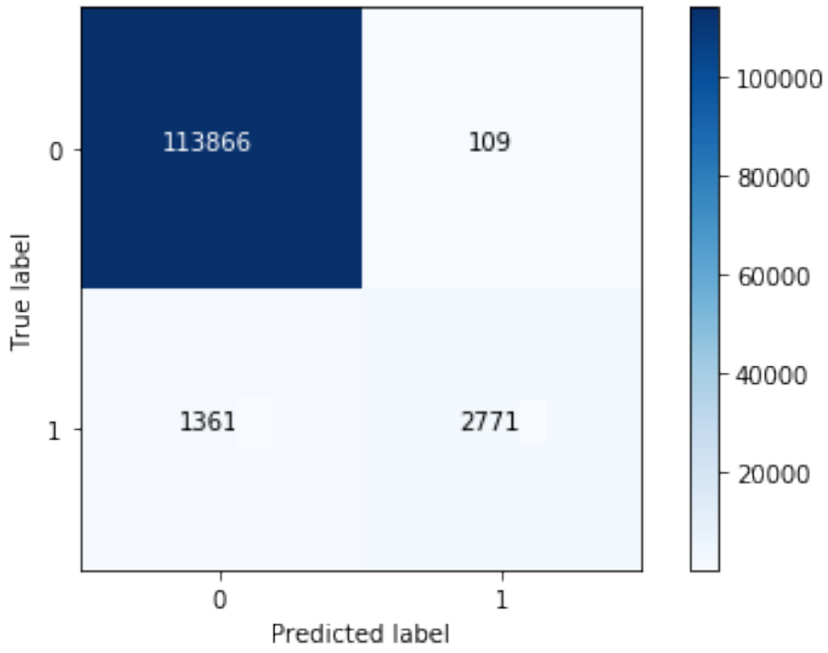

Figure 4.1: Confusion Matrix

From Table 4.1, it is easy to find out that LightGBM based model outperforms the Deep Learning based NN(emsemble) model which was proposed in [8] on both AUC Score and Accuracy. The score of our model as compared to that proposed in [12] is also far much better.

When using tree based algorithms like LightGBM or XGBoost, it is quite easy to show the feature importance of every feature. In practice, it is common to make a trade-off between accuracy performance and computation speed. We can choose important features based on the output of out feature importance graph as it would help us by providing a guidline so as to which feature contributes more towards prediction and which feature less.

Also we have managed to create a simple blockchain network that stores the hash value of the users banking activity in it and works as expected.

4.2 CONCLUSION

This paper have proposed to integrate Machine learning with Blockchain Technology so as to design a system that can efficiently segregate suspicious transactions and stop them from completing. The paper proposes a Light GBM based Machine learning model as it can tackle the skewed nature of the real world transaction dataset efficiently. The accuracy achieved by the model proposed is quite good as compared to models proposed in other paper related to this.
A Blockchain network is also created to store the hashed value of metadata of a transaction in it through which one could check for integrity of the data by comparing the hash value stored with newly computed value.

4.3 FUTURE WORK

The following work can be integrated to improve and extend the functionalities of the project:

- More extensive work on feature engineering part can be done on the dataset which may improve the accuracy and prediction time of the model.
- Other approaches like CatBoost ML algorithm can also be experimented with.
- More banking features can be implemented on blockchain to increase automation, authenticity and integrity of the work performed.
- Scalability of the proposed solution also needs some more research.

REFERENCES

[1] Maha A. Alanezi, Mawra T. Homeed, Zahra S. Mohamed, and Ahmed M. Zeki. Comparing naïve bayes, decision tree and logistic regression methods in fraudulent credit card transactions. In *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*, pages 1–5, 2020.

[2] Yeming Chen and Xinyuan Han. Catboost for fraud detection in financial transactions. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 176–179, 2021.

[3] Wenkai Deng, Ziming Huang, Jiachen Zhang, and Junyan Xu. A data mining based system for transaction fraud detection. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 542–545, 2021.

[4] Dingling Ge, Jianyang Gu, Shunyu Chang, and JingHui Cai. Credit card fraud detection using lightgbm model. In *2020 International Conference on E-Commerce and Internet Technology (ECIT)*, pages 232–236, 2020.

[5] Shayan Ghosh. Research by transunion finds digital fraud attempts increasing from india.

[6] Nana Kwame Gyamfi and Jamal-Deen Abdulai. Bank fraud detection using support vector machine. In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pages 37–41, 2018.

[7] Xinyi Hu, Haiwen Chen, and Ranxin Zhang. Short paper: Credit card fraud detection using lightgbm with asymmetric error control. In *2019 Second International Conference on Artificial Intelligence for Industries (AI4I)*, pages 91–94, 2019.

[8] Xiong Kewei, Binhui Peng, Yang Jiang, and Tiying Lu. A hybrid deep learning model for online fraud detection. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 431–434, 2021.

[9] Pushkar Mandot. What is lightgbm, how to implement it? how to fine tune the parameters?

[10] E. A. Minastireanu and G. Mesnita. J. inform. assur. cybersecur. In *Light gbm machine learning algorithm to online click fraud detection*, 2019.

[11] J T S Quah and M. Sriganesh. Expert systems with applications. In *Real-time credit card fraud detection using computational intelligence[J]*, volume 35, pages 1721–1732, 2008.

[12] Du Shaohui, GuanWen Qiu, Huafeng Mai, and Hongjun Yu. Customer transaction fraud detection using random forest. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 144–147, 2021.

[13] Zijian Song. A data mining based fraud detection hybrid algorithm in e-bank. In *2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pages 44–47, 2020.