A Research Paper on Credit Card Fraud Detection

BORA MEHAR SRI SATYA TEJA[1], BOOMIREDDY MUNENDRA[2], Mr. S. GOKULKRISHNAN[3]

[1]*Final Year Student, Department of Computer Science and Engineering, SCSVMV, Kanchipuram*
[2]*Final Year Student, Department of Computer Science and Engineering, SCSVMV, Kanchipuram*
[3]*Assistant Professor, Department of Computer Science and Engineering, SCSVMV, Kanchipuram*

Abstract – *Credit card frauds are one of the most common frauds happening now. Many companies have been increasing their payment modes to online, rising the threat for online frauds. Many fraudsters started using different methods to steal the money used to made the online transactions. So, our aim is to use different machine learning algorithms to check whether the transactions made are fraud or genuine. So, we will be categorising the transactions into different groups so that we can apply different machine learning algorithms on them. Then different classifiers will be trained over the groups independently. Then the best classifier with a good accuracy score will be used to predict the fraud transactions. In this paper we will be using a dataset containing. The dataset is a collection of online transactions made by some anonymous people using their credit cards. This dataset is very unstable i.e., it has a large portion of genuine transactions and a very small number of fraud transactions.*

Key Words: Accuracy, Error-rate, Sensitivity, Specificity

1. INTRODUCTION

Credit card fraud means unauthorized operation of an account that is used to make transactions without the actual owner of the account or the bank authority's knowledge. We need to take necessary precautions while doing these transactions to avoid these frauds. Also, the bank authorities need to use the latest technologies to predict these frauds so that they can alert their customers beforehand.

Fraud detection means (for our dataset) is to predict the transactions that are made by the account holders which are actually done by other people who has access to the account. This is a very complex problem that needs the attention of the account holder as well as the bank authorities so that their other customers need not suffer from the same problem. But this problem has a problem of class imbalance. The number of genuine transactions done by a customer will be far higher than the fraud transactions happened or even be zero. Also, the customer can do a transaction that deviates from his previous transactions that can be misinterpreted as a fraud transaction.

Also, the payment requests sent are checked by automatic tools that confirms which request need to be confirmed. These algorithms check these requests and report suspicious requests to professionals who operate behind and they in turn investigate them by contacting the owners of the accounts whether the transactions are genuine or not

1.1 LITERATURE SURVEY

There were different techniques that were used to predict the fraud transactions like Outlier detection, unsupervised outlier detection, Peer group analysis and breakpoint analysis.

Outlier detection detects the abnormal transactions made by the user which are different in scale, rage and type of transaction compared to the previous transactions, but these types of transactions can be actually done by the customer and so the prediction can be wrong.

Unsupervised outlier detection on the other hand does predict the required data, It just simply understands the behaviour of the customer transactions. Peer group analysis is another method that has been used which involves the comparison of entities that share similar characteristics.

A breakpoint is a structural change in data like an anomaly. Breakpoint analysis simply is analysing these breakpoints to better understand their existence and occurrence.

Although supervised learning methods are used in fraud detection there is a possibility that they fail at some cases.

2. PROBLEM STATEMENT

Credit cards are an essential financial tool that enables its holders to make purchases and the luxury of paying back the amount later. Credit card holders have an advantage of paying the amount back later after a certain time. This makes the credit cards an easy target for the fraudsters. Without the owner's knowledge a good amount of money can be withdrawn by these fraudsters and they make it look like the actual owners of these cards made the withdrawal. The fraudsters make does this very carefully and anonymously that makes it difficult to stop and even catch them. In 2017, there were data breaches and approximately 179 million records among which Credit card frauds were the most common form. With many frauds happening all over the world with credit card

frauds on the top, this makes this a serious issue to look after. Credit card dataset is largely imbalanced because there will be more valid data compared with a fraudulent bone. Banks are now moving to EMV cards, which store their data on integrated circuits making some card payments safer, but still leaving non-card payment frauds on advanced rates. According to 2017, the US Payments Forum report, felons have loosened their focus on conditioning related to CNP deals as the security of chip cards were increased.

2.1 PROPOSED SYSTEM

Card payments are always different when compared to former payments made by the client. This creates a problem called conception drift. Concept drift can be said as a variable which changes over time and in unlooked-for ways. These variables create a high imbalance in data. The main agenda of our exploration is to overcome the problem of Concept drift to apply on real- world script. In our proposed system we will be using different machine learning algorithms like Decision trees, Random Forest and other algorithms and calculate their accuracy scores and then choose the best algorithm with the best accuracy score. We will also calculate the confusion matrix for each of the algorithm and take that into consideration along with the accuracy score to choose the best algorithm. Also, we need to consider the fact that our data set that we are about to look at is very much imbalanced.

2.2 DATASET

The dataset comprises transactions made by European credit cards Holders in September 2013. This dataset presents deals that passed in two days, where we've 492 frauds out of deals. The dataset is largely unstable, the positive class (frauds) account for 0.172 of all deals. Features v1,v2,v3…v28 are the key features achieved with PCA, the only features which haven't been converted with PCA are' Time 'and' Quantum'. Point' Time 'contains the seconds ceased between each sale and the first sale in the dataset. The point' Quantum's the sale Quantum, this point can be used as the amount. Point Class is the response variable and it takes values 1and 0 for fraud and genuine respectively.

3. STEPS AND IMPLEMENTATION

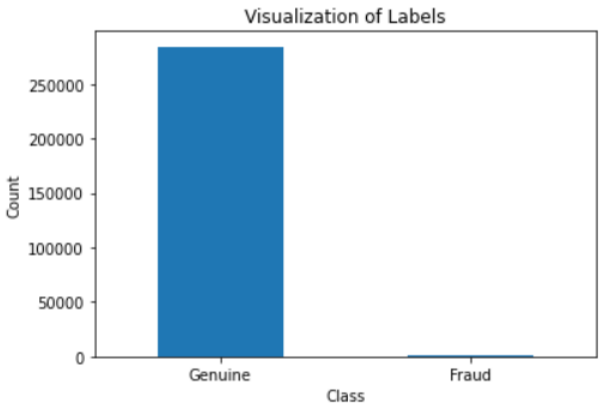
Steps to develop the Classifier in Machine Learning

- Complete the Exploratory Data Analysis on the dataset
- Apply different ML algorithms on our dataset
- Train and evaluate the models to pick the best one

Step 1. Complete the Exploratory Data Analysis on the dataset

First, we will import the required modules, load the dataset, and perform EDA on it. Then we will make sure there are no null values in our dataset. The feature that we will be focusing is "Amount".

Now, if we traverse the existence of each class tag and plot the data using matplotlib the plot will be as follows



We can observe from the above bar graph that the genuine transactions are over 99%. So, to avoid this problem we can apply the scaling techniques on the "Amount" feature to transform them to the range of values. We will remove the "Amount" column and add a new column with the scaled values in its place. We will also remove the "Time" column as it is not required.

Step 2: Use ML Algorithms to the Dataset

Let's use the Random Forest and Decision Tree Classifiers which are present in the sklearn package as RandomForestClassifier() and DecisionTreeClassifier() respectively.

```
# Decision Tree Classifier
from sklearn.tree import DecisionTreeClassifier
decision_tree = DecisionTreeClassifier()
decision_tree.fit(train_X, train_Y)

predictions_dt = decision_tree.predict(test_X)
decision_tree_score = decision_tree.score(test_X, test_Y) * 100
```

```
# Random Forest
from sklearn.ensemble import RandomForestClassifier
random_forest = RandomForestClassifier(n_estimators= 100)
random_forest.fit(train_X, train_Y)

predictions_rf = random_forest.predict(test_X)
random_forest_score = random_forest.score(test_X, test_Y) * 100
```

Step 3: Train and Evaluate the Models

Now, Let's train and evaluate the recently created models and pick the best one. Train the decision tree and random forest models using the fit() function. Note down the

predictions made by the models using the predict () function and evaluate.

Let's visualize the scores of each of our classifiers

```
# Print scores of our classifiers

print("Random Forest Score: ", random_forest_score)
print("Decision Tree Score: ", decision_tree_score)
```

```
Random Forest Score:  99.96254813150287
Decision Tree Score:  99.9204147794436
```

The Random Forest classifier has somewhat an advantage over the Decision Tree classifier.

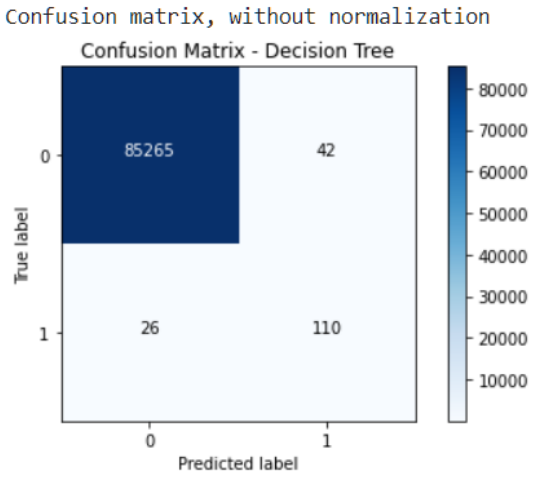
Now we will calculate the accuracy, precision, recall, and f1-score for both of the classifiers by creating a function commonly used to calculate these values

```
# The below function prints the following necesary metrics

def metrics(actuals, predictions):
    print("Accuracy: {:.5f}".format(accuracy_score(actuals, predictions)))
    print("Precision: {:.5f}".format(precision_score(actuals, predictions)))
    print("Recall: {:.5f}".format(recall_score(actuals, predictions)))
    print("F1-score: {:.5f}".format(f1_score(actuals, predictions)))
```

The above function is used commonly to calculate the evaluation metrics for both Random Forest and Decision Tree models.

Now If we visualize the confusion matrix of the ***Decision Tree model.***



The evaluation metrics of the ***Decision Tree model*** will be as follows

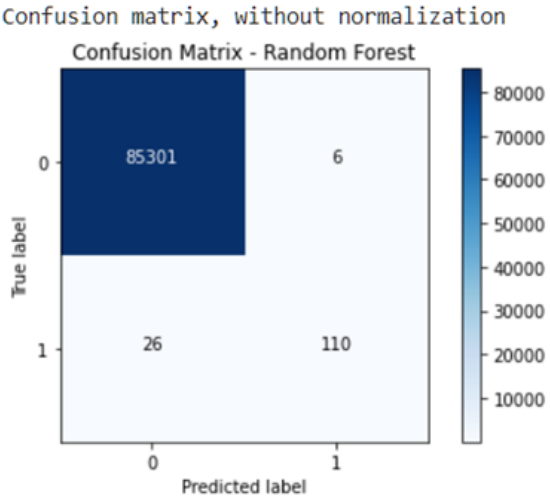
Accuracy: 0.99920

Precision: 0.72368

Recall: 0.80882

F1-score: 0.76389

Now if we visualize the confusion matrix of the ***Random Forest model***



The evaluation metrics of the ***Random Forest model*** will be as follows

Accuracy: 0.99963

Precision: 0.94828

Recall: 0.80882

F1-score: 0.87302

Address the Class-Imbalance issue

The Random Forest does better than the Decision Trees. But our dataset has a serious problem of class imbalance. The genuine transactions are more than 99% and the fraud transactions instituting 0.17%.

With such a diffusion, if we train our model without taking care of the imbalance issues, it predicts the data as genuine transactions as there is more data about them and hence gets more accuracy even though there are some fraud transactions and these are ignored as there is less data about them. The class imbalance problem can be resolved by many methods. Oversampling is one of them.

Oversample the minority class is one of the methods to address the imbalanced datasets. The best solution involves doubling examples in the minority class, even though these instances does not contribute any new data to the model.

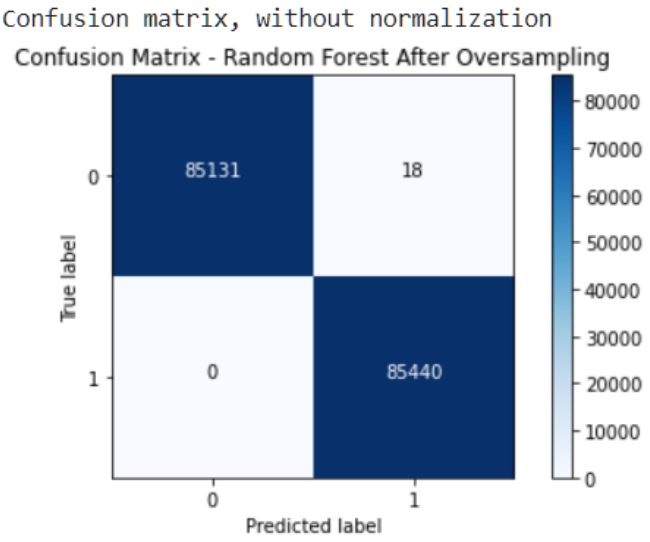
As a substitute, new instances may be produced by duplicating existing ones. The Synthetic Minority Oversampling Technique, or SMOTE for brief, may be a method of knowledge augmentation for the minority

class. The above SMOTE is existing in the imblearn package. Let's import that and resample our data.

So, we resampled our data and we split it using train_test_split () with a split of 70-30. As we can see from previous results that the Random Forest algorithm performed better than the Decision Tree algorithm, we will apply the it to our resampled data (after oversampling).

4. RESULTS AND DISCUSSION

Easily, Random Forest model works better than Decision Trees. But if we observe our dataset suffers a serious problem of class imbalance. The genuine (not fraud) deals are further than 99 with the fraud deals constituting of 0.17. With similar kind of distribution, if we train our model without taking care of the imbalance issues, it predicts the label with more significance given to genuine deals (as there are more data about them) and hence obtains further fragility. The class imbalance problem can be resolved by reasonable number of ways. Over slice is one of them. Finally, after oversampling the confusion matrix and the accuracy scores are calculated.



The evaluation metrics for the ***Random Forest model (after oversampling)*** are as follows:

Accuracy: 0.99989

Precision: 0.99979

Recall: 1.00000

F1-score: 0.99989

As we can see the accuracy scores of the Random Forest model after the oversampling which is done to avoid the class imbalance issue, is quite good and better than the different algorithm approaches. So we can say that the Random Forest algorithm does a good job of predicting the anomalies in a huge imbalanced dataset.

5. CONCLUSION

Credit card fraud is the biggest frauds that are being happened right now around the whole ground. This paper has explained how credit card frauds have been happening and we studied these frauds using a dataset that consists of transactions made in the real world. We saw how different machine learning algorithms are used to predict the fraud transactions on our dataset and we also addressed the class imbalance issue of our dataset and used oversampling to finally use Random Forest classifier that got a good accuracy score.

6. REFERENCES

[1] Credit Card Fraud Detection Based on Transaction Behavior -by John Richard D. Kho, Larry A. Vea" published by Proc. of the 2017 IEEE Region 10 Conference (TENCON), Malaysia, November 5-8, 2017

[2] CLIFTON PHUA1, VINCENT LEE1, KATE SMITH1 & ROSS GAYLER2 " A Comprehensive Survey of Data Mining-based Fraud Detection Research" published by School of Business Systems, Faculty of Information Technology, Monash University, Wellington Road, Clayton, Victoria 3800, Australia

[3] "Survey Paper on Credit Card Fraud Detection by Suman" Research Scholar, GJUS&T Hisar HCE, Sonepat published by International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 3, March 2014

[4] "Research on Credit Card Fraud Detection Model Based on Distance Sum – by Wen-Fang YU and Na Wang" published by 2009 International Joint Conference on Artificial Intelligence

[6] **[5]** "Credit Card Fraud Detection: A Realistic Modelling and a Novel Learning Strategy" published by IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, VOL. 29, NO. 8, AUGUST 2018