iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: VI Month of publication: June 2023

DOI: https://doi.org/10.22214/ijraset.2023.54097

www.ijraset.com

Call: ✆08813907089 | E-mail ID: ijraset@gmail.com

Fraud Detection of Credit Card Using Machine Learning

Wazir Ahmed[1], Prity Roy[2], Utsha Talukder[3], Soumyadip Debroy[4], Suman Saha[5], Debasish Bhattacharjee[6], Nirupam Saha[7], Moloy Dhar[8]
[1, 2, 3, 4, 5,6,7,8] Department of Computer Science & Engineering, Guru Nanak Institute of Technology, Kolkata, India
wazirahmed78692@gmail.com, prityroy12300@gmail.com, talukderutsha09@gmail.com, sdebroy10@gmail.com, suman.saha5858@gmail.com, debasishbhattacharjee3102000@gmail.com, nirupam.saha@gnit.ac.in, moloy.dhar@gnit.ac.in

Abstract: This paper is focused on credit card fraud detection in real world scenarios. Nowadays credit card frauds are increasing in number as compared to earlier times. Criminals are using fake identity and various technologies to trap the users and get the money out of them.

Therefore, it is very essential to find a solution to these types of frauds. In this proposed paper we designed a model to detect the fraud activity in credit card transactions. This system can provide most of the important features required to detect illegal and illicit transactions. As technology changes constantly, it is becoming difficult to track the behavior and pattern of criminal transactions.

To come up with the solution one can make use of technologies with the increase of machine learning, artificial intelligence and other relevant fields of information technology; it becomes feasible to automate this process and to save some of the intensive amounts of labor that is put into detecting credit card fraud. Initially, we will collect the credit card usage data-set by usersand classify it as trained and testing dataset using a random forest algorithm and decision trees. Using this feasible algorithm, we can analyze the larger data-set and user provided current data-set. Then augment the accuracy of the result data. Proceeded with the application of processing of some of the attributes provided which can find affected frauddetection in viewing the graphical model of data visualization. The performance of the techniques is gauged based on accuracy, sensitivity, and specificity, precision. The results is indicated concerning the best accuracy for Random Forest are unit 98.6% respectively.

Keywords: Machine Learning, Scaling, Random Forest, Artificial Intelligence, credit card

I. INTRODUCTION

Nowadays Credit card usage has been drastically increased across the world, now people believe in going cashless and are completely dependent on online transactions. The creditcard has made the digital transaction easier and more accessible. A huge number of dollars of loss are caused every year by the criminal credit card transactions. Fraud is as old as mankind itself and can take an unlimited variety of different forms. The PwC global economic crime survey of 2017 suggests that approximately 48% of organizations experienced economic crime. Therefore, there's positively a necessity to unravel the matter of credit card fraud detection. Moreover, the growth of new technologies providessupplementary ways in which criminals may commit a scam. The use of credit cards is predominant in modern day society and credit card fraud has been kept on increasing in recent years. Huge Financial losses have been fraudulent effects on not only merchants and banks but also the individual person who are using the credits. Fraud may also affectthe reputation and image of a merchant causing non-financial losses that. For example, ifa cardholder is a victim of fraud with a certain company, he may no longer trust their business and choose a competitor. Fraud Detection is the process of monitoring the transaction behavior of a cardholder to detect whether an incoming transaction is authentic and authorized or not otherwise it will be detected as illicit. In a planned system, we are applying the random forest algorithm for classifying the credit card dataset. Random Forest is an associate in the nursing algorithmic program for classification and regression. Hence, it is a collection of decision tree classifiers. The random forest has an advantage over the decision tree as it corrects the habit of over fitting to their training set. A subset of the training set is sampled randomly so that to train each individual tree and then a decision tree is built, each node then splits on a feature designated from a random subset of the complete feature set. Even for large data sets with many features and data instances, training is extremely fast in the random forestand because each tree is trained independently of the others. The Random Forest algorithm has been found to provide a good estimate of the generalization error and to beresistant to over fitting.

II. LITERATURE SURVEY

With growing advancement in the electronic commerce field, fraud is spreading all over the world, causing major financial losses. In the current scenario, Major

Meta learning strategy, neural network cause of financial losses is credit card fraud; it not only affects tradesperson but also individual clients. Decision tree, Genetic algorithm, HMM are the presented methods used to detect credit card frauds. In contemplating system for fraudulent detection, artificial intelligence concept of Support Vector Machine (SVM) & decision tree is being used to solve the problem. Thus, by the implementation of this hybrid approach, financial losses can be reduced to greater extent.

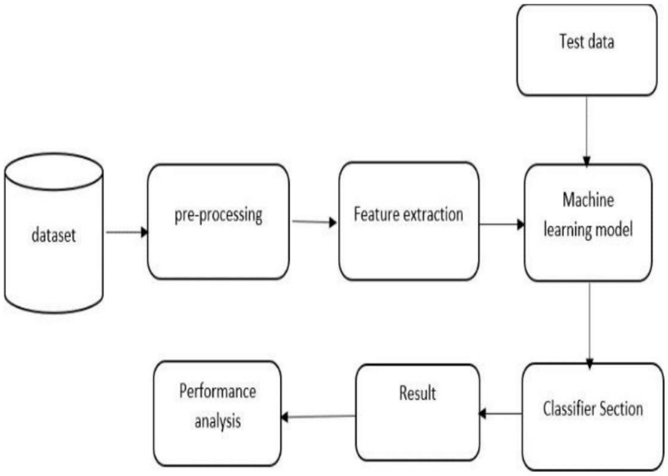
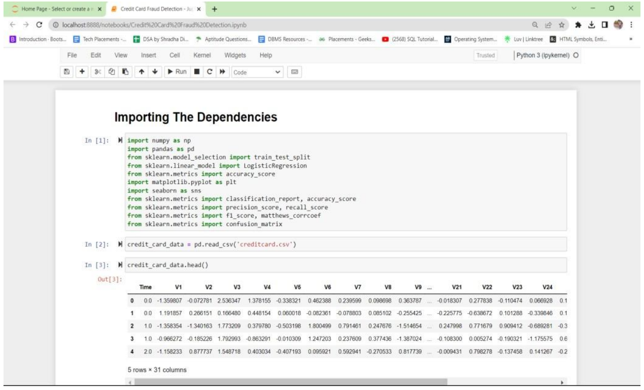


Fig. 1

Mobile payment fraud is the unauthorized use of mobile transaction through identity theft or credit card stealing to fraudulently obtain money. Mobile payment fraud is a fast growing issue through the emergence of smart phone and online transition services. In the real world, a highly accurate process in mobile payment fraud detection is needed since financial fraud causes financial loss. Therefore, our approach proposed the overall process of detecting mobile payment fraud based on machine learning, supervised and unsupervised method to detect fraud and process large amounts of financial data. Moreover, our approach performed sampling process and feature selection process for fast processing with large volumes of transaction data and to achieve high accuracy in mobile payment detection. F-measure and ROC curve are used to validate our proposed model.

We propose a Machine learning model to detect fraudulent credit card activities in online financial transactions. Analyzing fake transactions manually is impracticable due to vast amounts of data and its complexity. However, adequately given informative features, could make it is possible using Machine Learning. This hypothesis will be explored in this paper.



Importing The Dependencies

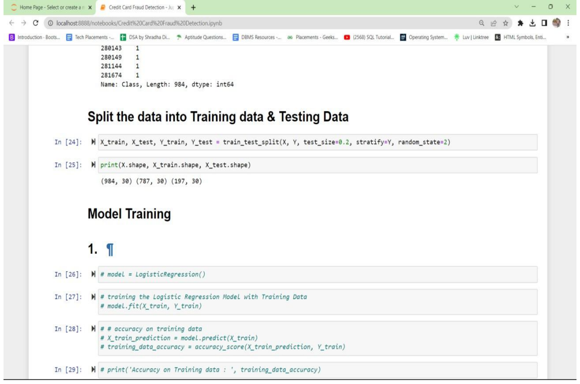
```
In [1]:  import numpy as np
         import pandas as pd
         from sklearn.model_selection import train_test_split
         from sklearn.linear_model import LogisticRegression
         from sklearn.metrics import accuracy_score
         import matplotlib.pyplot as plt
         import seaborn as sns
         from sklearn.metrics import classification_report, accuracy_score
         from sklearn.metrics import precision_score, recall_score
         from sklearn.metrics import f1_score, matthews_corrcoef
         from sklearn.metrics import confusion_matrix

In [2]:  credit_card_data = pd.read_csv('creditcard.csv')

In [3]:  credit_card_data.head()

Out[3]:
   Time        V1        V2        V3        V4        V5        V6        V7        V8        V9  ...       V21       V22       V23       V24
0   0.0 -1.359807 -0.072781  2.536347  1.378155 -0.338321  0.462388  0.239599  0.098698  0.363787  ... -0.018307  0.277838 -0.110474  0.066928  0.1
1   0.0  1.191857  0.266151  0.166480  0.448154  0.060018 -0.082361 -0.078803  0.085102 -0.255425  ... -0.225775 -0.638672  0.101288 -0.339846  0.1
2   1.0 -1.358354 -1.340163  1.773209  0.379780 -0.503198  1.800499  0.791461  0.247676 -1.514654  ...  0.247998  0.771679  0.909412 -0.689281 -0.3
3   1.0 -0.966272 -0.185226  1.792993 -0.863291 -0.010309  1.247203  0.237609  0.377436 -1.387024  ... -0.108300  0.005274 -0.190321 -1.175575  0.6
4   2.0 -1.158233  0.877737  1.548718  0.403034 -0.407193  0.095921  0.592941 -0.270533  0.817739  ... -0.009431  0.798278 -0.137458  0.141267 -0.2

5 rows × 31 columns
```

Fig. 2

To classify fraudulent and legitimate credit card transaction by supervised learning Algorithm such as Random Forest. To help us to get awareness about the fraudulent and without loss of any financially.

III. MODULES

1) Data collection
2) Data pre-processing
3) Feature extraction
4) Evaluation model



```
280143    1
280149    1
281144    1
281674    1
Name: Class, Length: 984, dtype: int64
```

Split the data into Training data & Testing Data

```
In [24]:  X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, stratify=Y, random_state=2)

In [25]:  print(X.shape, X_train.shape, X_test.shape)

          (984, 30) (787, 30) (197, 30)
```

Model Training

1. ¶

```
In [26]:  # model = LogisticRegression()

In [27]:  # training the Logistic Regression Model with Training Data
          # model.fit(X_train, Y_train)

In [28]:  # # accuracy on training data
          # X_train_prediction = model.predict(X_train)
          # training_data_accuracy = accuracy_score(X_train_prediction, Y_train)

In [29]:  # print('Accuracy on Training data : ', training_data_accuracy)
```

Fig. 3

A. *Data Collection*

Data used in this paper is a set of product reviews collected from credit card transactions records. This step is concerned with selecting the subset of all available data that you will be working with. ML problems start with data preferably, lots of data (examples or observations) for which you already know the target answer. Data for which you already know the target answer is called labelled data.

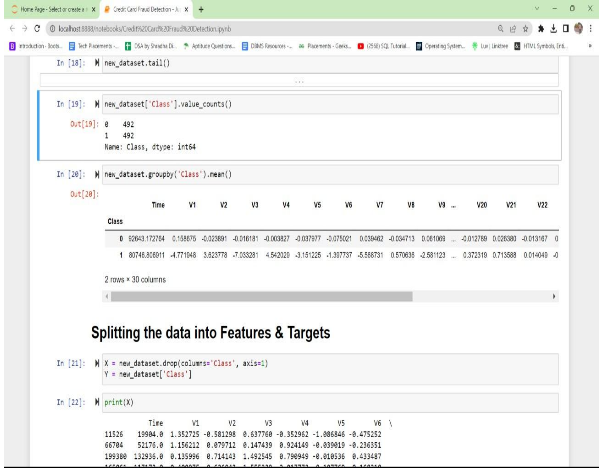
B. *Data Pre-Processing*

Pre-processing is the process of three important and common steps as follows:

1) Formatting: It is the process of putting the data in a legitimate way that it would be suitable to work with. Format of the data files should be formatted according to the need.Most recommended format is .csv files.

2) Cleaning: Data cleaning is a very important procedure in the path of data science as it constitutes the major part of the work. It includes removing missing data and complexitywith naming category and so on. For most of the data scientists, Data Cleaning continuesof 80% of work.

3) Sampling: This is the technique of analyzing the subsets from whole large datasets, which could provide a better result and help in understanding the behavior and pattern ofdata in an integrated way



```
In [18]:  new_dataset.tail()

          ...

In [19]:  new_dataset['Class'].value_counts()

Out[19]:  0    492
          1    492
          Name: Class, dtype: int64

In [20]:  new_dataset.groupby('Class').mean()

Out[20]:
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V20	V21	V22	
Class															
0	92643.172764	0.158675	-0.023891	-0.016181	-0.003827	-0.037977	-0.075021	0.039462	-0.034713	0.061069	...	-0.012789	0.026380	-0.013167	0
1	80746.806911	-4.771948	3.623778	-7.033281	4.542029	-3.151225	-1.397737	-5.568731	0.570636	-2.581123	...	0.372319	0.713588	0.014049	-0

2 rows × 30 columns

Splitting the data into Features & Targets

```
In [21]:  X = new_dataset.drop(columns='Class', axis=1)
          Y = new_dataset['Class']

In [22]:  print(X)

              Time        V1        V2        V3        V4        V5        V6  \
11526      19904.0  1.352725 -0.581298  0.637760 -0.352962 -1.086846 -0.475252
66704      52176.0  1.156212  0.079712  0.147439  0.924149 -0.039019 -0.236351
199380    132936.0  0.135996  0.714143  1.492545  0.790949 -0.010536  0.433487
```

Fig. 4

IV. TRAINING AND TESTING DATA

A. Data Visualization

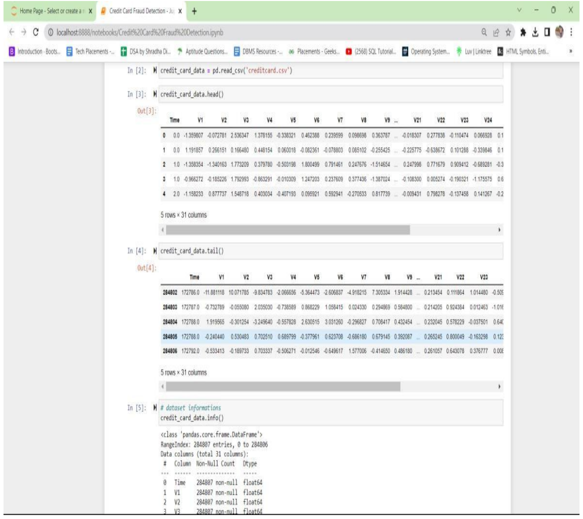
Data Visualisation is the method of representing the data in a graphical and pictorial way, data scientistsdepict a story by the results they derive from analysing and visualising the data. The best tool used is Tableau which has many features to play around with data and fetch wonderful results.

B. Feature Extraction

Feature extraction is the process of studying the behavior and pattern of the analyzed data and draw the features for further testing and training. Finally, our models are trained using the Classifier algorithm. Weuse classify module on Natural Language Toolkit library on Python. We use the labelled dataset gathered.The rest of our labelled data will be used to evaluate the models. Some machine learning algorithms were used to classify pre-processed data. The chosen classifiers were Random forest. These algorithms are verypopular in text classification tasks.

C. Evaluation Model

Model Evaluation is an essential part of the model development process. It helps to find the best model that represents our data and how well the selected model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can effortlessly generate overoptimistically and over fitted models. To avoid overfitting, evaluation methods such as hold out and cross-validations are used to test to evaluate model performance. The result will be in the visualized form. Representation of classified data in the form of graphs. Accuracy is well-defined as the proportion of precise predictions for the test data. It can be calculated easily by mathematical calculation i.e. dividing the number of correct predictions by the number of total predictions



```
In [2]:  credit_card_data = pd.read_csv('creditcard.csv')

In [3]:  credit_card_data.head()

Out[3]:
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.1
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.1
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.3
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.6
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.2

5 rows × 31 columns

```
In [4]:  credit_card_data.tail()

Out[4]:
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	
284802	172786.0	-11.881118	10.071785	-9.834783	-2.066656	-5.364473	-2.606837	-4.918215	7.305334	1.914428	...	0.213454	0.111864	1.014480	-0.50
284803	172787.0	-0.732789	-0.055080	2.035030	-0.738589	0.868229	1.058415	0.024330	0.294869	0.584800	...	0.214205	0.924384	0.012463	-1.01
284804	172788.0	1.919565	-0.301254	-3.249640	-0.557828	2.630515	3.031260	-0.296827	0.708417	0.432454	...	0.232045	0.578229	-0.037501	0.64
284805	172788.0	-0.240440	0.530483	0.702510	0.689799	-0.377961	0.623708	-0.686180	0.679145	0.392087	...	0.265245	0.800049	-0.163298	0.12
284806	172792.0	-0.533413	-0.189733	0.703337	-0.506271	-0.012546	-0.649617	1.577006	-0.414650	0.486180	...	0.261057	0.643078	0.376777	0.00

5 rows × 31 columns

```
In [5]:  # dataset informations
         credit_card_data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 284807 entries, 0 to 284806
Data columns (total 31 columns):
 #   Column  Non-Null Count   Dtype
---  ------  --------------   -----
 0   Time    284807 non-null  float64
 1   V1      284807 non-null  float64
 2   V2      284807 non-null  float64
 3   V3      284807 non-null  float64
```

Fig. 5

V. ALGORITHM

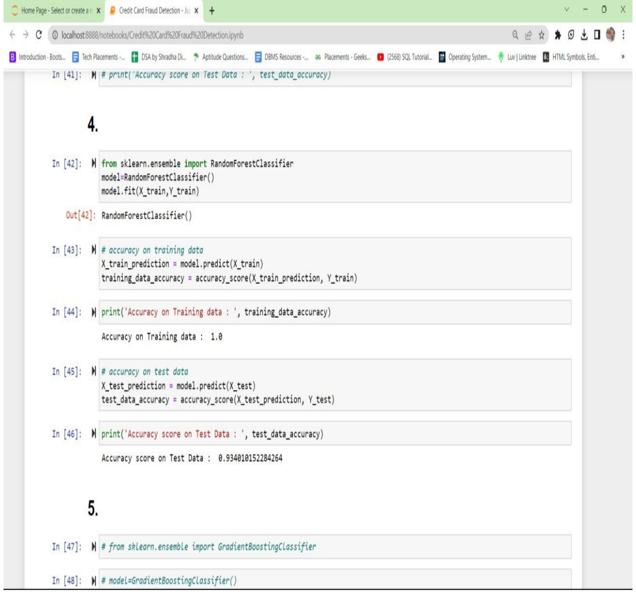
Random forest algorithm here used, which is a supervised machine learning algorithm based on ensemble learning. Ensemble learning is an algorithm where the predictions are derived by assembling or bagging different models or similar model multiple times. The random forest algorithm works in a similar way and uses multiple algorithm i.e. multiple decision trees, resulting in a forest of trees, hence the name "Random Forest". The random forest algorithm can beused for both regression and classification tasks.

1) The random forest algorithm is not biased and depends on multiple trees where eachtree is trained separately based on the data, therefore biasedness is reduced overall.

2) It's a very stable algorithm. Even if a new data point is introduced in the dataset itdoesn't affect the overall algorithm rather affect the only a single tree.

3) It works well when one has both categorical and numerical features.

4) The random forest algorithm also works well when data possess missing values, or when it's not been scaled properly. Thus, using this Random forest algorithm and decision trees algorithm we have extracted the accurate percentage of detection of fraud from the given dataset by studying its behavior. A confusion matrix is basically a summary of prediction results or a table which is used to describe the performance of the classifier on a set of test data where true values are known. It provides visualization of analgorithm's performance and allows easy identification of classes. Thus, resulting in the computing of most performance measures by giving insights not only the errors being made by the classification model but also tells the type of errors being made. Trained Data and Testing Data is represented in a confusion matrix which portrays:



```
In [41]:  # print('Accuracy score on Test Data : ', test_data_accuracy)
```

4.

```
In [42]:  from sklearn.ensemble import RandomForestClassifier
          model=RandomForestClassifier()
          model.fit(X_train,Y_train)

Out[42]:  RandomForestClassifier()

In [43]:  # accuracy on training data
          X_train_prediction = model.predict(X_train)
          training_data_accuracy = accuracy_score(X_train_prediction, Y_train)

In [44]:  print('Accuracy on Training data : ', training_data_accuracy)

          Accuracy on Training data :  1.0

In [45]:  # accuracy on test data
          X_test_prediction = model.predict(X_test)
          test_data_accuracy = accuracy_score(X_test_prediction, Y_test)

In [46]:  print('Accuracy score on Test Data : ', test_data_accuracy)

          Accuracy score on Test Data :  0.934010152284264
```

5.

```
In [47]:  # from sklearn.ensemble import GradientBoostingClassifier

In [48]:  # model=GradientBoostingClassifier()
```

Fig. 6

VI. CONCLUSION

Hence, we have acquired the result of an accurate value of credit card fraud detection i.e. 0.9994802867383512 (99.93%) using a random forest algorithm with new enhancements. In comparison to existing modules, this proposed module is applicable for the larger dataset and provides more accurate results. The Random Forest algorithm will provide better performance with many training data, but speed during testing and application will still suffer. Usage of more pre-processing techniques would also assist. Our future work will try torepresent this into a software application and provide a solution for credit card fraud using the new technologies like Machine Learning, Artificial Intelligence and Deep Learning.

REFERENCES

[1] P. Yogendra Prasad; A Sreni Chowdary; Cherapalli Bavitha; Earagaraju Mounisha; Chatna Reethika, "A Comparison Study of Fraud Detection in Usage of Credit Cards using Machine Learning",2023,7th International Conference on Trends in Electronics and Informatics (ICOEI)

[2] C.H Sumanth; Pokala Pavan Kalyan; Bolisetti Ravi; S Balasubramani., "Analysis of Credit Card Fraud Detection using Machine Learning Techniques",2022, 7th International Conference on Communication and Electronics Systems (ICCES)

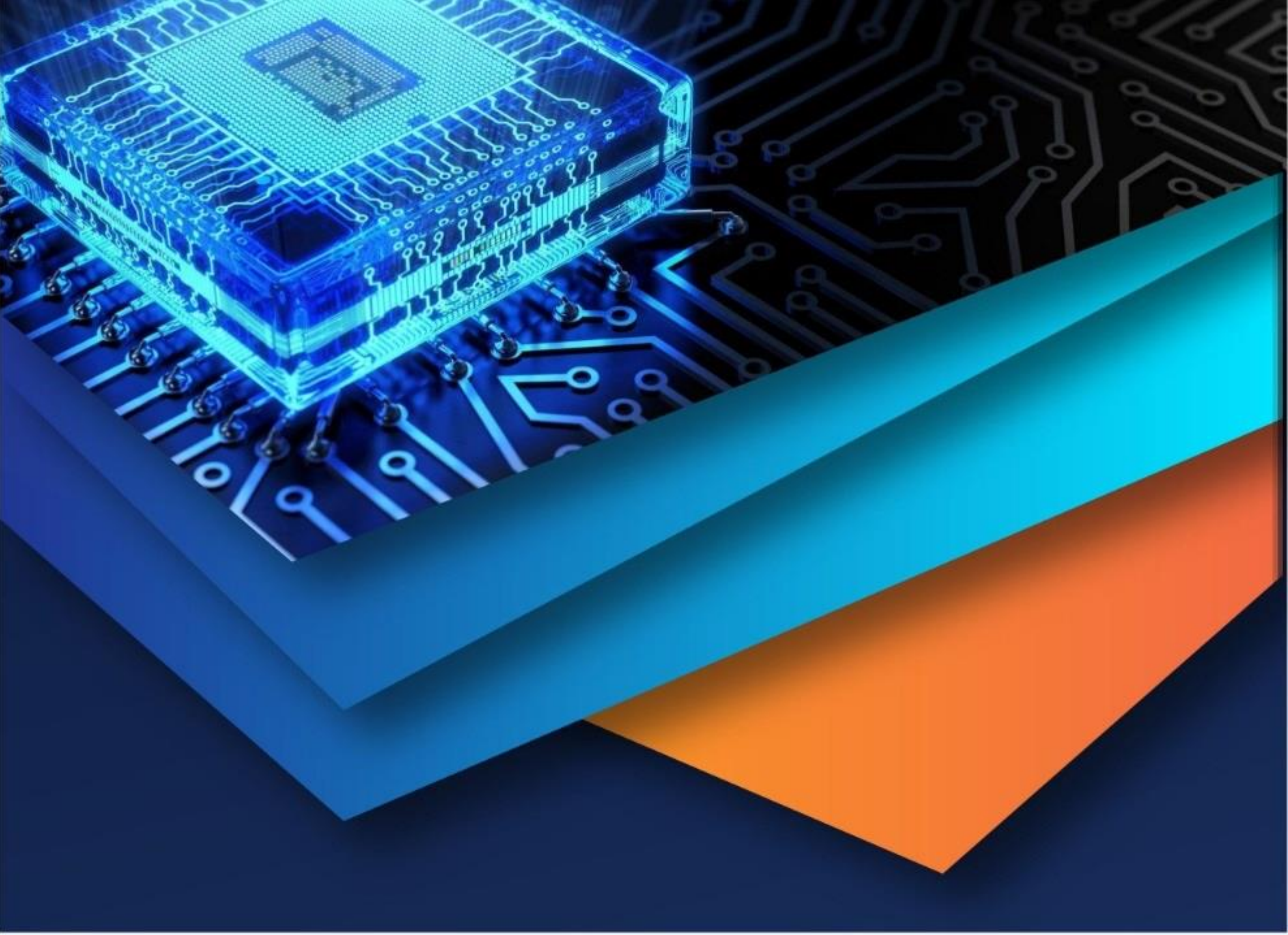
[3] Gupta, Shalini, and R. Johari. "A New Framework for Credit Card Transactions Involving Mutual Authentication between Cardholder and Merchant." International Conference on Communication Systems and Network Technologies IEEE, 2021:22-26.

[4] Y. Gmbh and K. G. Co, "Global online payment methods: the Full year 2020," Tech.Rep., 3 2020.

[5] Bolton, Richard J., and J. H. David. "UnsupervisedProfiling Methodsfor FraudDetection." Proc Credit Scoring and Credit Control VII (2020): 5– 7.

[6] Drummond, C., and Holte, R. C. (2019). C4.5, class imbalance, and cost sensitivity: why under-sampling beats oversampling. Proc of the ICML Workshop on Learning fromImbalanced Datasets II, 1–8.

[7] Quah, J. T. S., and Sriganesh, M. (2020). Real-time credit card fraud detection usingcomputational intelligence. Expert Systems with Applications, 35(4), 1721-1732



10.22214/IJRASET

45.98

IMPACT FACTOR:
7.129

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)