Credit Card Fraud Detection Using Random Forest Algorithm

Hardik Jethava[1], Feon Jaison[2]
[1]*Department of Master of Computer Applications School of CS&IT Jain (Deemed-to-be-University) Bangalore, India*
[2]*Associate Professor, Department of Master of Computer Application School of CS&IT Jain (Deemed-to-be-University) Bangalore, India*

***Abstract*— Financial fraud is a constant threat to the financial industry, with far-reaching effects. In order to detect credit card fraud in internet transactions, data mining was essential. Due to two fundamental causes, detecting credit card fraud, which is a data mining challenge, gets difficult: first, genuine and fraudulent behaviours profiles are constantly changing, and second, credit card fraud data sources are extremely skewed. The methodology used to gather the dataset, the parameters used, and the method(s) utilised to identify fraud in credit card transactions all have an influence on the accuracy of fraud detection. The performance of several classification algorithms on significantly unbalanced credit card fraud data is investigated in this study. This study collected a total of 284,807 credit card transactions from European consumers. On the skewed statistics, a hybrid strategy of under and up sampling is used. The raw and transformed data are subjected to the one approach. To finish the task, Python is used. The techniques' accuracy tested.**

Index Terms: **Dataset, Credit card, Random Forest Algorithm.**

I.INTRODUCTION

in order to prevent credit card fraud, The use of data mining techniques is one of the most effective ways to detect credit theft. There are a few financial frauds is becoming more prevalent in the financial sector, businesses, and government, with far-reaching implications. Theft is defined as a deliberate use of deception for the purpose of obtaining financial benefit. Credit card transactions have increased as people become more reliant on web-based technologies. Credit card fraud is on the rise, owing to the fact that credit cards are the most popular method of payment for both print and virtual purchases. Inner card fraud and exterior card fraud are the two types of credit card fraud. Internal card fraud happens when customers and banks work together to conduct fraud using a fictitious identity, and outer card fraud happens when a stolen credit card is exploited to obtain money through dubious means. The most prevalent sort of credit card fraud, exterior card fraud, has sparked a lot of interest. Detecting fraudulent transactions via manual method detection methods takes a lot of time and is inefficient; however, manual processes have become obsolete as a result of the emergence of big data. On the other hand, financial institutions have focused their efforts on two sorts of procedures for detecting fraudulent transactions: valid (genuine) and illegal (false) transactions. Identifying credit card fraud entails examining a card's purchase history.

II. OBJECTIVES

Our goal is to use machine learning models to classify credit card fraud as accurately as possible using data collected across Europe over two days in September 2013. We decided we'd use a Random Forest Algorithm, after doing some preliminary data analysis. One of the first issues we saw was the huge disparity in the set of data: frauds account for only 0.172 percent of all fraud transactions. In this scenario, false negatives in our forecasts are far worse than false positives, because systematic errors indicate that somebody keeps getting away with credit card fraud. False - positive, on the other hand, add to the complexity and potential inconvenience of a cardholder having to confirm that they did, in fact, close the deal (and not a thief)

III. LITERATURE SURVEY

SonalMehndiratta [1] developed “Credit Card Fraud Detection Techniques” This research looks at a range of malware detection methods based on a set of parameters in this paper. By collecting historical data, predictive analysis methodologies can be utilised to detect fraud. Among the techniques used are Markov Chain Designs, Genetic Algorithms, Convolutional Neural, Naive Bayes, and KNN classifiers. Fraud prediction is performed in this study primarily through two phases: features extraction and classification. It is proposed to utilise a hybrid technique to detect credit card fraud in the near future.

Kuldeep Randhwa Et.al [2] established “Credit card fraud detection using AdaBoost and majority voting” created a credit card fraud detection system based on machine learning Popular models were first used, but hybrid methods including such Classifiers and qualified majority procedures emerged. To evaluate the model's performance, a publicly available set was used. While a scam was being examined, another dataset from the financial institution was used. The noise was then introduced to the data set, allowing the algorithm's toughness to be evaluated. The approaches were based on theoretical results that suggest the majority of the voting systems are accurate at detecting credit card fraud. The sample dataset has been injected with noise ranging from 10% to 30% for some further testing of the hybrid models. Several voting strategies received a good rating of 0.942 for 30 percent higher noise. As a conclusion, it was discovered that the voting system worked well even when there was a lot of noise.

Krishana Modi.Et.al [3] developed “Review on Fraud Detection Methods in Credit Card Transaction” The study tested and contrasted various methods for detecting fraudulent transactions. Any of these approaches, or a combination of them, can be used to detect fraudulent behaviour. By adding new features to the model, it may be possible to improve its accuracy. Financial institutions employ data collection in a different style to detect fraud behaviour. Any of these ways can be used to determine a client's regular user behaviour based on previous behaviour. As a result, below is a survey of many detection strategies that were given over time.

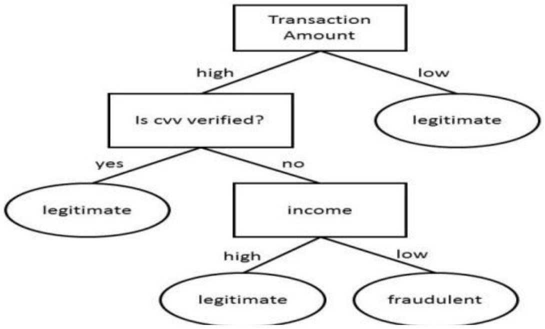
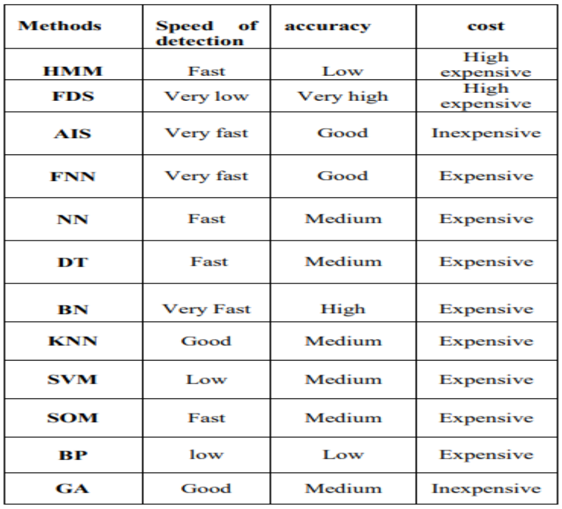


Figure 1 Example of Decision Tree

The distinction between a real and a fake transaction is depicted in Figure 1.

John O. Awoyemi [4] developed “Credit card fraud detection using machine learning techniques: A comparative analysis” conducted a study in which the effectiveness of a different of algorithms was assessed when they were applied to skewed credit card fraud data. The 284,807 activities of European cardholders were used to construct the credit card transaction data collection. A hybrid method of the under and up - sampling is used to deal with the skewed statistics. Python has three alternative techniques for raw and post data. Accuracy, sensitivity, accuracy, balanced KNN classifier, and other metrics are used to evaluate the performance of these approaches. The results imply that k-NN outperforms naïve Markov and logistic regression techniques in terms of performance.

Comparison of different methods



Methods	Speed of detection	accuracy	cost
HMM	Fast	Low	High expensive
FDS	Very low	Very high	High expensive
AIS	Very fast	Good	Inexpensive
FNN	Very fast	Good	Expensive
NN	Fast	Medium	Expensive
DT	Fast	Medium	Expensive
BN	Very Fast	High	Expensive
KNN	Good	Medium	Expensive
SVM	Low	Medium	Expensive
SOM	Fast	Medium	Expensive
BP	low	Low	Expensive
GA	Good	Medium	Inexpensive

Table-1 Comparison of different methods

Table 1 shows a comparison of several approaches, with detection speed, efficiency, and cost taken into account.

IV. PROPOSED SYSTEM

For dataset classification and regression, the proposed system employs the Random Forest Algorithm and Neural Networks. We will first gather the Credit Card dataset, after which we will perform analysis on it. Following the analysis of the dataset, the dataset must be cleaned.In most datasets, there'll be many similar and null values, hence a cleanup process is required to eliminate all of those identical and null entries. Then, in order to compare and analyze the dataset, we must divide it into two categories: Training dataset and Testing dataset. After partitioning the dataset, we must use the Random Forest Algorithm, which will provide us with a higher level of accuracy when it comes to credit card fraud activities.using the Random Forest Algorithm, which will be presented in the form of a confusion matrix. an effectiveness analysis will be conducted based on the data classification described above. The reliability of credit card fraud activities may be achieved through this analysis, which will then be shown in the form of a graphical depiction.

A. Software Requirement

- Python 3.9.2:

Figure 2 Python 3.9.2

Python is high-level, general-purpose programming language that is interpreted. The use of considerable indentation in its design philosophy emphasises code readability. Its language elements and object-oriented approach are aimed at assisting programmers in writing clear, logical code for both small and large-scale projects.

- PyCharm Community Edition

Figure 3 PyCharm Community Edition

PyCharm is an integrated development environment for computer programming, with a focus on the Python programming language. JetBrains, a Czech firm, developed it.

- NumPy 1.20

Figure 4 NumPy 1.20

NumPy is a Python library that adds support for huge, multi-dimensional arrays and matrices, as well as a large handful of high arithmetic operations to operate on these arrays.

- Pandas

Figure 5 Pandas

pandas are a data manipulation and analysis software package for the Python programming language.

- Scikit-learn

Figure 6 Scikit-learn

Scikit-learn is a Python-based machine learning library that is available for free. It includes support vector machines as well as other classification, regression, and clustering methods.

V. WORKING SYSTEM

This architecture technique is utilized using a Kaggle dataset that we already have. The dataset we'll be using is available for download on Kaggle. It comprises information on credit card transactions that occurred in the process of two days, with 492 fraudulent transactions out of 284,807 total. The dataset's variables are all numerical. Due to privacy concerns, the data has been altered using PCA transformation(s). Time and Amount are the only two features that haven't altered. The seconds passed between every transaction and also the first transaction in the dataset is stored in time.

After that, we partitioned the dataset into two parts: training dataset and test dataset. The training data is a set of data used to teach a software how to learn and deliver advanced results using technologies such as neural networks. The Test Dataset is a sample of data that is used to offer an unbiased assessment of a final model's fit on the training dataset.

Then there are Algorithms to train. In our system, we use one algorithm: the random forest algorithm

A.RANDOM FOREST ALGORITHM

Random Forest is also known as Random Decision Forest (RFA), and it is used for classifying, regression, and other tasks that require many decision trees to be constructed. This Random Forest Algorithm is based on classifier, and it has the advantage of being able to perform both classification and regression. When compared to all other existing systems, the Random Forest Algorithm provides superior accuracy, and it is the most often used algorithm. The application of the RandomForest algorithm in credit card fraud detection can offer you an accuracy of around 99 percent, according to this article.

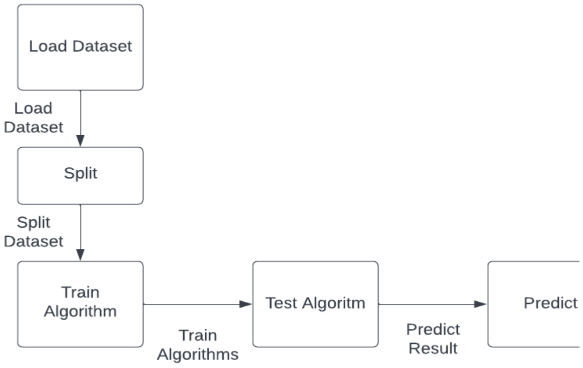


Figure 7 System Architecture

Figure 7 shows how we collect all of the data and then load it into two parts: trained dataset and test dataset. The training dataset and test datasetare used to train the random forest algorithm and the test algorithm will test all the data that it is fraud transaction and clean transaction and after that it predict the fraud transaction in graph

VI. METHODOLOGY

Step 1: First, we load the dataset that we had previously downloaded from Kaggle

Step 2: Next, we split the dataset into two parts: training and test.

Step 3: Then it's time to train the algorithm (we're using the random forest algorithm).

Step 4: after train algorithm it shows the accuracy of the algorithm

Step 5: we have to upload dataset and it detect fraud transaction from dataset and see total test transaction with clean and fraud signature.

Step 6: after the detect fraud from dataset we can generate in graphical format

VII. RESULTS

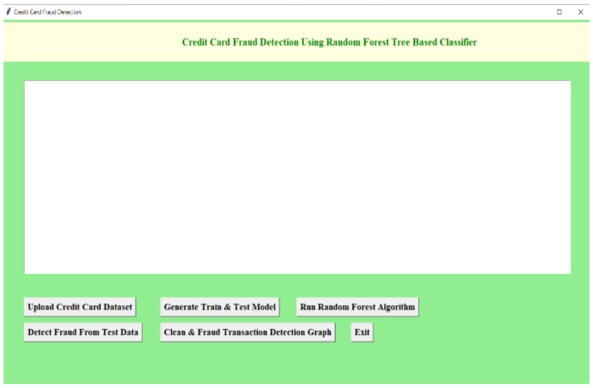
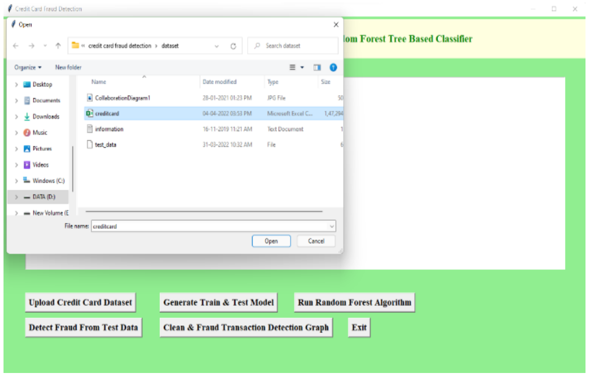

Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit

Figure8 Initial Phase

Figure 8 depicts the initial stage of the project, which is a desktop programmed that looks like this.

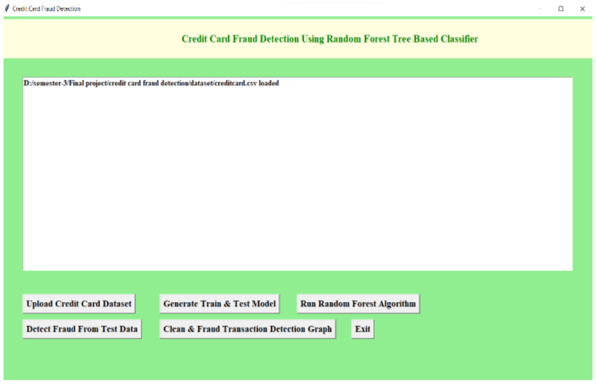

Credit Card Fraud Detection
Open
credit card fraud detection › dataset
Search dataset
Organize
New folder
Desktop
Documents
Downloads
Music
Pictures
Videos
Windows (C:)
DATA (D:)
New Volume (E
Open
Cancel
om Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit

Name	Date modified	Type	Size
CollaborationDiagram1	28-01-2021 01:23 PM	JPG File	50
creditcard	04-04-2022 03:53 PM	Microsoft Excel C...	1,47,294
information	16-11-2019 11:21 AM	Text Document	1
test_data	31-03-2022 10:32 AM	File	6

File name: creditcard

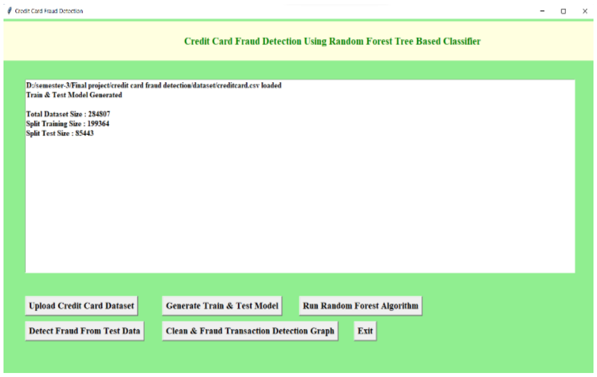
Figure 9 Upload Credit Card Dataset

Figure 9 shows that we must upload the credit card dataset that we have already downloaded and saved in the project, so we must upload it and access it.After uploading dataset will get Figure 10


Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit

```
D:/semester-3/Final project/credit card fraud detection/dataset/creditcard.csv loaded
```

Figure 10Credit Card Dataset

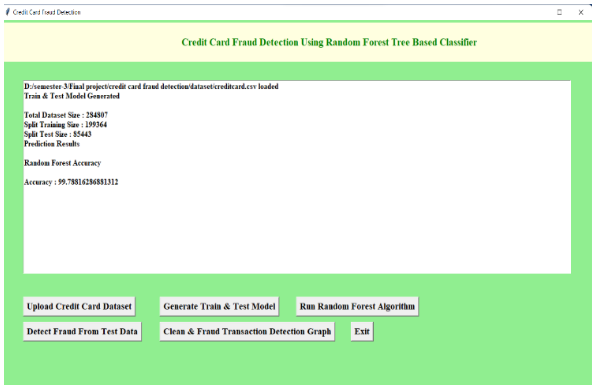

Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit

```
D:/semester-3/Final project/credit card fraud detection/dataset/creditcard.csv loaded
Train & Test Model Generated

Total Dataset Size : 284807
Split Training Size : 199364
Split Test Size : 85443
```

Figure 11 Generate Train & Test Model

We divided the dataset into two sections in Figure 11 the training dataset and the test dataset. We're working with a total of 284807 data points. The train dataset after splitting is 199364, while the test dataset is 85443.


Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit

```
D:/semester-3/Final project/credit card fraud detection/dataset/creditcard.csv loaded
Train & Test Model Generated

Total Dataset Size : 284807
Split Training Size : 199364
Split Test Size : 85443
Prediction Results

Random Forest Accuracy

Accuracy : 99.78816286881312
```

Figure 11 Run Random Forest Algorithm

In Figure 11 we can see Random Forest generate 99.78% percent accuracy while building model on train and test data.

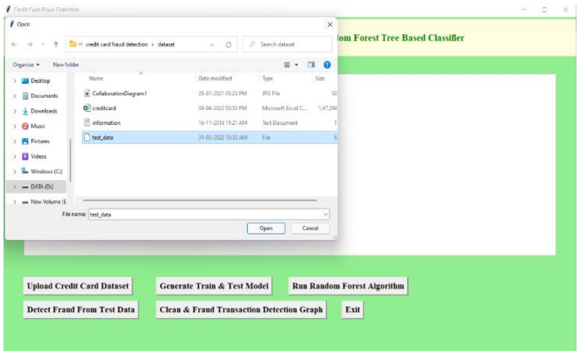
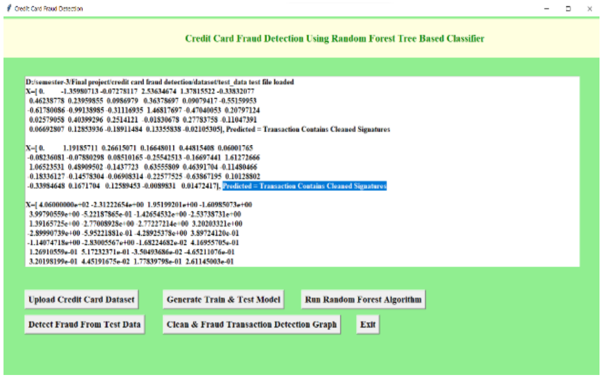


Figure 12 Detect Fraud from Test Data

click on 'Detect Fraud from Test Data' button to upload test data and to predict whether test data contains normal or fraud transaction


Credit Card Fraud Detection
Credit Card Fraud Detection Using Random Forest Tree Based Classifier
Upload Credit Card Dataset
Generate Train & Test Model
Run Random Forest Algorithm
Detect Fraud From Test Data
Clean & Fraud Transaction Detection Graph
Exit

```
D:/semester-3/Final project/credit card fraud detection/dataset/test_data test file loaded
X=[ 0.         -1.35980713 -0.07278117  2.53634674  1.37815522 -0.33832077
  0.46238778  0.23959855  0.0986979   0.36378697  0.09079417 -0.55159953
 -0.61780086 -0.99138985 -0.31116935  1.46817697 -0.47040053  0.20797124
  0.02579058  0.40399296  0.2514121  -0.01830678  0.27783758 -0.11047391
  0.06692807  0.12853936 -0.18911484  0.13355838 -0.02105305], Predicted = Transaction Contains Cleaned Signatures

X=[ 0.          1.19185711  0.26615071  0.16648011  0.44815408  0.06001765
 -0.08236081 -0.07880298  0.08510165 -0.25542513 -0.16697441  1.61272666
  1.06523531  0.48909502 -0.1437723   0.63555809  0.46391704 -0.11480466
 -0.18336127 -0.14578304 -0.06908314 -0.22577525 -0.63867195  0.10128802
 -0.33984648  0.1671704   0.12589453 -0.0089831   0.01472417], Predicted = Transaction Contains Cleaned Signatures

X=[ 4.06000000e+02 -2.31222654e+00  1.95199201e+00 -1.60985073e+00
  3.99790559e+00 -5.22187865e-01 -1.42654532e+00 -2.53738731e+00
  1.39165725e+00 -2.77008928e+00 -2.77227214e+00  3.20203321e+00
 -2.89990739e+00 -5.95221881e-01 -4.28925378e+00  3.89724120e-01
 -1.14074718e+00 -2.83005567e+00 -1.68224682e-02  4.16955705e-01
  1.26910559e-01  5.17232371e-01 -3.50493686e-02 -4.65211076e-01
  3.20198199e-01  4.45191675e-02  1.77839798e-01  2.61145003e-01
```

Figure 13 Detect Fraud from Test Data

uploading test dataset and after uploading test data will get prediction details which shows in figure 13.beside each test data application will display output as whether transaction contains cleaned or fraud signatures.

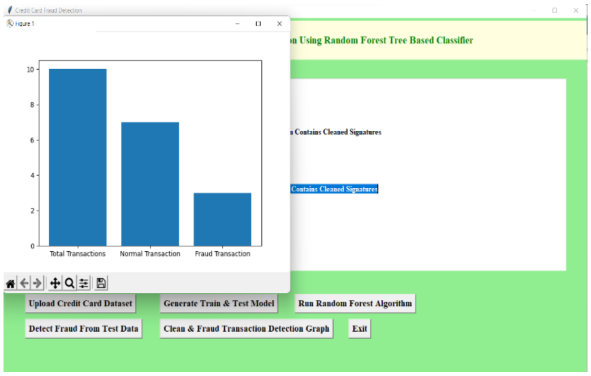


Figure 14 Fraud Transaction Detection Graph

Figure 14 shows the total test data and number of normal and fraud transaction detected. In above graph x-axis represents type and y-axis represents count of clean and fraud transaction

VIII. CONCLUSION

Credit card fraud is unquestionably a form of criminal deception. This article evaluated recent results in this field and outlined the most common types of fraud, as well as how to detect them. This paper also includes a detailed explanation of how machine learning can be used to improve fraud detection findings, as well as the algorithm, pseudocode, explanation, and experimentation results. While the method achieves a precision of over 99.7%, when only a tenth of the set of data is considered, it only achieves a precision of 28%.When the complete dataset is given into the system, however, the precision increases to 33%. Due to the large disparity between both the number of legal and authentic transactions, this large percentage of correctness is to be expected. Because the complete dataset is made up of only two days' worth of transaction records, it's only a small portion of the data that could be made public if this research were to be used commercially. Because the application is predicated on machine learning methods, it will only get more efficient over time as more data is sent into it.

While we didn't achieve our target of 100 % accuracy in detecting fraud, we did create a system that can get extremely close to it given enough time and data. As with any effort of this nature, there is potential for improvement. Because of the nature of this project, multiple algorithms can be merged as modules and their findings mixed to improve the quality of the result output.

REFERENCE

[1] Anuruddha Thennakoon; Chee Bhagyani; Sasitha Premadasa; ShalithaMihiranga; Nuwan Kuruwitaarachchi 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence);10-11 Jan. 2019

[2] John O. Awoyemi; Adebayo O. Adetunmbi; Samuel A. Oluwadare; 2017 International Conference on Computing Networking and Informatics (ICCNI) 29-31 Oct. 2017.

[3] DejanVarmedja; Mirjana Karanovic; SrdjanSladojevic; Marko Arsenovic; Andras Anderla; 2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH) 20-22 March 2019

[4] PranaliShenvi; Neel Samant; Shubham Kumar; Vaishali Kulkarni; 2019 IEEE 5th International Conference for Convergence in Technology (I2CT) 29-31 March 2019

[5] Deepti Dighe; Sneha Patil; Shrikant Kokate; 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)16- 18 Aug. 2018

[6] Krishna Modi; Reshma Dayma; 2017 International Conference on Intelligent Computing and Control (I2C2); 23-24 June 2017

[7] S P Maniraj;AdityaSaini;Shadab Ahmed ; International Journal of Engineering and Technical Research 08(09); September 2019;vol 08;page no. 110-115.

[8] S. Abinayaa, H. Sangeetha, R. A. Karthikeyan, K. Saran Sriram, D. Piyush; International Journal of Engineering and Advanced Technology (IJEAT) ;4, April, 2020; vol 09; page no. 1199-1201.

[9] K.RatnaSree Valli , P.Jyothi , G.Varun Sai , R.Rohith Sai Subash; Quest Journals Journal of Research in Humanities and Social Science; 2 June 2019;Volume 8; page no: 04-11 .

[10] Lakshmi S V; Selvani Deepthi Kavila; International Journal of Applied Engineering Research ISSN; 04 November 2018; Volume 13, pp. 16819-16824