International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

Machine Learning Algorithms For Fraud Detection In Blockchain

M Praveen Reddy [1],A Bhavesh Satya Sai Likhith[2], Kolakani Srikanth[3],P Chandravikas[4], P Shiva Sai Goud[5]
[2,3,4,5] UG Scholars, Department of CSE, **AVN Institute of Engineering and Technology,**Hyderabad, Telangana, India.
[1]Assistant Professor, Department of CSE, **AVN Institute of Engineering and Technology**, Hyderabad, Telangana, India.

ABSTRACT :

Fraudulent transactions have a huge impact on the economy and trust of a blockchain network. Consensus algorithms like proof of work or proof of stake can verify the validity of the transaction but not the nature of the users involved in the transactions or those who verify the transactions. This makes a blockchain network still vulnerable to fraudulent activities. One of the ways to eliminate fraud is by using machine learning techniques. Machine learning can be of supervised or unsupervised nature. In this paper, we use various supervised machine learning techniques to check for fraudulent and legitimate transactions. We also provide an extensive comparative study of various supervised machine learning techniques like decision trees, Naive Bayes, logistic regression, multilayer perceptron, and so on for the above task .

INTRODUCTION :

The problem of detecting fraudulent transactions is being studied for a long time. Fraudulent transactions are harmful to the economy and discourage people from investing in bitcoins or even trusting other blockchain-based solutions. Fraudulent transactions are usually suspicious either in terms of participants involved in the transaction or the nature of the transaction. Members of a blockchain network want to detect Fraudulent transactions as soon as possible to prevent them from harming the blockchain network's community and integrity. Many Machine Learning techniques have been proposed to deal with this problem, some results appear to be quite promising [4], but there is no obvious superior method. This paper compares the performance of various supervised machine learning models like SVM, Decision Tree, Naive Bayes, Logistic Regression, and few deep learning models in detecting fraudulent transactions in a blockchain network. Such comparative study will help decide the best algorithm based on accuracy and computational speed trade-off. Our goal is to see which users and transactions have the highest probability of being involved in fraudulent transactions.

EXITING SYSTEM :

We applied eight different supervised learning algorithms to the dataset. The dataset contains information about trust on different nodes or ratings given to them. This information is useful in detecting if a certain node's transaction can be fraudulent or not. The following table summarizes the accuracy obtained in each case.

PROPOSED SYSTEM :

The workflow for detecting fraudulent activity is summarised in Figure 1. Essentially, after the Blockchain network has approved a transaction after all basic checks, our proposed system kicks in and does additional checks to detect if the transaction can be fraudulent. This approach makes sure that there is no extra overhead of even checking the transactions that the Blockchain network itself can easily invalidate.

The work done can be divided mainly into three phases:

1. Preprocessing phase

2. Building and training various models

3. Performance evaluation of all the models.

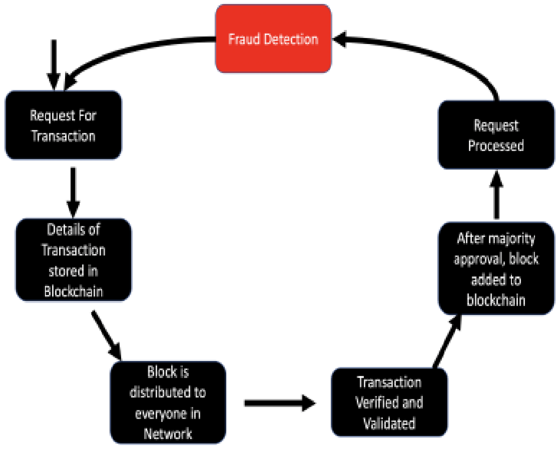

Fraud Detection
Request For Transaction
Details of Transaction stored in Blockchain
Block is distributed to everyone in Network
Transaction Verified and Validated
After majority approval, block added to blockchain
Request Processed

Fig. 1. Workflow of applying check for Fraud Detection

We preprocess using node-embedding in the network using the node2vec algorithm. Then, we read and convert the shorter version of concatenated rating dataset into a dataframe. Then, we create a function for the perception store features. This function extracts the features of a node using the ”source” and ”target” columns of the dataset. These features are then stored in a CSV file. We then run the node2vec algorithm in python and create a dictionary of nodes and corresponding embeddings. We also create a network edge list file and then reduce embeddings dimensionality for 2D projections. This dimensionnality reduction can be obtained using algorithms like t-SNE.

We then normalize the features extracted from the node2vec algorithm and create a file that contains the normalized values. We assign a score of 1 if the transaction is rated badly (fraud) and 0 otherwise. We then calculate the mean and standard deviation of the node features and save it to a CSV file. We then divide all our obtained data into train and test sets.

Phase II- Building various models, training and testing them.

We divide our data into train(0.8) and test(0.2) data. We then check the ratio of fraudulent and honest transactions in our train and test sets. We use machine following machine learning and deep learning models to predict if a transaction is fraudulent:

1.Logistic Regression: This is a simple linear classifier. Logistic regression works well for binary classification problems.

2. Multilayer Perceptron: Multilayer perceptron helps in separation data that cannot be classified using a linear classifier by introducing non linearity.

3. Naive Bayes: This model uses the Bayes theorem to calculate the probability of a transaction being fraudulent.

4. Adaboost: This is an ensemble learning method to boost the performance of binary classifiers.

5. Decision Tree: This classifier has a sequence of conditions and questions on data based on various features.

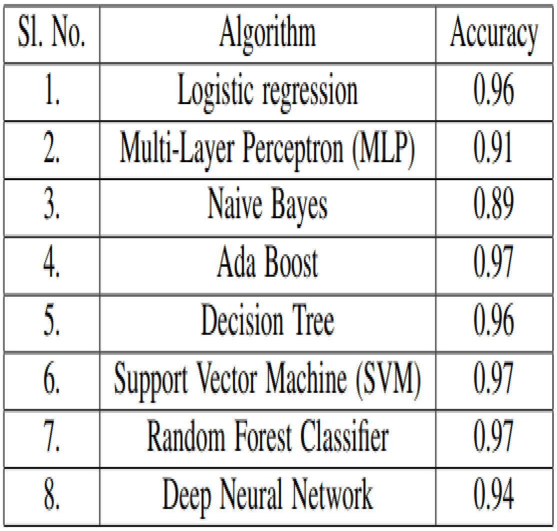
6. SVM: It uses a kernel method to transform the data in the dataset, and based on these transitions, it finds a boundary between all possible outputs.

7. Random Forest Classifier: This classifier fits a number of decision trees on small batches of the dataset.

8. Neural Network: This model consists of six dense layers and four hidden layers. Relu and sigmoid were used as activation functions.

Phase III-Evaluation of models on test set

We evaluate all our classification models using bootstrap sampling. In machine learning, bootstrap sampling involves drawing sample data with replacement from the dataset to estimate a parameter. So we first choose the number of bootstrap samples. Then, we choose the sample size. Then, for each bootstrap sample, we draw a sample with chosen bootstrap size (with replacement) and test the sample's data. For this purpose, we use the accuracy metric, which is a standard metric used in machine learning problems. We then take the mean of all accuracies obtained in this fashion to evaluate the skill of our model.

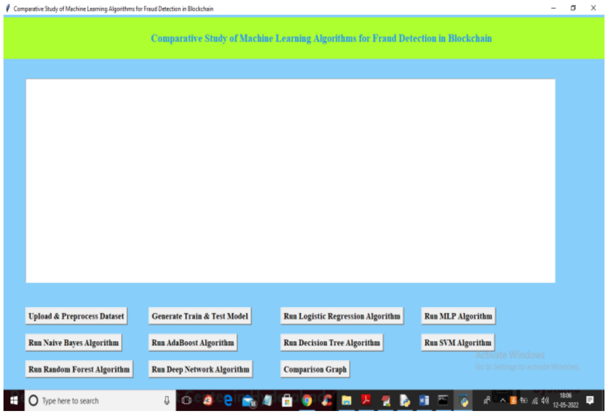


Sl. No.	Algorithm	Accuracy
1.	Logistic regression	0.96
2.	Multi-Layer Perceptron (MLP)	0.91
3.	Naive Bayes	0.89
4.	Ada Boost	0.97
5.	Decision Tree	0.96
6.	Support Vector Machine (SVM)	0.97
7.	Random Forest Classifier	0.97
8.	Deep Neural Network	0.94

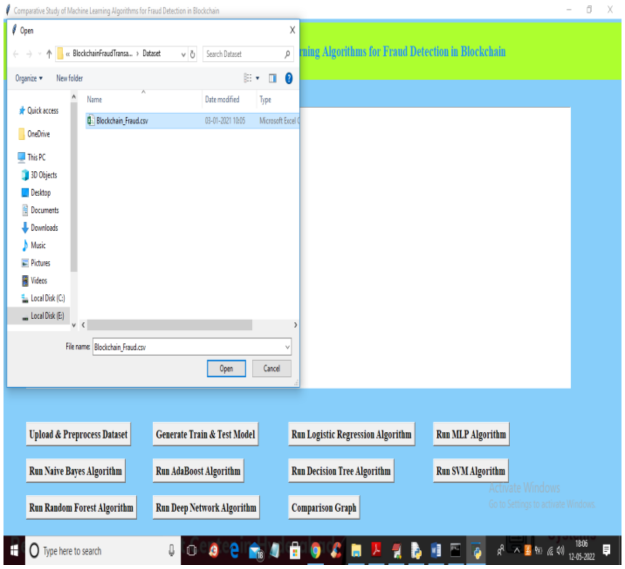
We observed that using Ada Boost, SVM, and Random Forest classifier gave the best results among the seven different algorithms. Also, since these algorithms already provide an accuracy of 97% we would like to build a fraud detector that will use the scores and decisions from the three algorithms together to decide if a transaction is fraudulent or not finally.

SCREENSHOTS :

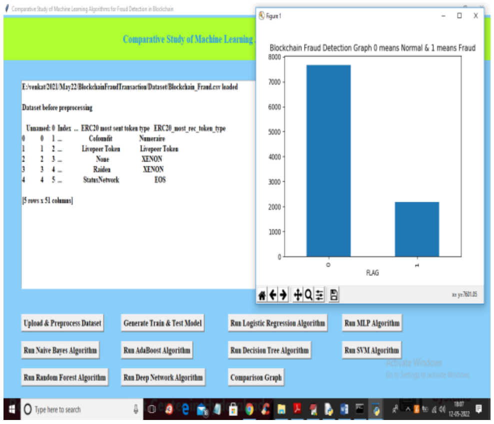
To run project double click on 'run.bat' file to get below screen


Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Upload & Preprocess Dataset
Generate Train & Test Model
Run Logistic Regression Algorithm
Run MLP Algorithm
Run Naive Bayes Algorithm
Run AdaBoost Algorithm
Run Decision Tree Algorithm
Run SVM Algorithm
Run Random Forest Algorithm
Run Deep Network Algorithm
Comparison Graph
Type here to search

In above screen click on 'Upload & Preprocess Dataset' button to upload and read dataset and then remove missing values


Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Open
BlockchainFraudTransa... › Dataset
Search Dataset
Organize
New folder
Quick access
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
Name
Date modified
Type
Blockchain_Fraud.csv
03-01-2021 10:05
Microsoft Excel C
File name: Blockchain_Fraud.csv
Open
Cancel
Upload & Preprocess Dataset
Generate Train & Test Model
Run Logistic Regression Algorithm
Run MLP Algorithm
Run Naive Bayes Algorithm
Run AdaBoost Algorithm
Run Decision Tree Algorithm
Run SVM Algorithm
Run Random Forest Algorithm
Run Deep Network Algorithm
Comparison Graph
Type here to search

In above screen selecting and uploading dataset and then click on 'Open' button to load dataset and get below output


Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Comparative Study of Machine Learning
Figure 1
Blockchain Fraud Detection Graph 0 means Normal & 1 means Fraud
8000
7000
6000
5000
4000
3000
2000
1000
0
0
1
FLAG
Upload & Preprocess Dataset
Generate Train & Test Model
Run Logistic Regression Algorithm
Run MLP Algorithm
Run Naive Bayes Algorithm
Run AdaBoost Algorithm
Run Decision Tree Algorithm
Run SVM Algorithm
Run Random Forest Algorithm
Run Deep Network Algorithm
Comparison Graph
Type here to search

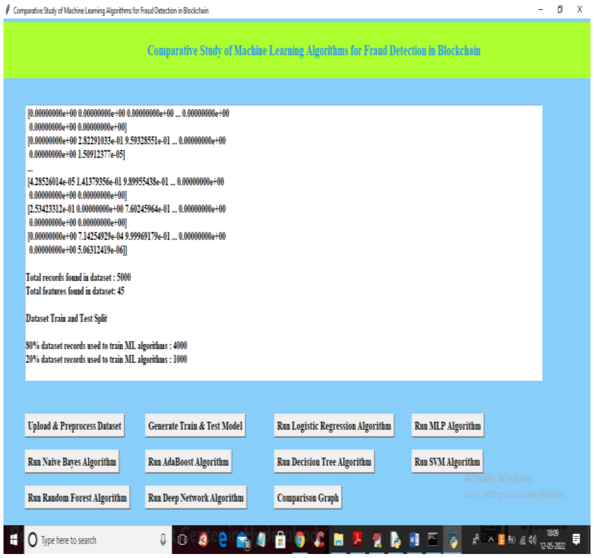
```
E:/venkat/2021/May22/BlockchainFraudTransaction/Dataset/Blockchain_Fraud.csv loaded

Dataset before preprocessing

  Unnamed: 0  Index  ...  ERC20 most sent token type  ERC20_most_rec_token_type
0          0      1  ...                   Cofoundit                  Numeraire
1          1      2  ...               Livepeer Token             Livepeer Token
2          2      3  ...                        None                     XENON
3          3      4  ...                      Raiden                     XENON
4          4      5  ...               StatusNetwork                       EOS

[5 rows x 51 columns]
```

In above screen dataset loaded and dataset contains some non-numeric data and ML algorithms will not take such data so we need to remove and graph x-axis contains type of transaction and y-axis contains number of records and now close above graph and then click on 'Generate Train & Test Model' button to get below output


Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Upload & Preprocess Dataset
Generate Train & Test Model
Run Logistic Regression Algorithm
Run MLP Algorithm
Run Naive Bayes Algorithm
Run AdaBoost Algorithm
Run Decision Tree Algorithm
Run SVM Algorithm
Run Random Forest Algorithm
Run Deep Network Algorithm
Comparison Graph
Type here to search

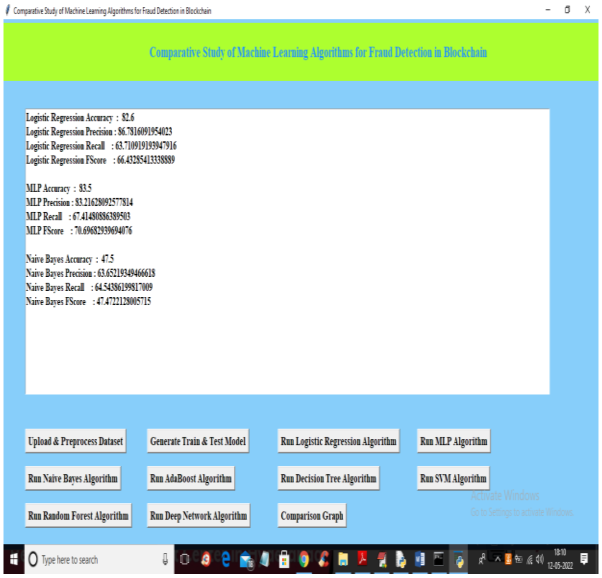
```
[0.0000000e+00 0.0000000e+00 0.0000000e+00 ... 0.0000000e+00
 0.0000000e+00 0.0000000e+00]
[0.0000000e+00 2.82291033e-01 9.59328551e-01 ... 0.0000000e+00
 0.0000000e+00 1.50912377e-05]
...
[4.28526014e-05 1.41379356e-01 9.89955438e-01 ... 0.0000000e+00
 0.0000000e+00 0.0000000e+00]
[2.53423312e-01 0.0000000e+00 7.60245964e-01 ... 0.0000000e+00
 0.0000000e+00 0.0000000e+00]
[0.0000000e+00 7.14254929e-04 9.99969179e-01 ... 0.0000000e+00
 0.0000000e+00 5.06312419e-06]]

Total records found in dataset : 5000
Total features found in dataset: 45

Dataset Train and Test Split

80% dataset records used to train ML algorithms : 4000
20% dataset records used to train ML algorithms : 1000
```

In above screen we can see all data converted to numeric format and we can see total records found in dataset with total columns and then split dataset into train and test and now train and test data is ready and now click on each button to run all algorithms and get below output

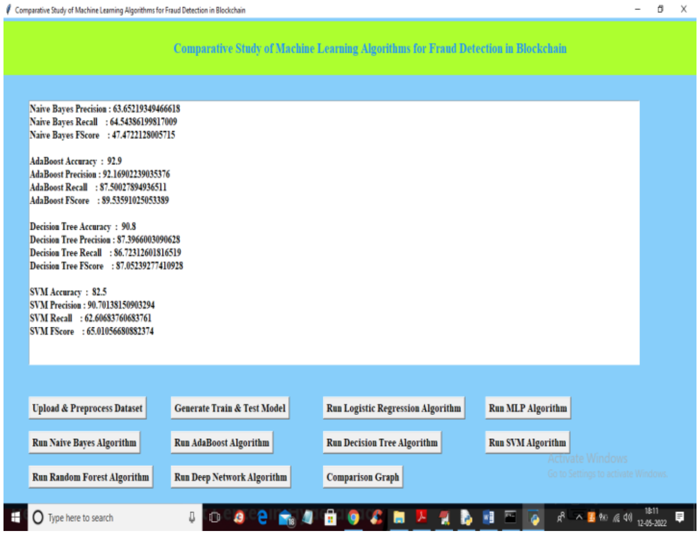

Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Upload & Preprocess Dataset
Generate Train & Test Model
Run Logistic Regression Algorithm
Run MLP Algorithm
Run Naive Bayes Algorithm
Run AdaBoost Algorithm
Run Decision Tree Algorithm
Run SVM Algorithm
Run Random Forest Algorithm
Run Deep Network Algorithm
Comparison Graph
Type here to search

```
Logistic Regression Accuracy  :  82.6
Logistic Regression Precision : 86.7816091954023
Logistic Regression Recall    : 63.710919193947916
Logistic Regression FScore    : 66.43285413338889

MLP Accuracy  :  83.5
MLP Precision : 83.21628092577814
MLP Recall    : 67.41480886389503
MLP FScore    : 70.69682939694076

Naive Bayes Accuracy  :  47.5
Naive Bayes Precision : 63.65219349466618
Naive Bayes Recall    : 64.54386199817009
Naive Bayes FScore    : 47.4722128005715
```

In above screen we can see the performance or accuracy of each algorithm and below is the remaining algorithm accuracy


Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Upload & Preprocess Dataset
Generate Train & Test Model
Run Logistic Regression Algorithm
Run MLP Algorithm
Run Naive Bayes Algorithm
Run AdaBoost Algorithm
Run Decision Tree Algorithm
Run SVM Algorithm
Run Random Forest Algorithm
Run Deep Network Algorithm
Comparison Graph
Type here to search

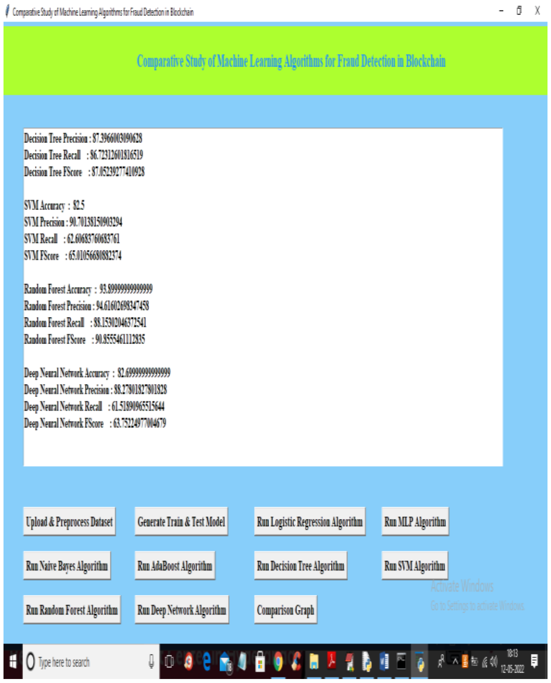
```
Naive Bayes Precision : 63.65219349466618
Naive Bayes Recall    : 64.54386199817009
Naive Bayes FScore    : 47.4722128005715

AdaBoost Accuracy  :  92.9
AdaBoost Precision : 92.16902239035376
AdaBoost Recall    : 87.50027894936511
AdaBoost FScore    : 89.53591025053389

Decision Tree Accuracy  :  90.8
Decision Tree Precision : 87.3966003090628
Decision Tree Recall    : 86.72312601816519
Decision Tree FScore    : 87.05239277410928

SVM Accuracy  :  82.5
SVM Precision : 90.70138150903294
SVM Recall    : 62.60683760683761
SVM FScore    : 65.01056680882374
```

In above screen we can see accuracy of AdaBoost, Decision Tree and SVM and below is the accuracy of remaining algorithms


Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Comparative Study of Machine Learning Algorithms for Fraud Detection in Blockchain
Upload & Preprocess Dataset
Generate Train & Test Model
Run Logistic Regression Algorithm
Run MLP Algorithm
Run Naive Bayes Algorithm
Run AdaBoost Algorithm
Run Decision Tree Algorithm
Run SVM Algorithm
Run Random Forest Algorithm
Run Deep Network Algorithm
Comparison Graph
Type here to search

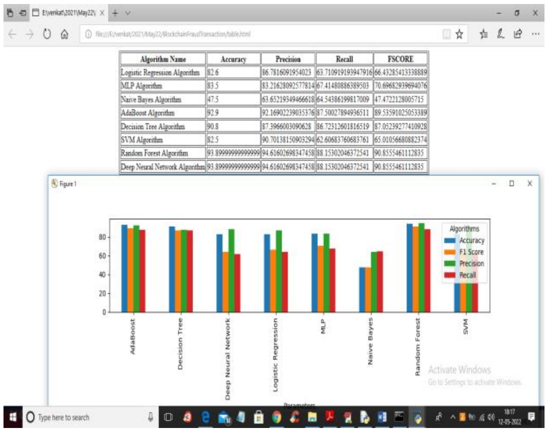
```
Decision Tree Precision : 87.3966003090628
Decision Tree Recall    : 86.72312601816519
Decision Tree FScore    : 87.05239277410928

SVM Accuracy  :  82.5
SVM Precision : 90.70138150903294
SVM Recall    : 62.60683760683761
SVM FScore    : 65.01056680882374

Random Forest Accuracy  :  93.89999999999999
Random Forest Precision : 94.61602698347458
Random Forest Recall    : 88.15302046372541
Random Forest FScore    : 90.8555461112835

Deep Neural Network Accuracy  :  82.69999999999999
Deep Neural Network Precision : 88.27801827801828
Deep Neural Network Recall    : 61.51890965515644
Deep Neural Network FScore    : 63.75224977004679
```

In above screen we can see random forest and Deep neural accuracy and in all algorithms Random Forest is giving better accuracy. Now click on 'Comparison Graph' button to get below output



Algorithm Name	Accuracy	Precision	Recall	FSCORE
Logistic Regression Algorithm	82.6	86.7816091954023	63.710919193947916	66.43285413338889
MLP Algorithm	83.5	83.21628092577814	67.41480886389503	70.69682939694076
Naive Bayes Algorithm	47.5	63.65219349466618	64.54386199817009	47.4722128005715
AdaBoost Algorithm	92.9	92.16902239035376	87.50027894936511	89.53591025053389
Decision Tree Algorithm	90.8	87.3966003090628	86.72312601816519	87.05239277410928
SVM Algorithm	82.5	90.70138150903294	62.60683760683761	65.01056680882374
Random Forest Algorithm	93.89999999999999	94.61602698347458	88.15302046372541	90.8555461112835
Deep Neural Network Algorithm	93.89999999999999	94.61602698347458	88.15302046372541	90.8555461112835

In above screen we can see the accuracy, precision, recall and FSCORE of each algorithm in graph and tabular format and in all algorithms Random Forest giving better result

CONCLUSION :

A method has been proposed for the detection of fraudulent transactions in a blockchain network using machine learning. In this method, various supervised learning approaches like support vector machines, decision trees, logistic regression, and dense neural networks were analyzed. A thorough comparative analysis of all the approaches is performed through accuracy. This work can be extended for the comparative study of unsupervised algorithms like clustering. In the future, we also plan to do an exhaustive study on fraudulent activities in a private blockchain.

REFERENCES :

[1] Cai, Y., Zhu, D. Fraud detections for online businesses: a perspective from blockchain technology. Financ Innov 2, 20 (2016). https://doi.org/10.1186/s40854-016-0039-4

[2] Hyvarinen, H., Risius, M. & Friis, G. A Blockchain-Based Approach ¨ Towards Overcoming Financial Fraud in Public Sector Services. Bus Inf Syst Eng 59, 441–456 (2017). https://doi.org/10.1007/s12599-017-0502- 4

[3] Xu, J.J. Are blockchains immune to all malicious attacks?. Finance Innov 2, 25 (2016). https://doi.org/10.1186/s40854-016-0046-5

[4] Ostapowicz M., Zbikowski K. (2019) Detecting Fraudulent Accounts on ˙ Blockchain: A Supervised Approach. In: Cheng R., Mamoulis N., Sun Y., Huang X. (eds) Web Information Systems Engineering – WISE 2019. WISE 2020. Lecture Notes in Computer Science, vol 11881. Springer, Cham. https://doi.org/10.1007/978-3-030-34223-4 2

[5] Podgorelec, B., Turkanovic, M. and Karakati ´ c, S., 2020. A Machine ˇ Learning-Based Method for Automated Blockchain Transaction Signing Including Personalized Anomaly Detection. Sensors, 20(1), p.147.

[6] Farrugia S, Ellul J, Azzopardi G. Detection of illicit accounts over the Ethereum blockchain. Expert Systems with Applications. 2020 Jul 15;150:113318.

[7] Pham, Thai, and Steven Lee. ”Anomaly detection in bitcoin network using unsupervised learning methods.” arXiv preprint arXiv:1611.03941 (2016).

[8] Monamo, Patrick, Vukosi Marivate, and Bheki Twala. ”Unsupervised learning for robust Bitcoin fraud detection.” 2016 Information Security for South Africa (ISSA). IEEE, 2016.

[9] Shi, Fa-Bin, et al. ”Anomaly detection in Bitcoin market via price return analysis.” PloS one 14.6 (2019): e0218341.

[10] Li, Ji, et al. ”A Survey on Blockchain Anomaly Detection Using Data Mining Techniques.” International Conference on Blockchain and Trustworthy Systems. Springer, Singapore, 2019.

[11] P. N. Sureshbhai, P. Bhattacharya and S. Tanwar, ”KaRuNa: A Blockchain-Based Sentiment Analysis Framework for Fraud Cryptocurrency Schemes,” 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 2020, pp. 1-6, doi: 10.1109/ICCWorkshops49005.2020.9145151.

[12] Brenig, Christian, and Gunter M ¨ uller. ”Economic analysis of cryptocur- ¨ rency backed money laundering.” (2015).

[13] Lorenz, Joana, et al. ”Machine learning methods to detect money laundering in the Bitcoin blockchain in the presence of label scarcity.” arXiv preprint arXiv:2005.14635 (2020).

[14] Bartoletti, Massimo, Barbara Pes, and Sergio Serusi. ”Data mining for detecting Bitcoin Ponzi schemes.” 2018 Crypto Valley Conference on Blockchain Technology (CVCBT). IEEE, 2018.