International Journal of Engineering and Management Research
e-ISSN: 2250-0758 | p-ISSN: 2394-6962
https://ijemr.vandanapublications.com

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

Online Fraud Detection Using Machine Learning Approach

Viswanatha V[1], Ramachandra A.C[2], Deeksha V[3] and Ranjitha R[4]
[1]Assistant Professor, Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology, Bangalore, INDIA
[2]Professor, Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology, Bangalore, INDIA
[3]Student, Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology, Bangalore, INDIA
[4]Student, Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology, Bangalore, INDIA

[1]Corresponding Author: viswas779@gmail.com

Received: 04-07-2023 Revised: 20-07-2023 Accepted: 04-08-2023

ABSTRACT

Online extortion discovery has ended up a tremendous issue in today's advanced age and poses a danger to individuals, businesses, and budgetary teachers all over the world. The increment in extortion illustrates the require for compelling extortion discovery, especially within the setting of anti-money laundering (AML) endeavors. This extent is planned to create a machine learning based arrangement utilizing Python to distinguish and avoid online extortion in genuine time.

The proposed framework employment chronicled exchange information, combining different components such as client behavior, exchanges, and budgetary information. First, the information control prepare is utilized to clean the information and change over it into organize reasonable for the preparing show. At that point, different machine learning calculations such as calculated relapse, choice trees, irregular timberlands or angle boosting are used to build predictive algorithms that can spot fraud. The extended concludes with the usage of the created show in a genuine world online exchange environment, permitting for genuine time extortion location and avoidance. The system's adequacy is persistently checked and assessed, and essential overhauls and advancements are made to adjust to advancing extortion designs and procedures. By and large, this extends points to supply a strong and proficient arrangement utilizing Python and machine learning strategies to combat online extortion. By precisely recognizing false exchanges in genuine time, this framework can altogether contribute to fortifying AML endeavors and ensuring people and organizations from money related misfortunes and reputational harm related with online extortion.

Keywords-- Unique Information Mining, Online Fraud Detection, Machine Learning, Decision Tree Algorithm

I. INTRODUCTION

Online extortion has become a predominant issue in today's advanced age. With the expanding dependence on innovation and the web for different exchanges, hoodlums have found modern and advanced ways to misdirect and dupe clueless people and businesses. The result is required for successful extortion discovery frameworks has ended up more vital than ever some time recently. One such structure has risen as effective apparatus in combating online extortion is online extortion detection[1]-[2]

Online extortion discovery alludes to the utilize of progressed calculations and machine learning procedures to distinguish and anticipate false exercises happening over the web. It includes analyzing huge volumes of information in real-time to identify designs, inconsistencies, and suspicious behavior that will demonstrate false action. By leveraging fake insights and information analytics, online extortion location frameworks can rapidly distinguish potential fraudsters and take fitting activities to relieve the dangers related with online extortion [3]-[4].

The presentation of online extortion locations has revolutionized the way organizations approach extortion avoidance. Conventional strategies of extortion discovery, such as manual audits and rule-based frameworks, were regularly time-consuming, incapable, and inclined to human mistake. These strategies depended intensely on human instinct and were incapable to keep up with the quickly advancing strategies utilized by fraudsters [[4]-[5]
In differentiate, online extortion discovery frameworks are competent of handling endless sums of information in real-time, permitting for prompt distinguishing proof and reaction to false exercises. These frameworks can analyze different information focuses at the same time, counting

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

client behavior, exchange history, gadget data, and relocation information, to construct a comprehensive profile of each client. By comparing this profile against known designs of false behavior, these frameworks can precisely distinguish and hail suspicious exercises [6]-[7].
Because of the emergence of cutting-edge innovation, the mode of payment of a person has changed altogether. The utilize of Online Installment mode such as Online Managing an account, Charge Card, Credit Card etc. has ended up well known and is getting to be important in day-to-day exercises since it permits bank customers to buy products and administrations from the shopping websites.

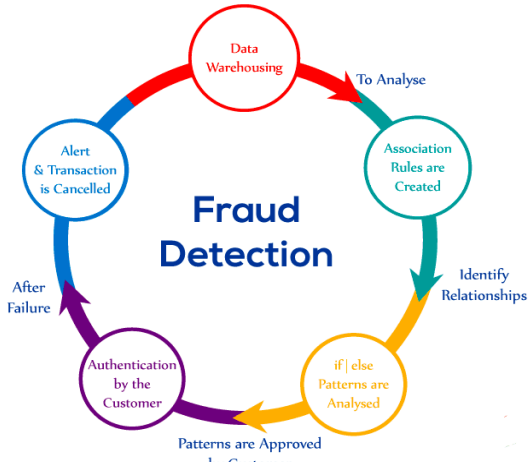

Figure.1: Online fraud detection

Figure 1 shows the process involved in online fraud detection and some techniques used in online fraud detection include machine learning, behavioral analysis, and data mining.

Extortion bargains with cases that happen due to criminal reasons which are troublesome to recognize. Extortion can be basically partitioned into two types:

 Offline Extortion: Most of the offline extortion episodes happen due to the taking of purse/wallet that contains critical documents. Archives such as Driving Permit, ID card etc. contain pivotal data such as title, date of birth, exchange slips etc.

 Online Extortion: Online extortion happens when fraudster present their site as a veritable site in arrange to obtain pivotal individual information of a client and perform illegal exchanges on such client account.

Online extortion discovery employing a decision tree classifier includes employing a choice tree calculation to classify whether a transaction or movement is false or not. The method ordinarily includes the taking after steps:

1. Information Collection: Collect significant information related to online exchanges or exercises, counting highlights such as exchange sum, IP address, area, gadget data, client behavior designs, etc.

2. Information Preprocessing: Clean and preprocess the collected information by dealing with lost values, outliers, and changing categorical factors into numerical representations.

3. Include Determination: Select the foremost pertinent highlights that are likely to contribute to extortion location. This step makes a difference in decreasing the dimensionality of the information and making strides the model's performance.

4. Part the Information: Part the preprocessed information into preparing and testing sets. The training set is utilized to prepare the choice tree classifier, whereas the testing set is utilized to assess its performance.

5. Choice Tree Training: Train the choice tree classifier on the preparing information. The calculation will learn designs and rules from the information to form expectations approximately extortion.

6. Show Assessment: Assess the execution of the prepared choice tree classifier utilizing different assessment measurements such as precision, accuracy, review, and F1-score. This step makes a difference in evaluating the adequacy of the demonstration in identifying online fraud.

7. Hyper parameter Tuning: Optimize the choice tree classifier by tuning its hyper parameters to progress its execution encouragement. Common hyper parameters incorporate greatest profundity, least tests for a part, and measure for splitting.

8. Foreseeing Extortion: Utilize the prepared choice tree classifier to foresee whether a modern exchange or action is false or not based on its features.

9. Checking and Overhauling: Ceaselessly screen the model's execution and overhaul it with modern information to adjust to changing extortion designs and move forward exactness.

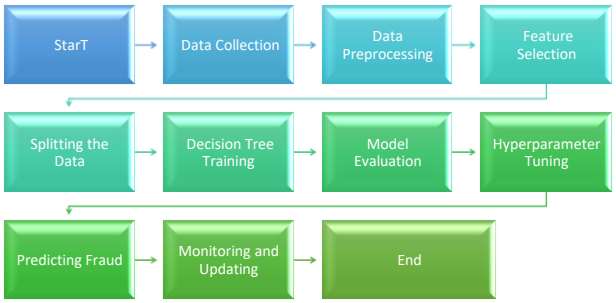

Figure.2: Flowchart of online fraud detection

Figure 2 represents the flowchart of online fraud detection from information collection and preprocessing to checking and overhauling.

Strategies to Take Individual Information

There are different strategies or procedures that cyber hoodlums used to commit the crime:

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

- Hacking: Programmer could be a individual who looks for and misuses weakness in computer framework. Aggressors break into industry or individual databases.
- Phishing: Phishing could be a false endeavor, not many through e-mail, to take your individual information.
- Spoofing: The word "parody" implies hoax, trap, or deceive. Spoofing alludes to deceiving or deluding the computer clients. Usually ordinarily done by covering up one's identity or faking the personality of another client on the Internet.
- Spyware: The computer client unconsciously downloads software from the Web that contains spyware. Spyware collects individual data from your computer and transmits it to fraudsters or attackers.
- Bear Surfing: An assailant observes a bank client from a adjacent area as the client punches in his personal data. In the event that the client is giving his personal data over the phone (e.g., to a inn or car rental company), the assailant may tune in to the conversation so as to get individual data of bank customer
- Dumpster Jumping: An aggressor goes through a customer's trash cans or waste containers to get individual information of bank client such as bank articulation, payment receipt etc.

II. LITERATURE REVIEW

The outbreak of COVID-19 burgeons newborn services on online platforms and simultaneously buoys multifarious online fraud activities. Due to the rapid technological and commercial innovation that opens an ever-expanding set of products, insufficient labeling data renders existing supervised or semi-supervised fraud detection models ineffective in these emerging services. However, the ever-accumulated user behavioral data on online platforms might be helpful in improving Investigating fraud in children's services. For this purpose, in this article, we first present the user behavior, which consists of orderly arranged actions, from the large-scale unlabeled data sources for online fraud detection. Recent studies illustrate that accurate extraction of user intentions (formed by consecutive actions) in behavioral sequences can propel improvements in the performance of online fraud detection. analyze the nature of online fraud; We are developing a model called UB-PTM that investigates fraudulent knowledge. Actions on three institutional tasks with varying degrees of detail, i.e., actions, The level of intent and consistency of large untagged data. Extensive experimentation with three downstream tasks to detect online fraud at the transaction and user level has resulted in our UB-PTM It can outperform modern designs for certain tasks [8]-[9]

Increased use of computer innovations and continued business development Exchanging money through e-commerce systems such as credit card systems, multimedia transmission systems, health protection systems, etc. Unfortunately, these systems are used by both real customers and fraudsters. Scammers also use different approaches to hack e-commerce platforms. Fraud Prevention System (FPS) cannot guarantee: Legitimate security of e-commerce systems. In any case, the interaction between FPS and FDS can help secure e-commerce systems. All things considered, there are issues and challenges that prevent the execution of FDSs, such as concept float, underpins genuine time location, skewed conveyance, expansive sum of information etc. This study paper points to supply a precise and comprehensive outline of these issues and challenges that deter the execution of FDSs. We have chosen five electronic commerce frameworks, which are credit card, media transmission, healthcare protections, car protections and online sell off. The predominant extortion sorts in those E-commerce frameworks are presented closely. Encourage, state-of-the-art FDSs approaches in chosen E-commerce frameworks are efficiently presented. At that point a brief talk on potential investigate patterns within the close future and conclusion are displayed.[10]

Progressively, customers depend on social data channels, such as user-posted online audits, to form buy choices. These audits are accepted to be fair-minded reflections of other consumers' encounters with the items or administrations. Whereas broadly accepted, the writing has not tried the presence or non-existence of survey control. By utilizing information from Amazon and Barnes & Respectable, our ponder examines if merchants, distributors, and scholars reliably control online customer surveys. We record the presence of online survey control and appear that the control technique of firms appears to be a monotonically diminishing work of the product's genuine quality or the cruel buyer rating of that item. Control in this way diminishes the instructive substance of online surveys. Shoppers moreover get it the presence of control but can as it were mostly adjusted it based on their desires of the general level of control. Hence, merchants can manipulate online analysts to alter the ultimate comes about. It moreover appears that both price and surveys serve as pointers of quality within the early stages of a product's launch on the Amazon commercial center. So, at this arrangement, higher costs increment deals instead of diminishing them. In the afterward stages, costs play a ordinary part. In other words, higher costs lead to lower deals. At long last, Barnes & Noble's normal control level is higher than Amazon's [11]-[12].

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

Nowadays illicit exercises with respect to online money related exchanges have ended up progressively complex and borderless, coming about in colossal budgetary misfortunes for both sides, clients, and organizations. Numerous strategies have been proposed to extortion anticipation and discovery within the online environment. Be that as it may, all these strategies other than having the same objective of recognizing and combating false online exchanges, they come with their possess characteristics, focal points and drawbacks. In this setting, this paper reviews the existing inquiry about drained extortion discovery with the point of recognizing calculations utilized and analyzing each of these calculations based on certain criteria. To analyze the investigation ponders within the field of extortion discovery, the efficient quantitative writing audit strategy was connected. A progressive typology is built based on the machine learning calculations most frequently mentioned in logical papers and their characteristics. did. Therefore, our article highlights the most excellent strategies for discovery in a other way. It anticipates extortion by combining three determination criteria: financial soundness, scope, and taken a toll [13].

Utilizing remote versatile terminals has gotten to be the standard of Web exchanges, which can confirm the character of clients by passwords, fingerprints, sounds, and pictures. In any case, once this personality information is stolen, conventional data security strategies will not dodge online exchange extortion. The existing convolutional neural organize show for extortion location should create numerous subordinate highlights. This paper proposes a extortion discovery demonstrate based on the convolutional neural arrange within the field of online exchanges, which develops an input highlight sequencing layer that actualizes the reorganization of crude exchange highlights to create diverse convolutional designs. Its centrality is that distinctive combinations entering the convolution bit will create diverse subordinate highlights. The advantage of this model is that it takes low-dimensional and non-derivative online value-based information as input. The whole arrangement comprises of highlight arrangement layers, four convolutional and pooling layers, and completely associated layers. Test comes about approved with commercial bank online exchange information appear that the demonstrate gives great extortion location execution without subordinate capacities. In expansion, the exactness can be stabilized at approximately 91% and the review rate at almost 94%, an increment of 26% and 2%, separately, compared to the existing negative location CNN [14].

Online trading is a common user behavior with incredible growth rates today. Completely hassle-free and becomes a habit of the user. The most common example can be seen on various e-commerce platforms where more and more users increase the number of transactions every day. With such a large online trading network, it is expected that there will be certain loopholes that make such platforms fraudulent and conflicting. Therefore, in this project, we will be discussing a fraud detection and verification system designed primarily for online transactions for e-commerce. This document prescribes certain methods that can be used to authenticate online transactions and follows verification mechanisms that, to some extent, help verify the authenticity of transactions if they turn out to be fraudulent. The aspects described in this white paper can be looked at in more detail to build a robust fraud detection system, but they certainly form the basis for building a viable fraud detection system [15].

Online audit extortion has advanced in advancement by propelling shrewd campaigns where a bunch of facilitated members work together to provide misleading surveys for the assigned targets. Such collusive extortion is considered much harder to guard against as these campaign members are able of sidestepping discovery by forming their behaviors collectively so as not to seem suspicious. The display work complements existing things about by investigating more inconspicuous behavioral trails associated with collusive audit extortion. A novel factual demonstration is proposed to assist characterize, recognize, and estimate collusive extortion in online audits. The proposed demonstration is totally unsupervised, which bypasses the trouble of manual comment required for supervised modeling. It is additionally exceptionally adaptable to incorporate collusive properties that can be utilized for superior modeling and forecasting. Tests with two genuine information sets appear the adequacy of the proposed strategy and the change of learning and prescient abilities [15]

Fraud in current online advertising activities increases the risk to online marketing; Advertising industry and e-business. Click fraud is considered one of the most important problems on the internet. Advertising. Despite ongoing efforts by online advertisers to improve traffic filtering, we are still looking for the best protection to detect fraudulent clicks. Therefore, effective Online advertising requires fraud detection algorithms. The purpose of our article is Determine the accuracy of one of the latest machine learning algorithms for detecting click fraud. Online environment. In this study, we studied the click patterns of a data set of 200 million processing. Click in 4 days. The main goal is to evaluate the user's click path on the portfolio and Flags IP addresses that generate a lot of clicks but don't lead to app installs. As a methodology we LightGBM - using experimental tests for the Gradient Boosted Decision Tree Method. this algorithm Provides 98% accuracy. In our study, the literature review was the central source for testing us result [16]

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

While the number of online auctions continues to grow, the number of online auction scams is also increasing. To avoid detection, scammers often disguise normal trading behavior by disguising themselves as honest participants. So staying vigilant isn't enough to prevent scams. Online auction participants need a more proactive approach to protecting their interests, such as an early fraud detection system. In practice, both accuracy and timeliness are equally important in developing an effective detection system. Immediate but misleading messages to the user are not acceptable. However, a lengthy discovery process does not help traders place orders on time. Detection results would be more useful if potential scammers could be reported as soon as possible. This study proposes a new fraud early detection method that considers accuracy and timeliness at the same time. A modified wrapper procedure was developed to select a subset of properties from a large pool of candidate properties to determine the most appropriate properties to discriminate between regular traders and cheaters. We then use these properties to propose an additional step-by-step modeling procedure to extract features from the last part of a trader's trade history, reducing the time and resources required for modeling and data collection. You can get early fraud detection models by building decision trees or learning from instances. Our experimental results show that the augmented hybrid model significantly improves the fraud detection accuracy, while the performance of the selected attributes is higher than that of the other attribute sets [17].

Internet banking and e-commerce have experienced explosive growth over the past few years and promise tremendous growth in the future. This has made it easier for scammers to use clever new ways to commit credit card fraud over the internet. This paper focuses on real-time fraud detection and presents a new and innovative approach to understanding spending patterns to decipher potential fraud cases. Decode, filter, and analyze customer behavior using self-organizing maps to detect fraud [18].

This study research attempts to prohibit privacy and loss of money for individuals and organizations by creating a reliable model which can detect the fraud exposure in the online recruitment environments. This research presents a major contribution represented in a reliable detection model using ensemble approach based on Random Forest classifier to detect Online Recruitment Fraud (ORF). The detection of Online Recruitment Fraud is characterized by other types of electronic fraud detection by its modern and the scarcity of studies on this concept. The researcher proposed a detection model to achieve the objectives of this study. For feature selection, the support vector machine method is used and for classification and detection, ensemble classifier using Random Forest is employed. This model is applied using a free dataset called EMSCAD (Employment Scam Aegean Dataset). A preprocessing step was applied before selection and sorting were accepted. The result showed an accuracy of 97.41%. The results also show key features and important factors for search purposes, including the presence of company profile features, the presence of company logos and industry specificities.

E-commerce includes online shopping, banking, financial institutions, and governments. End Fraud in many areas of business and everyday life. These activities Telecom, most common in credit card fraud detection, network intrusion, finance and insurance and Scientific applications lose billions of dollars everyday increase in credit card transactions Use both online and offline at the same time. to be strong and develop an effective fraud detection algorithm is the key to reduction. trade loss. There have been numerous approaches Implemented for fraud detection. this article shows an approach used for fraud detection in e-commerce.

With the enhancement in technology e-banking like credit Card, Debit Card, Mobile Banking, and Internet Banking is the popular medium to transfer the money from one account to another. E-Banking is gaining popularity day by day, which increases the online transaction with the increase in online shopping, online bill payment like electricity, Insurance Premium and other charges, online recharges and online reservation of railways, bus etc., so the fraud cases related to this are also increasing and it puts a great burden on the economy, affecting both customers and financial bodies. It not only costs money, but also a great amount of time to restore the harm done. The purpose is to prevent the customer from online transaction by using specific techniques i.e. based on Data Mining and Artificial Intelligence technique. The risk score is calculated by Bayesian Learning Approach to analyze whether the transaction is genuine or fraudulent based on the two parameters: Customer Spending Behavior and Geographical Locations. customers over spending Behavior you can see using KMEAN clustering Algorithm and Geolocation Current Your geographic location is compared to your previous location. A trade is considered if the risk indicator is greater than 0.5. Fraudulent Transactions and Security Mechanisms Enter a 4-digit random number to authenticate the user. appear on the screen and input by the real user Your code is in the correct order.

III. METHODOLOGY

The methodologies include the algorithm used, dataset used and flowchart of the data used and implemented. Below is the provided step by step explanation of the algorithm used.

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

Algorithm Used: The decision tree algorithm is a widely used supervised learning technique employed for both classification and regression tasks. It constructs a structured model resembling a flowchart, driven by input features.

Tree Construction: The algorithm commences by considering the entire dataset as the root node, and selects the optimal feature for partitioning the data.

Feature Split: The chosen feature is utilized to divide the data into subsets, thereby creating branches or paths within the decision tree.

Recursive Splitting: The process of feature splitting is iteratively applied to each subset until a predefined stopping criterion is satisfied.

Leaf Node Assignment: Leaf nodes are assigned class labels or regression values based on the majority class or mean value of the target variable within each respective subset.

Prediction: To make predictions, the algorithm traverses the decision tree by evaluating feature values and ultimately reaching a leaf node to obtain the final prediction.

Easy to comprehend and interpret accommodates numerical and categorical data handles missing values gracefully captures non-linear relationships effectively.

Prone to over fitting, necessitating proper regularization techniques - Can be sensitive to changes in the dataset, leading to instability. Exhibits bias towards features with high cardinality or many levels In conclusion, decision trees offer versatility and transparency in model interpretation. However, caution must be exercised to address overfitting issues and effectively manage the algorithm's limitations.

Import necessary libraries and module: from sklearn. tree import Decision tree classifier, import pandas as pd and few other libraries are imported.

Read the dataset: use necessary commands to read the dataset provided and collect the information from it.

Preprocess the data: here the data collected is pre-processed so that feature extraction can take place easily.

Load the data: The data pre-processed now will be loaded into the training model to train the model properly.

Splitting the data: The data is splitted into training and testing data in proportion of 80:20 out of 100.

It checks for the target: if present then goes to next step or else declares it as unsupervised model as it does not have specified output.

Checks for target data: we have used the decision tree classifier therefore uses discrete data.

Training model: after all the necessary adjustments in the code we train the model using decision tree algorithm.

Testing data: after training the model is tested and accuracy is obtained with necessary graphs. A small comparison plot is also included in algorithm.

Results: as last step the results are obtained, and the model execution ends. Few additional steps of regularization imputation and many other commands are included in the code to improve the accuracy and prevent the model from overfitting and underfitting. This also increases the generalization of the model and allows it to predict the new data with more accuracy.

Decision Trees are supervised learning methods that can be used to solve classification and regression issues, however they are typically chosen for Classification issues. It is a tree-structured classifier, with internal nodes denoting dataset features, branches denoting decision rules, and each leaf node denoting the classification result. There are two nodes—the Decision Node and the Leaf Node—in a decision tree. Decision nodes are used to make decisions and have many branches, whereas Leaf nodes represent the results of those decisions and do not have any additional branches. Based on the characteristics of the available dataset, judgments or tests are run.

Here is a brief explanation of the flowchart of model training in figure 3.

1. Start: The flowchart begins with the start symbol, indicating the beginning of the decision tree algorithm.

2. Load Dataset: The algorithm loads the dataset, which contains the input features and target variable.

3. Define Features and Target: The feature columns and target column are defined, specifying the variables to be used for training the decision tree.

4. Split Data: The dataset is split into training and testing sets using the train_test_split function, allocating a portion of the data for model evaluation.

5. Data Imputation: The SimpleImputer object is used to handle missing values in the dataset, replacing them with the mean value of the respective feature.

6. Build Decision Tree: The DecisionTreeClassifier object is created, representing the decision tree model. It is trained on the training data using the fit function.

7. Predictions: The trained decision tree is utilized to make predictions on the test set, using the predict function.

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

IV. FLOWCHART

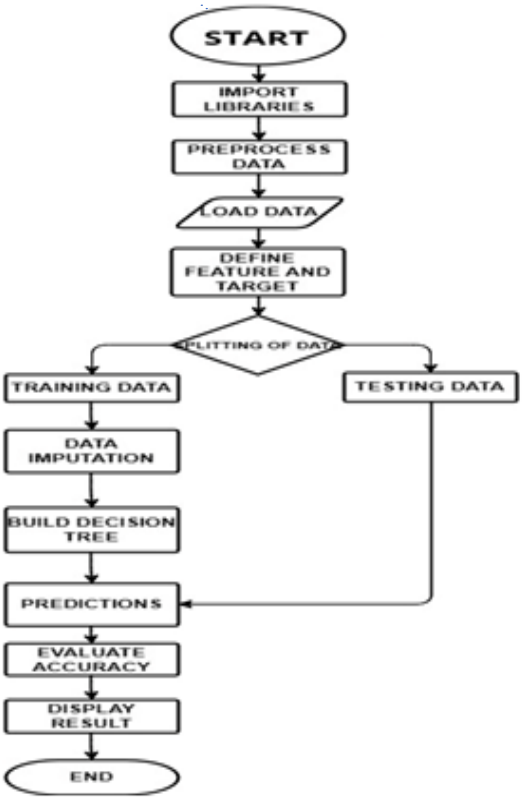

Figure 3: Flowchart of model training

8. Evaluate Accuracy: The accuracy of the model is calculated by comparing the predicted values with the actual target values using the accuracy score function.
9. Display Results: The accuracy score is printed to the console, providing an assessment of the model's performance.
10. End: The flowchart concludes with the end symbol, indicating the completion of the decision tree algorithm.

The flowchart shown in figure 3 provides a visual representation of the steps involved in training and evaluating the decision tree model, aiding in understanding the overall process and facilitating communication between different stakeholders.

Datasets: The dataset currently used in the model training is taken from the Kaggle website. Kaggle is a popular platform for data scientists and machine learning practitioners to discover and share datasets, as well as participate in data science competitions. It hosts a vast collection of datasets from various domains, allowing users to access and analyse real-world data. Kaggle datasets are typically provided in structured formats such as CSV, Excel, or SQL, and may contain a wide range of variables or features. These datasets cover diverse topics including healthcare, finance, social sciences, computer vision, natural language processing, and more. Users can explore and download datasets from Kaggle for their own analysis, model training, or research purposes. They can also contribute by uploading and sharing their own datasets with the Kaggle community. Kaggle datasets are accompanied by detailed descriptions, documentation, and often include pre-split train/test datasets to facilitate model development and evaluation. Additionally, many datasets come with sample code notebooks, known as kernels, which provide examples and insights on how to work with the data. By leveraging the vast collection of Kaggle datasets, data scientists can gain access to high-quality, real-world data to explore, analyse, and build predictive models or develop insights for a wide range of applications.

V. RESULTS AND DISCUSSIONS

The machine learning model utilizes python code in Jupyter notebook to train and test the model. The model uses decision tree algorithm. The code starts from training the feature and target columns from dataset. Numpy is a Python library for efficient numerical computations, offering multidimensional arrays and mathematical functions. It is widely used in scientific computing and data analysis. Pandas is a powerful data manipulation library built on top of Numpy. It provides high-level data structures like Data Frames and Series, making data manipulation and analysis easier. Data Frames are two-dimensional tables with labelled rows and columns, while Series are one-dimensional labelled arrays. Pandas offers tools for data cleaning, pre-processing, merging, reshaping, and analysing structured data. It supports flexible indexing, filtering, and grouping operations, allowing easy extraction and manipulation of specific subsets of data. Numpy and Pandas are widely used in scientific computing, data analysis, and machine learning applications. Numpy provides efficient storage and manipulation of large arrays, while Pandas provides intuitive data manipulation capabilities. Both libraries integrate well with other Python data ecosystem tools like Matplotlib and Scikit-learn. Numpy and Pandas are essential for data manipulation, analysis, and preprocessing in Python. Together, they provide efficient and convenient tools for working with arrays and structured data. The first code snippet used in code in summary, the code performs data preprocessing, splits the data into training and test sets, trains two Decision Tree Classifiers with different parameters, and evaluates their accuracy in predicting the target variable. Here a simple imputer is included. The Simple Imputer is a class from the scikit-learn library that provides a simple strategy for handling missing values in a dataset. It is used to replace

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

missing values with a chosen strategy, such as the mean, median, or most frequent value of the respective feature.
In summary, next code snippet utilizes a trained Decision Tree Classifier to generate a visual representation of the decision boundaries in the feature space. The decision boundaries are plotted as filled contour regions, and the training points from the dataset are displayed with colours representing different features. This visualization helps in understanding how the classifier separates and categorizes the data points based on their feature values. Decision boundaries are lines or surfaces that separate different classes in a classification problem. They define the regions where a classifier assigns different labels. They are determined by learning from the training data and finding the optimal separation between classes. Decision boundaries are influenced by the features used for classification and can vary in complexity. Visualizing decision boundaries helps understand how a classifier distinguishes classes and identifies areas of uncertainty. The complexity of decision boundaries reflects the relationships between features. They are crucial for evaluating classifiers and assessing their generalization ability. Decision boundaries aid in model interpretation, evaluation, and decision-making, providing a visual representation of class separation in the feature space it is shown in figure 4.

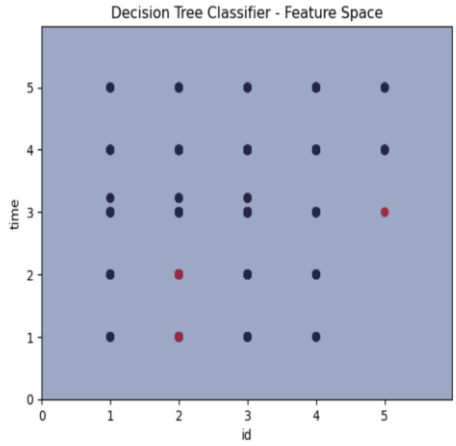

Figure 4: Decision boundaries in feature space [time=black colour, id=red colour]

The distribution of transaction types in the dataset is depicted using a pie chart in the given code. The generated pie chart gives you a general idea of how different transaction types are distributed throughout the dataset and enables you to see how many of each type there are. The size of each pie slice shows the relative frequency or occurrence of a certain transaction type in the dataset, and each slice represents a particular transaction type. The use of logarithmic trends in data analysis and visualization is to better understand and represent exponential or multiplicative relationships between variables**.** By plotting data on a logarithmic scale, it compresses large ranges of values and magnifies smaller changes, making it easier to observe patterns and trends. In the provided code, the logarithmic trends of different features in the dataset are visualized using line plots, allowing for a clearer understanding of the relationships and patterns present in Figure 5.

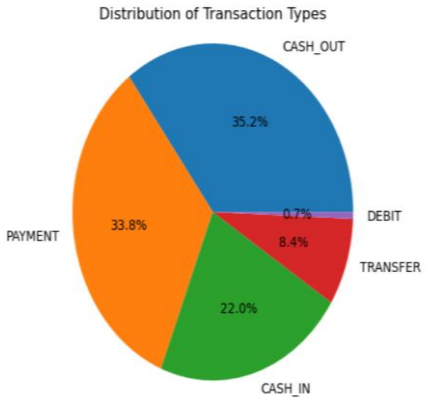

Figure 5: Pie Chart

The code employments the seaborn library (imported as sns) and matplotlib.pyplot (imported as plt) to make and show a histogram. The plt.figure (fig size=(10, 6)) line sets the measure of the figure, guaranteeing the coming about plot has appropriate dimensions.sns.histplot() is utilized to form the histogram. The information parameter is set to the Data Frame df, demonstrating that the 'amount' column will be utilized for the histogram. The x parameter indicates the column to be plotted. The kde=True contention includes a part thickness estimation plot to the histogram.Additional customizations are made utilizing matplotlib. plt.title () sets the title of the plot as 'Histogram of Exchange Amount'. plt.xlabel () and plt.ylabel () are utilized to name the x-axis and y-axis, respectively.Finally, plt.show() is called to show the histogram. The coming about histogram provides a visual representation as in figure 6, of the dissemination of exchange sums within the dataset. The x-axis speaks to the exchange sum, and the y-axis speaks to the recurrence of exchanges falling inside each sum run. The histogram makes a difference to recognize designs, exceptions, and the in general shape of the dispersion, giving bits of knowledge into the exchange sums show within the dataset. Figure 6 represents the expansion of the bit thickness estimation plot (kde=True) gives a smoothed representation of the dissemination, permitting for way better visualization of the basic thickness as.

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

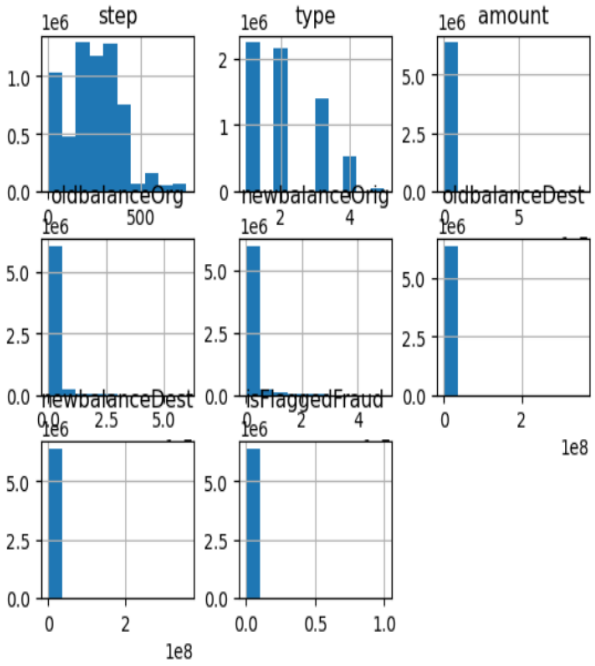

Figure 6: Histogram

The code to begin with calculates the relationship framework by calling df.corr() on the DataFrame df. The relationship lattice speaks to the pairwise relationships between the numeric columns within the dataset.

Next, the code makes a heatmap utilizing seaborn's heatmap() work. The annot=True parameter permits showing the relationship values on the heatmap, and the cmap='coolwarm' parameter sets the color plot for the heatmap.

We set the figure measure utilizing plt.figure(figsize=(10, 8)) to guarantee the heatmap has fitting dimensions.

Additional customizations are made utilizing matplotlib (imported as plt). We set the plot title utilizing plt.title().

Finally, plt.show() is called to show the heatmap.

The coming about heatmap visualizes the correlations between factors within the dataset. Positive relationships are demonstrated by hotter colors (e.g., ruddy), negative relationships by cooler colors (e.g., blue), and no relationship by a neutral color (e.g., white).

By analyzing the heatmap, you'll recognize connections between factors, such as solid positive or negative relationships, which can give experiences into the dataset's designs and conditions. As shown in figure 7.

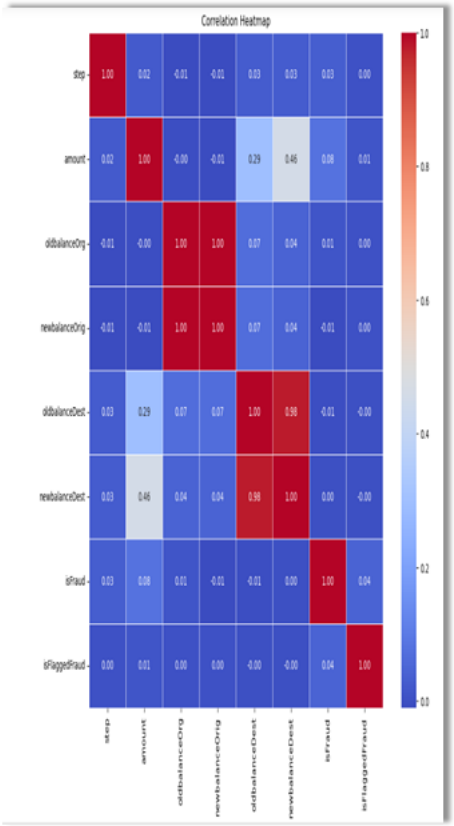

Figure 7: Correlation Heatmap

Begin at the best hub (root hub) of the tree. Each hub speaks to a choice based on a highlight and a edge value.

Based on the highlight and limit esteem at each hub, take after the fitting department. On the off chance that the condition is fulfilled, move to the cleared out child hub; something else, move to the correct child node.

Continue navigating the tree until you reach a leaf node. Leaf hubs speak to the ultimate choice or prediction. The color of the hubs within the chart demonstrates the lesson dispersion. Filled hubs have a larger part of tests having a place to a particular class.

The number in each hub speaks to the lesson dispersion and the check of tests for each class.

By analysing the choice tree chart, you'll get it the choice rules utilized by the calculation and pick up bits of knowledge into how distinctive highlights contribute to the classification process.

Make beyond any doubt to alter the variable names (demonstrate, xtrain, ytrain, x.columns) agreeing to your particular code and information.

In order to understand the behaviour of the features with respect to each other scatter plots are plotted. The code uses the Pandas and Plotly libraries to create a subplot with 6-line plots representing different features of

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

a dataset. The dataset is loaded using Pandas, and a subplot with 6 rows and 1 column is created using Plotly's `make subplots () `.

The output of the code is a countplot that displays the distribution of the 'target' column. Each bar represents the count of occurrences for each unique value in the 'target' column. The countplot provides insights into the class distribution and can be used to analyze the balance or imbalance of classes in the dataset.

The decision tree is shown visually as the code's output. The arrows show the flow from one node to another based on the decisions, and each node in the tree reflects a decision based on a feature. The majority class for each node is represented by the node's color. The decision-making process of the decision tree classifier based on the characteristics and their significance in forecasting is depicted visually in the plot.

In summary, next code loads a dataset using Pandas, separates the features and the target variable, and creates an instance of the DecisionTreeClassifier from scikit-learn. It then fits the classifier on the dataset and plots the decision tree using the tree.plot_tree() function. The resulting plot visualizes the decision tree structure, with each node representing a decision based on a specific feature, and the leaf nodes indicating the predicted classes. The plot is displayed using matplotlib. pyplot. show (). A decision tree plot is a graphical representation that illustrates the decision-making process of a decision tree classifier or regressor. It visually depicts the hierarchical structure of the decision tree, where nodes represent decision points and branches represent possible outcomes. The plot displays the feature names and their corresponding thresholds at each decision node. Terminal nodes, or leaves, signify the final decisions or predicted outcomes. The size and colour of the nodes can be utilized to convey the sample count or class distribution at each node. The plot offers a clear and intuitive representation of the decision tree's logic and decision boundaries. It aids in comprehending how the decision tree makes predictions based on various features and thresholds. Figure 8, The plot can be customized to include class labels, feature importance, and other pertinent information. It is valuable for interpreting and explaining the decision tree model to stakeholders or non-technical audiences. The decision tree plot allows for easy identification of significant features and their influence on the decision-making process. It assists in identifying regions of high predictive accuracy as well as areas where the model may struggle to make accurate predictions. The plot can be generated using various Python libraries, such as scikit-learn, matplotlib, or plotly. It helps identify potential overfitting or underfitting issues by visualizing the complexity of the decision tree. The plot serves as a valuable tool for model validation, comparison, and explanation. Figure 9**.** It facilitates effective communication between data scientists, analysts, and stakeholders by providing a visual representation of the decision tree's logic. The decision tree plot is a powerful visualization tool that enhances the interpretability and transparency of decision tree models. Plot has a max depth of 2 as shown in Max depth is command used to reduce the complexity of model and improve model accuracy. Figure 10.

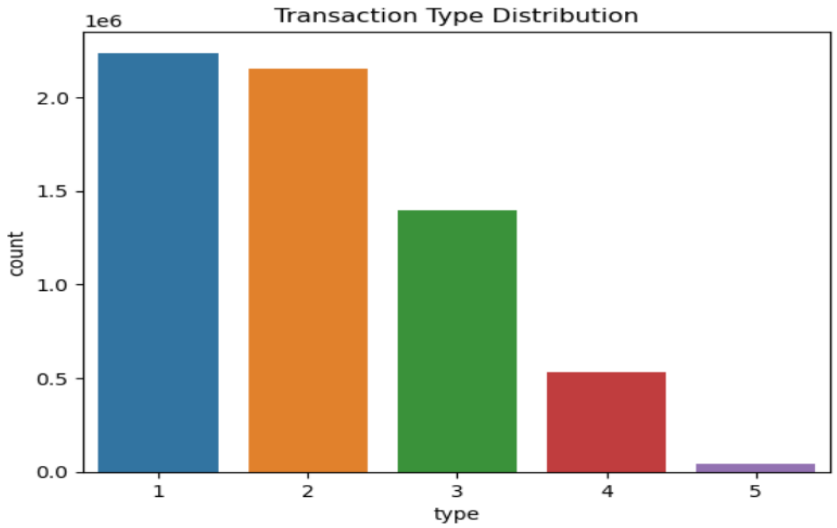

Figure 8: Transaction Distribution Graph

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

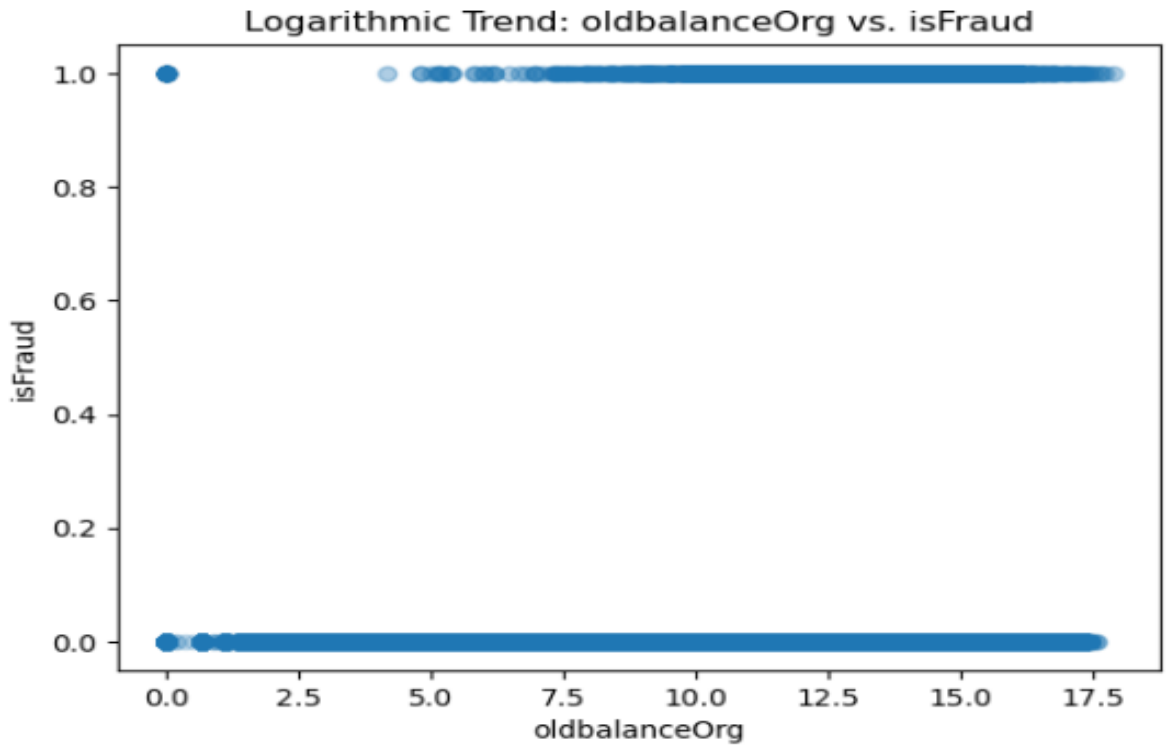

Figure 9: Logarithmic trend, oldbalanceOrg

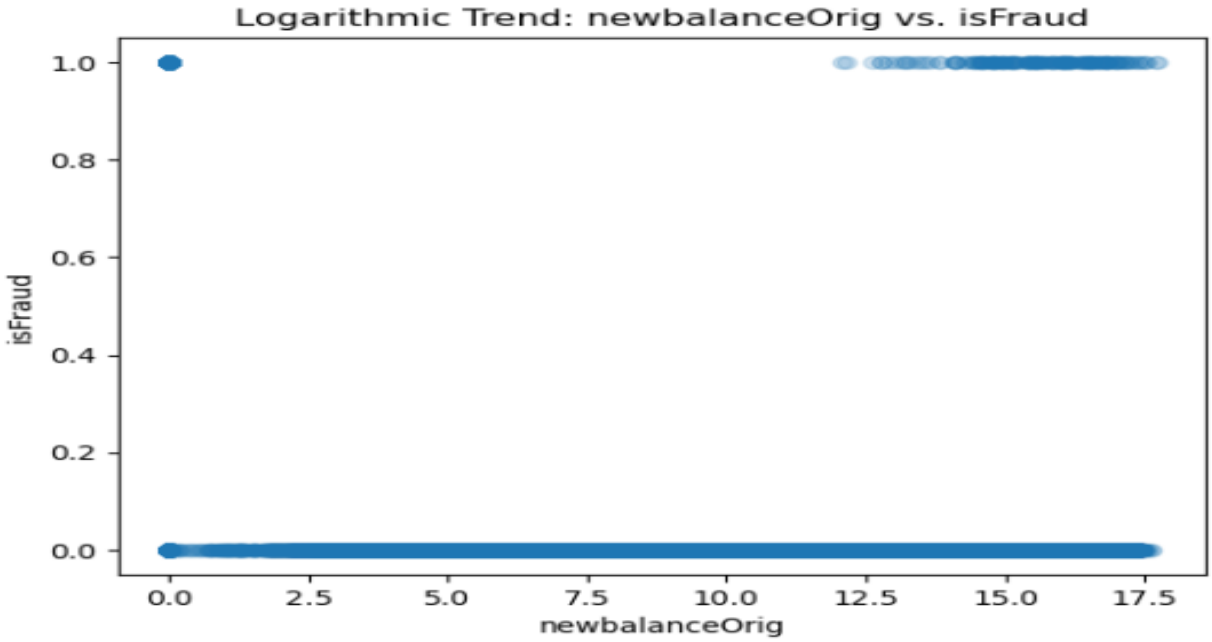

Figure.10: Logarithmic trend, newbalanceOrg

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

By looking at the outputs we can say that for this application regression tree is the best fit. The code analyses the "archive.csv" dataset by performing various data analysis and visualization tasks. Firstly, the dataset is loaded using pandas. Then, violin plots are created to visually explore the distribution of features in relation to the target variable. A correlation heatmap is generated to investigate the relationships between features and the target variable. The data is prepared for training by splitting it into training and testing sets. Missing values are handled through data imputation using a simple imputer. A decision tree classifier is trained on the training data. Predictions are made on the test set, and the accuracy score is calculated to evaluate the classifier's performance. A confusion matrix is created to visualize the classification results. Finally, a table visualization is generated to provide an overview of the dataset. Overall, this code provides valuable insights into feature distributions, correlations, and the effectiveness of the decision tree classifier for predicting the target variable. Figure 11.

A classification report summarizes the performance of a classification model on a dataset, providing an evaluation of its predictive accuracy for each class. It includes metrics such as precision, recall, F1-score, and support for each class. Precision measures the accuracy of positive predictions, while recall evaluates the model's ability to find positive instances. The F1-score combines precision and recall into a balanced measure of performance. Support indicates the number of instances for each class in the dataset. The classification report helps assess a model's strengths and weaknesses in classifying different classes, making it useful for comparing algorithms or different configurations.

Table 1: Model performance

Classification Report:

	precision	recall	f1-score	support
No fraud	1.00	1.00	1.00	1270904
fraud	0.89	0.88	0.88	1620
accuracy			1.00	1272524
macro avg	0.94	0.94	0.94	1272524
weighted avg	1.00	1.00	1.00	1272524

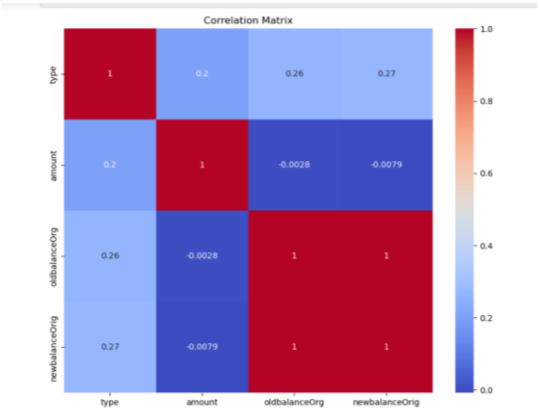

Figure 11: Correlation Matrix

VI. CONCLUSION AND FUTURE SCOPE

An online fraud detection system implemented using a decision tree algorithm shows promising results in detecting fraudulent transactions. A satisfactory accuracy score was obtained by analysing various characteristics such as transaction type, amount, old balance, and new balance. Decision tree algorithms are efficient at capturing complex patterns and making decisions based on feature values. Subdivide the data set based on the most informative features to create a tree structure for easier decision making. This algorithm has the advantage of being interpretable because the decision rules are easy to understand and visualize. Future Scope: Feature Engineering: Exploring additional features or transforming existing features may improve the model's performance. Consider incorporating time-based features, behavioural patterns, or derived features from domain knowledge. Ensemble Methods: Implementing ensemble methods such as Random Forest or Gradient Boosting can potentially enhance the model's accuracy and robustness. Ensemble methods combine multiple decision trees to make collective predictions, reducing overfitting and increasing overall performance. Anomaly Detection: Incorporating anomaly detection techniques alongside the Decision Tree algorithm can enhance fraud detection capabilities. Unsupervised learning methods like clustering or outlier detection can help identify suspicious patterns or outliers in transaction data. Real-time Monitoring: Integrating the model into a real-time monitoring system can enable continuous fraud detection and prevention. By implementing a streaming data pipeline and leveraging technologies like Apache Kafka or Apache Flink, the system can analyse incoming transactions in real-time. Improved Data Collection: Collecting more diverse and representative datasets can help enhance the model's generalization and effectiveness. Obtaining labelled data

Peer Reviewed & Refereed Journal
Volume-13, Issue-4 (August 2023)
https://doi.org/10.31033/ijemr.13.4.6

for both fraudulent and non-fraudulent transactions is crucial to improve the model's ability to accurately detect fraud. model evaluation. Thorough evaluation and validation of model performance on a variety of data sets and the use of appropriate evaluation metrics can provide a comprehensive assessment of performance. Consider testing the model's performance in different scenarios using methods such as cross-validation or A/B testing. Continuously updating and improving fraud detection systems, incorporating advanced algorithms, and continually updating new fraud models and methods are critical to ensuring effective fraud prevention in online transactions.

ABBREVIATIONS

- Online Fraud Detection (OFD)
- Decision tree (DT)
- False positive (FP)
- False negative (LN)
- True Positive (TP)
- True Voice (TN)
- Accuracy (ACC)
- Accuracy (PR)
- Recall (REC)
- F1 score (F1)

REFERENCES

[1] Abdallah, Aisha, Mohd Aizaini Maarof & Anazida Zainal. (2016). Fraud detection system: A survey. *Journal of Network and Computer Applications, 68*, 90-113.

[2] Hu, Nan, Ling Liu & Vallabh Sambamurthy. (2011). Fraud detection in online consumer reviews. *Decision Support Systems, 50*(3), 614-626.

[3] Minastireanu, Elena-Adriana & Gabriela Mesnita. (2019). An analysis of the most used machine learning algorithms for online fraud detection. *Informatica Economica, 23*(1).

[4] Zhang, Zhaohui, et al. (2018). A model based on convolutional neural network for online transaction fraud detection. *Security and Communication Networks*.

[5] Akoglu, Leman, Rishi Chandy & Christos Faloutsos. (2013). Opinion fraud detection in online reviews by network effects. *Proceedings of the International AAAI Conference on Web and Social Media, 7*(1).

[6] Chauhan, Nidhika & Prikshit Tekta. (2020). Fraud detection and verification system for online transactions: a brief overview. *International Journal of Electronic Banking, 2*(4), 267-274.

[7] Xu, Chang & Jie Zhang. (2015). Towards collusive fraud detection in online reviews. *IEEE International Conference on Data Mining*.

[8] Minastireanu, Elena-Adriana & Gabriela Mesnita. (2019). Light gbm machine learning algorithm to online click fraud detection. *J. Inform. Assur. Cybersecur*, 263928.

[9] Chang, Wen-Hsi & Jau-Shien Chang. (2012). An effective early fraud detection method for online auctions. *Electronic Commerce Research and Applications, 11*(4), 346-360.

[10] Kewei, Xiong, et al. (2021). A hybrid deep learning model for online fraud detection. *IEEE International Conference on Consumer Electronics and Computer Engineering*.

[11] Zhang, Ruinan, Fanglan Zheng & Wei Min. (2018). Sequential behavioral data processing using deep learning and the Markov transition field in online fraud detection. *arXiv preprint arXiv:1808.05329*.

[12] Chang, Wen-Hsi & Jau-Shien Chang. (2012). An effective early fraud detection method for online auctions. *Electronic Commerce Research and Applications, 11(*4), 346-360.

[13] Cao, Shaosheng, et al. (2019). Titant: Online real-time transaction fraud detection in ant financial. *arXiv:1906.07407*.

[14] AC, Ramachandra & Venkata Siva Reddy. (2022). *Bidirectional DC-DC converter circuits and smart control algorithms: A review*.

[15] Kumari, Ashwini, et al. (2018). Multilevel home security system using arduino & gsm. *Journal for Research 4*.

[16] Viswanatha, V., et al. (2020). Intelligent line follower robot using MSP430G2ET for industrial applications. *Helix-The Scientific Explorer, 10*(02), 232-237.

[17] Viswanatha, V. & R. Reddy. (2020). Characterization of analog and digital control loops for bidirectional buck–boost converter using PID/PIDN algorithms. *Journal of Electrical Systems and Information Technology, 7*(1), 1-25.

[18] Viswanatha, V., R. K. Chandana & A. C. Ramachandra. (2022). *IoT based smart mirror using raspberry pi 4 and yolo algorithm: A novel framework for interactive display*.