International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

Support Vector Machine based Credit Card Fraud Detection

Sushant Kumbhar
Dept. of Computer Engineering JSCOE Savitribai Phule Pune University Pune, INDIA

Ashish Lade
Dept. of Computer Engineering JSCOE Savitribai Phule Pune University Pune, INDIA

Jaykishan Pandey
Dept. of Computer Engineering JSCOE Savitribai Phule Pune University Pune, INDIA

Prof. A. B. Ghandat
Dept. of Computer Engineering JSCOE Savitribai Phule Pune University Pune, INDIA

Abhishek Patil
Dept. of Computer Engineering JSCOE Savitribai Phule Pune University Pune, INDIA

***Abstract*—Credit card fraud is a major concern in today's world, as it involves the illegal use of credit cards to obtain goods or services. The process of credit card fraud often involves "laundering" dirty money, making it difficult to trace the source of funds. With the large volume of financial transactions happening globally, It might be difficult to identify credit card theft. In the past, anti-fraud suites were introduced to detect suspicious activity on individual transactions, but they were not effective in detecting fraud across multiple bank accounts. To overcome this challenge, we propose using machine learning techniques, specifically the 'Structural Similarity' method, to identify common attributes and behavior across multiple bank account transactions. It can be challenging to identify credit card fraud from huge datasets, so we also suggest utilizing case reduction techniques to shrink the input dataset and then looking for pairs of transactions with similar characteristics and behaviours.**

Keywords- Support Vector Machine, Haar Cascade Algorithm, Fraud Detection

I. INTRODUCTION

The COVID-19 pandemic in 2020 led to a surge in online shopping as travel was restricted and many were confined to their homes. This increase in online shopping also resulted in a significant rise in online financial fraud, particularly involving credit and debit cards. According to a NCRB study, there was a 225% increase in such fraud incidents in 2020 compared to 2019, with a total of 1194 cases reported in 2020 as opposed to 367 in 2019. As a result of the pandemic, many people were introduced to the convenience of internet shopping, but also had to face the unfortunate reality of online fraud.

AIMS & OBJECTIVES

The aim of this project is to classify credit card transactions as either fraudulent or legitimate by using various classification techniques. It offers a user-friendly interface where users can upload a csv file containing transaction and payment information and determine whether the transactions fall into the fraudulent or non-fraudulent category. The core objectives of this project are to identify fraudulent transactions and to provide a solution to the real-world problem. The steps taken to achieve these goals are through the implementation of various classification techniques. The steps followed to manage these goals:

- Selection of dataset
- Visualization of the features through graphical information
- Identification and handling of null values in the dataset
- Data preprocessing to eliminate unnecessary parameters
- Training using SVM classifiers
- Evaluation of the model using test data
- Comparison of accuracy, precision, and recall to determine the optimal model
- Development of a Graphical User Interface to apply the model to real-time customer data and predict if transactions are legitimate or fraudulent.

II. LITERATURE SURVEY

The authors of this study evaluate the effectiveness of three machine learning algorithms in detecting credit card fraud by applying them to a set of fraud data. They compare the performance of the Random Forest, Decision Tree, and XGBOOST algorithms, and find that the Random Forest approach shows the highest accuracy in identifying fraud cases [8].

The authors present a technique to assess the daily changes in a face-to-face credit card transaction dataset, which pertains to cardholders who make purchases in-store. The method involves comparing the transactions of different days by means of a classification process, which evaluates the efficiency of the classification. The more effective the classification, the greater the difference in buying patterns between two days, and vice versa. This leads to the creation of a distance matrix that summarizes the shift in the dataset [6].

The dataset for credit card fraud detection from Kaggle machine learning, which has a highly skewed distribution, was employed in this study. The evaluation focuses on features that are marked either 1 for fraud or 0 for non-fraud cases. Fraud detection is a crucial aspect in the banking industry, and the existing systems suffer from misclassifications and high false positive rates. The paper explores the use of convolutional neural network layers to develop a model for detecting credit

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

card fraud, with the goal of achieving a high level of accuracy [7].

Credit card fraud detection is a widely researched subject, with a significant amount of research being conducted. Various statistical methods have been used to develop solutions for detecting credit card fraud are as follows:

A. K-nearest neighbor:

K-nearest neighbor (KNN) is a simple and widely used supervised learning technique for classification. The basic idea behind KNN is to divide the data into different clusters by defining K centroids, one for each cluster. However, the placement of these centroids can impact the accuracy of the results, as different locations can result in different outcomes. A crucial aspect of KNN is selecting the appropriate value of K, as a small value increases the likelihood of overfitting while a larger value may lead to increased computation time and underfitting of the model. When presented with new, unseen data, the KNN classifier searches the pattern space for the K closest examples, and uses them to make a prediction.

B. Logistic Regression:

Logistic Regression is a method for predicting discrete outcomes by using a combination of continuous and discrete predictors. In the case of binary logistic regression, the dependent variable has two levels. The goal of the analysis is to understand the impact of various explanatory variables, which can be either numerical or categorical, on the binary outcome of interest. The objective is to identify the most appropriate model that describes the relationship between the binary outcome and the set of independent variables.

C. Decision tree:

A decision tree is a visual tool that is used to aid in the process of decision making. It is represented as a diagram that has a flowchart-like structure, in which the nodes represent tests on different attributes, the branches represent the results of those tests, and the leaf nodes represent the final decisions or class labels. Decision trees are closely related to another decision-making tool, called influence diagrams, which are also used to calculate the potential outcomes of different options.

III. PROBLEM STATEMENT

Due to activities like illegal purchases, identity theft, and the use of fake cards, credit card fraud is a severe threat for financial institutions, businesses, and customers everywhere. Credit card fraud detection's major goal is to spot and stop fraudulent transactions in real-time while reducing the amount of false positives to prevent interfering with legal transactions. A trustworthy and effective fraud detection system that can examine a large number of transactions, identify odd trends, and rapidly notify pertinent parties of the need for action is needed in order to do this. The ultimate goal is to create a reliable and efficient fraud detection system that will guard against fraudulent activity and safeguard all stakeholders' financial interests.

IV. SYSTEM ARCHITECTURE

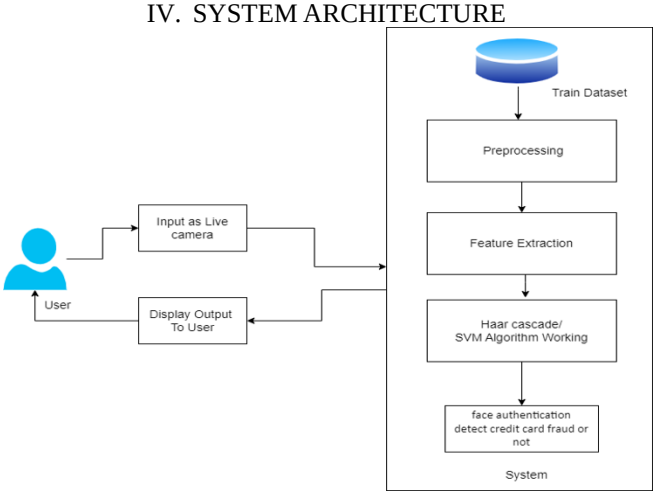

Fig. 1. Architecture Diagram.

V. ALGORITHMS

A. Support Vector Machine Algorithm

A popular approach in supervised machine learning called the Support Vector Machine (SVM) is particularly good at handling classification and regression issues. Finding a decision boundary or hyperplane that can successfully divide the data into multiple classes is the basic tenet of SVM. To do this, the algorithm identifies the most critical data points in the dataset, which are called support vectors. These support vectors are the key to creating a hyperplane that separates the data effectively, which is the primary objective of the SVM algorithm. The support vectors, which are the most extreme points in the dataset, are used to determine the decision boundary.

Support Vector Machine (SVM) is a powerful algorithm used in supervised machine learning for solving classification and regression problems. It works especially well when the data cannot be separated into classes using a straight line because this means the data cannot be separated into classes linearly. In these circumstances, SVM employs a method known as the kernel trick, which raises the data's dimension and makes it linearly separable. This enables the algorithm to find a decision boundary or hyperplane that can effectively classify the data.

Regression issues, where the algorithm discovers a function that can forecast a continuous value for a given input, can also be solved using SVM. The hyperplane-based function aims to minimize the difference between the expected and actual values. SVM is resilient to noise and overfitting and includes a regularization parameter that can aid in preventing overfitting. In summary, Support Vector Machine (SVM) is a powerful algorithm that finds a wide range of applications in machine learning, particularly for tasks such as binary classification, text categorization, bioinformatics, and face detection. It is a valuable tool because it can handle non-linear data, determine the optimum decision boundary by figuring out the support vectors, and is based on the principle of structural risk minimization. An equation representing the SVM decision function, An equation that expresses the SVM decision function and a representation of the SVM decision function are both essential components of the algorithm is Eq. (1)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

The Support Vector Machine (SVM) algorithm uses an

$$f(x) = sgn(x.w) + b \quad (1)$$

equation, known as Eq. (1), to determine the boundary that separates two classes. This equation includes an input vector, x, which comprises of weight and a constant term, b. The algorithm learns the values of the weight, w, and the constant, b, during the training phase, to classify new instances. The goal of the SVM approach is to find the ideal values of w and b in order to maximise the separation between the two classes. The algorithm's ultimate goal is to increase the margin between the two classes.

The SVM algorithm uses a metric called the Margin, which is the distance between the two hyperplanes, to determine the optimal hyperplane that separates the two classes. The larger the margin, the better the separation of the two classes. The SVM algorithm aims to maximize this margin to find the best decision boundary by $H : y = w.x + b = 0$ and two hyper planes are $H1 : y = w.x + b = +1$ & $H2 : y = w.x + b = -1$.The H stands for the threshold that divides the two classes, and $H1$ and $H2$ represent the margins that separate the two classes. $2/||w||$ denotes the margin. The vector's w's norm is represented as $||w||$. When the data is not perfectly separable, meaning there are instances of overlap between the two classes, the margin becomes "soft". This increases the likelihood of misclassification errors. To reduce these errors, the SVM algorithm aims to maximize the margin. By maximizing the margin, the algorithm tries to find the best decision boundary that separates the two classes with the least amount of errors. The slack variable $\sum i = 0$ is used to minimize the errors. We can show correctly classified classes using $\xi_I = 0$. Let assume the non-negative slack variable for misclassification is ξ_i.

In the instance of fraud detection, y = 1 for the positive class, therefore let y be the indication of the class and y = −1 is a class that represents a negative class. Either $x.w + b >= 1 - \sum i$ or $x.w + b >= -1 + \sum i$, which can be summed up by Eq, is necessary for SVM (2),

where, i = 1,2, . . . , n

Eq. (3) in SVM defines the optimization problem for figuring out the values of w and b.

$$y_i(w.x + b) >= 1 - \sum i \quad (2)$$

$$Min\frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\xi_i \quad (3)$$

SVM reduces complexity by minimizing $||w||^2/2$ and minimizes misclassification errors by minimizing the slack variable. The regularization parameter C is used to balance the trade-off between the two classes and weighs the classification errors. Utilizing the Lagrange function 4, the constrained optimization problem is resolved [10].

$$L(w,b,\xi,\alpha,\beta) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i\{y[wx+b]-1+\xi_i\} - \sum_{i=1}^{n}\beta_i\xi_i \quad (4)$$

$$\max_{\alpha\beta} w(\alpha,\beta) = \max_{\alpha\beta}\left\{\min_{w,b,\xi}(w,b,\xi,\alpha,\beta)\right\} \quad (5)$$

By minimising w, b & ξ and maximising α & β, the optimization problem is solved. The dual formulation in Eq. should be introduced to the problem in order to better resolve it (5),

This substitution transforms the issue into its twin formulation, which is defined by,

$$\max\left\{\sum_{i=1}^{n}\alpha - \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \langle x_i, y_j\rangle\right\} \quad (6)$$

and is optimised within the restrictions,

$$\sum_{i=1}^{n}\alpha_i y_i = 0 \text{ and } 0 \le \alpha_i \le C \text{ for } i = 1, 2, .., n \quad (7)$$

Eq. (7) is subjected to the Kuhn-Tucker condition in Eq (6),

$$\alpha_i\{y_i[w \cdot x_i + b] - 1 + \xi\} = 0_i \quad (8)$$

where, i = 1, 2, . . ., n.

The Lagrange vectors, sometimes referred to as support vectors, are used to characterise the hyperplane. When the data can be separated linearly, all support vectors are on the edge. Eq. (8) in the SVM algorithm determines the decision threshold.

$$f(x) = \sum_{i=1}^{Ns}\alpha_i y_i \langle x, x_i\rangle + b \quad (9)$$

The SVM algorithm uses an equation, This comprises the bias term b, the inner product of (x, xi), the number of support vectors NS, and the input vector x. When the data is nonlinear, meaning it cannot be separated by a straight line, a technique known as the kernel trick is applied to transform the input vector into a higher dimensional feature space. This enables the algorithm to find a decision boundary, also known as a hyperplane, that can effectively classify the data. Utilizing kernel functions, which offer a technique to extend linear algorithms to non-linear data by expressing them in terms of dot products, allows for the implementation of the kernel trick. Any function that meets Vapnik's definition of Mercer's condition can be employed as a kernel function [11]. The equation, which is a formal description of a kernel function in the SVM method,

The kernel function can transform the input vectors into a higher dimensional feature space.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

$$\langle x_i, x_j \rangle \rightarrow k\left(x_i, x_j\right) \quad (10)$$

B. Haar Cascade Algorithm

Identification of instances of things in photos and videos is the focus of the field of object detection, which combines computer vision, image processing, and deep learning. In this article, we will use a technique called Haar cascades for object detection. Paul Viola and Michael Jones created, Haar Cascade classifiers are a popular method for object detection. They are machine learning-based and trained using a large number of positive and negative images. Negative photos are of everything else that does not include the target object, whereas positive images are of the objects we want to detect.

Haar cascade is an algorithm that is capable of detecting objects in images, regardless of their size and location. It is a relatively simple algorithm that can operate in real-time. Haar cascade uses a cascading window approach, where it analyses the features in each window to determine whether an object could be present. The approach analyses windowsized sections of the picture to compute and match features using sample Haar features. It performs the role of a classifier by classifying data points as positive if they are a part of the identified item and negative if they are not.

Haar cascades are efficient and can perform well in real-time applications, however, they are not as precise as more recent object detection methods. They also have a downside of predicting many false positives. They are simple to implement and require less computing power. Using a collection of both positive and negative photos, the algorithm is trained. Positive photos are those we wish to detect, and negative images are those without the thing we're trying to find. The algorithm uses the cascading window approach, where it computes features in each window and classifies whether it could be an object. The Haar cascade features are used to traverse window-sized areas of the image to compute and match features. The algorithm uses the concept of weak classifiers and strong classifiers to detect the object. The weak classifier is a simple classifier that is not very accurate but when combined with many other weak classifiers it will form a strong classifier that is very accurate.

The Haar cascade algorithm is commonly used in real-time applications such as face detection, object tracking, and video surveillance. It is also applied in driver assistance systems in vehicles to detect pedestrians and other objects on the road. Despite its widespread usage, the Haar cascade algorithm has some limitations. It is not as precise as other contemporary object detection techniques and it is sensitive to changes in the lighting conditions of the image.

VI. METHODOLOGY

A. Methodology Description

A machine learning model for detecting credit card fraud must be built using a few crucial steps:

- Uploading the dataset: To start creating the model, the dataset comprising data on credit card transactions must be uploaded to the system.
- Data pre-processing and selection: At this point, the dataset's features are transformed into numerical data and the relevant attributes that best represent the behaviour of a particular credit card account are selected. It is important to only include features that are relevant to the task at hand, as including irrelevant features can make the classifier less efficient.
- Feature extraction: In this step, the input data are condensed into a smaller set of features that represent the key aspects of the data. For this, principal component analysis (PCA) is a helpful tool.
- SVM training and classification: A popular supervised machine learning approach for classification tasks is the Support Vector Machine (SVM). The SVM algorithm seeks to identify the best hyperplane in an N-dimensional space for classifying the data into distinct groups.
- Evaluation: Evaluation of the model's performance is the last phase. While accuracy is essential, when talking about credit card fraud detection, it is advised to use the fraud catching rate and false alarm rate. An effective method for assessing these indicators is the confusion matrix.

VII. METHODOLOGY

As stated in the reference [9], while accuracy is important in the fraud detection field, the metrics that matter most are the fraud catching rate and false alarm rate. The confusion matrix is used in this study to assess these rates. The confusion matrix is presented here in its conventional format.

Fig. 2. Confusion Matrix

		Predicted	
		Positive	**Negative**
Actual	Positive	TP	FN
	Negative	FP	TN

The anticipated and actual classes are denoted in the column and row, respectively, of the confusion matrix used in credit card fraud detection. The amount of legitimate transactions that were successfully classified as non-fraudulent transactions is shown by the True Positive (TP) value, which stands for the fraud catching rate. The false alarm rate, or false positive (FP) figure, shows the number of legitimate transactions that were mistakenly labelled as fraudulent. The number of fraudulent transactions that were mistakenly thought to be real is shown by the False Negative (FN) value. The number of fraudulent transactions that were accurately classified as such is shown in the True Negative (TN) figure.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

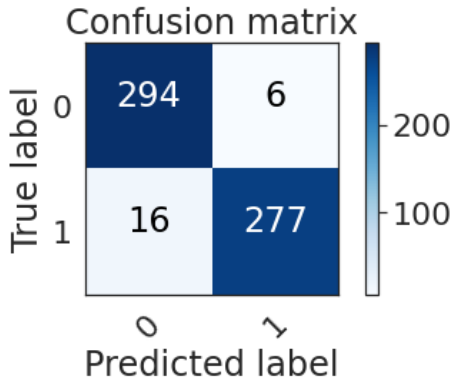

Fig. 3. Confusion Matrix

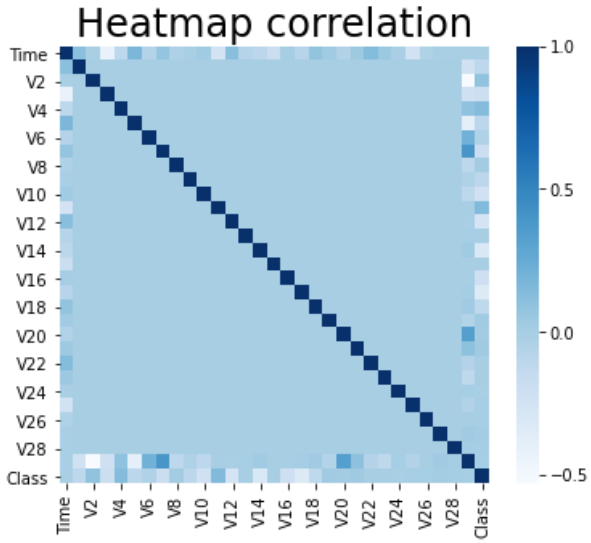

Fig. 4. Correlation of the features

VIII. CONCLUSION

In this paper, a novel strategy for combating credit card fraud is provided. It makes use of a behavior-based categorization algorithm with the assistance of Support Vector Machines (SVM). A huge number of financial transactions can be analysed by the proposed machine learning framework to look for possible money laundering activities. In order to increase the effectiveness of the framework, a number of case reduction techniques are used to cut down on the number of probable money laundering accounts that need to be examined, incorporating balance score filtering and matching transaction detection. The framework combines accounts that are more likely to be utilised in credit card fraud using the structural similarity method, facilitating the easy detection of patterns and trends that more traditional organising methods tend to conceal. The suggested framework offers a high degree of accuracy in detecting money laundering activities, according to the preliminary experimental results, making it a promising tool for preventing credit card fraud.

REFERENCES

[1] "Fatf-gafi.org - Financial Action Task Force (FATF)", Fatf-gafi.org,2016. [Online]. Available: http://www.Fatf-gafi.org. [Accessed: 22-Dec- 2015]

[2] Fatf-gafi.org, 'credit card fraud - Financial Action Task Force (FATF)', 2014.[Online]. Available: http://www.fatfgafi.org/faq/moneylaundering/. [Accessed: 22- Dec- 2015].

[3] Neo4j Graph Database, 'Neo4j, the World's Leading Graph Database', 2014. [Online]. Available: http://neo4j.com/. [Accessed: 22- Dec- 2015].

[4] A. C. Bahnsen, A. Stojanovic, D. Aouada, and B. Ottersten. Improving credit card fraud detection with calibrated probabilities. In SDM, 2014.

[5] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han. Outlier Detection for Temporal Data. Synthesis Lectures on Data Mining and Knowledge Discovery, Morgan \& Claypool Publishers, 2014.

[6] Yvan Lucas1,2, Pierre-Edouard Portier1, Lea Laporte, Sylvie Calabretto1, Liyun He-Guelton3, Frederic Oble3 and Michael Granitzer2 Dataset shift quantification for credit card fraud detection, 2019.

[7] Anu Maria Babu, Computer Science and Engineering, Saintgits College Of Engineering and Dr. Anju Pratap Computer Science \& Engineering Saintgits College Of Engineering Kerala, India, Credit Card Fraud Detection Using Deep Learning, 2020.

[8] Vinod Jain, Mayank Agrawal and Anuj Kumar Assistant Professor, Department of Computer Engineering and Applications, GLA University, Mathura. Performance Analysis of Machine Learning Algorithms in Credit Cards Fraud Detection 2020.

[9] Salvatore J Stoflo, David W Fan, Wenke Lee and Andreas L Prodromidis and Philip K Chan, “Cost –Based Modeling for Fraud and Intrusion Detection: Results from the JAM Project”, Proceedings of the DARPA Information Survivability Conference and Exposition, Vol. 2, pp. 130-144, 2000.

[10] Dheepa V., Dhanapal R.. "BEHAVIOR BASED CREDIT CARD FRAUD DETECTION USING SUPPORT VECTOR MACHINES", ICTACT Journal on Soft Computing, 2012

[11] Nan Li, Zetian Fu, Wengui Cai, Xiaoshuan Zhang. "Using Support Vector Machines to Predict the Variation of Organic Pollutants in Pond Water", Third International Conference on Natural Computation (ICNC 2007), 2007.