Improving credit card fraud detection using machine learning and GAN technology

Najwan Thair Ali[1,*] , Shahad Jasim Hasan[2], Ahmad Ghandour [3] and Zainab Salam Al-hchimy[4]
[1] Al-Mustaqbal University,Babil, Iraq
[2] Computer Center Babylon University ,Babil, Iraq
[3] Communication Engineering Islamic University ,Wardanieh ,Lebanon
[4] University of Alkafeel, Najaf, Iraq

Abstract. The motivation behind this study stems from identifying contemporary challenges associated with prosecuting electronic financial crimes. Highlights ongoing efforts to identify and address credit card fraud and fraud as there are many credit card fraud issues in the financial industry. Traditional methods are no longer able to keep up with modern methods of tracking the behavior of credit card users and detecting suspicious cases. Artificial intelligence technology offers promising solutions to quickly detect and prevent future fraud by credit card users. Datasets used to detect financial anomalies are affected by imbalances in financial transactions, and this study aims to address the imbalance of financial fraud datasets using adversarial algorithm techniques and compare them with the most commonly used methods in the scientific literature.The results showed that the function of the adversarial algorithm is consistent in several ways, including allowing researchers and interested parties to determine data growth rates, which helps bring the dataset closer to real-time data from financial markets and banks. This study proposes a hybrid machine learning model consisting of three machine learning algorithms: decision trees, logistic regression, and Naive Bayes algorithm, and calculates performance metrics such as accuracy, specificity, precision, and F1 score. Experimental results reveal varying degrees of accuracy in fraud detection. Model testing using the SMOTE method recorded an accuracy of 98.1% and an F-score of 98.3%. On the other hand, the oversampling and under sampling test methods showed similar performance, with the two methods recording an accuracy of 94.3 and 95.3 and an F-score of 94.7 and 95.1, respectively. Finally, the GAN method excelled, receiving a test score and accuracy of 99.9%, as well as exceptional precision, recall, and F1 score. As a result, we conclude that the GAN method is able to balance the data set, which in turn is reflected in the performance of the model in training and the accuracy of predictions when tested. Historical transaction analysis identifies behavioral patterns and adapts to evolving fraud techniques. This approach enhances transaction security and protects against potential financial losses due to fraud. This contribution allows financial institutions and companies to proactively combat fraudulent activities.

1. INTRODUCTION

The surge in websites and mobile applications has made online financial transactions a common practice, providing users with convenience and instant purchasing experiences. However, this ease has created a significant threat of fraudulent online transactions spreading widely, leading to unauthorized access and financial losses.[1]Financial institutions face enormous hurdles in detecting fraudulent activity within vast amounts of transaction data. The pivotal role that anomaly detection plays in identifying suspicious patterns and avoiding financial losses is undeniable. However, we note the difficulty of analyzing the complexities of sophisticated financial fraud using statistical methods and traditional rule-based methods [2].The modern era witnessed the availability of an electronic financial means known as the credit card. Which has spread widely in financial industry transactions due to the ease of use it provides. The credit card also provided many advantages to users, including free and completely interest-free deposits, obtaining personal loans, and electronic payment.[3]. On the other hand, the use of credit cards carries many risks related to the potential breach of financial privacy. Or manipulation of the user's financial balances, or the user himself performs manipulation, such as defaulting on payment, which prompts banks and banks to take strict measures such as stopping cards that record many financial manipulations and frauds [4].

[5]Models based on artificial intelligence and all its branches offer promising solutions in the field of tracking and detecting abnormal user behavior when using credit cards. The detected fraud methods and their classifications are shown in Figure 1[6].

* Corresponding author: najwan.thaeir@mustaqbal-college.edu.iq

Finally, through this research, we seek to focus on the financial trading data set by applying smart techniques that address the complexities in the data set and make it closer to real-world data. This contributes to the design and development of smart models capable of tracking user behavior and detecting abnormal patterns in financial trading.

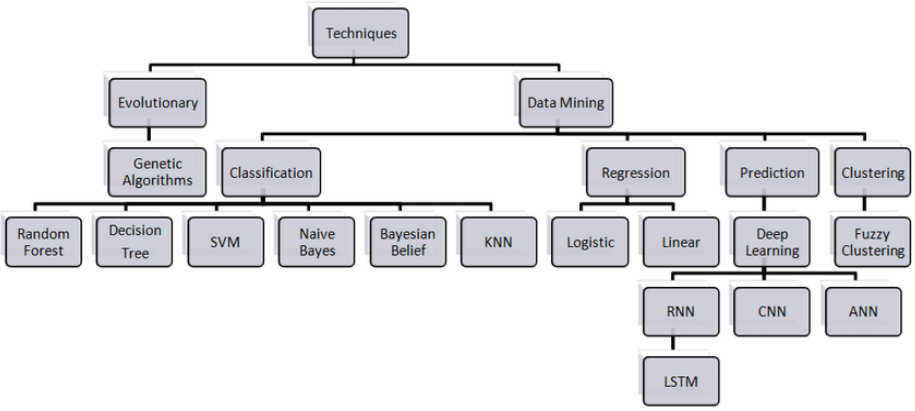

Fig.1 .Taxonomy for Frauds [6].

This research paper consists of the following sections. The second section presents work related to the proposed study, and provides a general analysis of the results. The structure of the proposed model is presented in the third section. While the methodological stages of the work are explained in the fourth section, which includes the data collection and the pre-processing stage. The fifth section presents the experimental results achieved, and the sixth section focuses on Discuss them, and finally, Section Seven presents the final conclusions

2. Related Work

The section presents the most important literature of previous works and their methodology based on machine learning in all its branches for the purpose of detecting and reducing abnormal behavior in financial transactions on credit cards. It also presents the most important techniques used in the pre-processing stages, which include balancing the categories in the data set used, studying the characteristics of the features, etc.

1. Mijwil et al. (2020)presents a proposal to evaluate the performance of three machine learning models – C4.5 Decision Tree, Bagging Ensemble, and Naïve Bayes – in predicting normal and fraudulent financial transactions. The results showed that the precision-recall curve areas for class 0 were very high, indicating excellent performance in distinguishing binary class 0. As for class 1, the Naïve Bayes model achieved 81%, the C4.5 model achieved 75.6%, and the Bagging model achieved 81%. Ensemble achieved a score of 83.8%. This confirms that the C4.5 Decision Tree model showed good performance, while the Bagging model was considered ideal and the Naïve Bayes model was acceptable in terms of detection.

2. Recommended by Ileberi et al. In their paper presented by B (2022) using different computational techniques, including genetic algorithms, artificial neural networks, logistic regression, fuzzy logic, decision trees, Bayesian networks, support vector machines, as well as hidden Markov models, and K-nearest neighbors, to build a model Intelligent credit card fraud detection. The prediction rate was high, reaching 99.6%.

3. Afriyie et al. (2023 Afriyie et al. (2023) presented a proposed model that focused on identifying and reducing fraudulent credit card transactions using a set of machine learning algorithms, namely random forest, decision tree, in addition to logistic regression. The accuracy of the models was evaluated based on the metrics of accuracy, sensitivity, specificity, accuracy, F1 score, AUC, and ROC. The highest accuracy recorded in the study was for the Random Forest algorithm, which reached 96%, with a sensitivity of 0.97 and a specificity of 0.95.

4. Cherif et al. (2022): The effectiveness of machine learning strategies in financial fraud detection is thoroughly analyzed in this study, using different statistical metrics such as ROC, MCC, G-mean, AUC, K-S, cost reduction rate, average run duration, and alert rate. The study presents a proposed credit card financial fraud detection system based on an artificial neural network (ANN) called AFDM. The model combines user behavior analysis with M-class support vector machine (SVM) and random forest (RF) classifiers. The study achieved an amazing accuracy of 99.99%.

5. Asha et al. (2021): Proposed hybrid models and unsupervised credit card fraud detection using autoencoders, achieving accuracy rates of 82.58% and 99.92% with Random Forest and ANN, respectively.

6. Rtayli et al. (2020This study presents the proposed hypernoise parameter model for credit card fraud (CCF) detection. Which consists of a set of machine learning algorithms GridSearchCV, SVM-RFE, SMOTE, and to enhance detection the RFC method (HPO, RFE) was used. The proposed model achieved 95% accuracy for DB3 and 100% accuracy for DB1 and DB2.

The model was also evaluated based on sensitivity, specificity, F-score, and accuracy. The model recorded a sensitivity of 100% for DB1 and DB2, 95% for DB3, and an AUC of 0.99.

7. Chang et al. (2022): The study provides a comparison involving the techniques Decision Tree, Logistic Regression, k-Nearest Neighbors, Random Forest, and Autoencoder on their effectiveness in detecting fraud in digital payment. The performance of the models was evaluated in terms of accuracy, sensitivity, and specificity. The random forest and logistic regression algorithms showed superior performance, achieving an accuracy rate of more than 96%, a sensitivity rate of 81%, and a specificity rate of 97%.

8. Sadineni et al. (2020): . The study examines the importance of precision, accuracy, and false alarm rate metrics. The authors emphasize the impact of data set characteristics and size on the effectiveness of the strategies used (artificial neural networks, decision trees, support vector machines, logistic regression, and random forests). The results indicate that random forest and artificial neural network (ANN) achieved the highest levels of accuracy, reaching 99.21% and 99.92%, respectively.

9. Karuna Chandra et al. (2023): This research also presents a method that combines deep learning techniques, including convolutional neural networks (CNN), long short-term memory (LSTM), and machine learning techniques represented by K-Nearest Neighbor (k-NN), to identify email fraud. -Trading commissions. . The results indicate that the k-NN algorithm achieved the highest accuracy among the algorithms examined, reaching 83.82 percent. In terms of validation and test accuracy, the CNN model achieved 63.44 and 49.39 percent, respectively, while the LSTM model scored 54.4 and 51.13 percent, respectively. The combined CNN-LSTM technique recorded an accuracy of 53.58 percent and a test accuracy of 52.18 percent. The authors recommend the use of deep learning algorithms in future studies to enhance the accuracy of fraud detection.

10. Błaszczyński et al. (2021): This paper provides an overview of different approaches to credit card fraud detection and introduces the DRSA-BRE method, specifically designed to address concerns associated with imbalanced data sets. The program showed an accuracy rate of 79.09% in identifying fraudsters and 82.56% in identifying non-fraudsters. It is worth noting that the DRSA-BRE model outperforms the Random Forest and Support Vector Machine models in the context of fraud scenarios.

After considering the literature and previous research, the prominent role and great effectiveness of artificial intelligence techniques in the field of credit card fraud detection is evident. A variety of approaches have been explored, including traditional algorithms and advanced techniques in the fields of machine learning and deep learning. Particular emphasis has been placed on addressing imbalanced datasets using techniques such as SMOTE and Near-Miss. Feature selection and hyperparameter tuning are vital to achieving excellent performance. This paper presents the use of GAN to process imbalanced datasets and compares it with techniques used in the literature. [7-16]. Table 1 shows summary of previous researches

Table 1: Summary of previous researches.

No.	Reference	Dataset	Features Extraction	Model	Weaknesses
1	[7]	The dataset includes more than 297,000 credit card transactions in September 2013 and November 2017 collected from the Kaggle platform, of which 3,293 were fraud transactions.	Taking advantage of the properties of algorithms to extract important features	Naïve Bayes, C4.5 Decision Tree,Bagging Ensemble Learning	1- The research paper does not address feature engineering 2- The research paper did not address the imbalance in the data set
2	[8]	credit card transactions	Synthetic Minority Oversampling Technique (SMOTE) method	Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), Artificial Neural Network (ANN), and Naive Bayes (NB)	Despite using one of the data set normalization methods, the results suffer from overfitting
3	[9]	Credit Card Transactions Fraud Detection Dataset	The synthetic minority oversampling (SMOTE) technique was used to balance the data. However, an undersampling technique was used to address the imbalance in the data set	Decision Tree, Logistic Regression , Random Forest	The results suffer from lack of consistency, and this indicates that the techniques used did not solve the problem of imbalance in the data set
4	[10]	Review	Screened and categorized a selection of 40 relevant articles based on the topics covered (e.g. the problem of class imbalance). and feature engineering) and machine learning techniques used (including traditional and deep learning models).	Review	limited exploration of deep learning

5	[11]	Credit Card Transactions Fraud Detection Dataset	-	Artificial Neural Network (ANN), support vector machines(SVM), k-nearest neighbors(KNN).	Failure to use data set balancing techniques, which reflects the unreliability of the results
6	[12]	Credit Card Transactions	Support vector machine redundant feature removal (SVM-RFE) technique to extract the best predictive features	Random Forest Classifier	-
7	[13]	a real credit card transaction dataset.	undersampling and feature reduction method using principal components analysis	logistic regression (LR), decision tree (DT), k-nearest neighbors (KNN), random forest (RF), and autoencoder	The effect of undersampling and oversampling is different for each algorithm which affects the prediction accuracy and reliability
8	[14]	credit card transaction dataset	none	Artificial Neural Networks , Decision Tree, Support Vector Machine, Logistic Regression and Random Forest .	The weak points are in the architecture of the algorithms used, as the ANN depends on the architecture of the devices used, and the DT suffers from over-equipping. While SVM requires more training time for larger data sets, RF suffers from excessive sensitivity to data with diverse values and attributes.
9	[15]	Fraudulent cashback transactions in the e-commerce sector in Indonesia	Feature extraction was performed using five different algorithms. Algorithms included Pearson correlation, which determines correlations between two columns based on values close to -1 or 1; Chi-square, which defines the degrees of freedom of the column indicating the strength of the relationship; Recursive Feature Elimination (RFE), which iteratively removes features until a specified number is reached; Least absolute shrinkage and selection operator (LASSO), a method for organizing features; and Random Forest, which includes decision trees for prediction through voting	K-Nearest Neighbor (k-NN), Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM)	The results showed a decrease in the accuracy of deep learning algorithms
10	[16]	credit card fraud data.	the ADASYN method is employed to achieve a balanced dataset	logistic regression, K-nearest neighbors, random forest, naive bayes, multilayer perceptron, ada boost, quadrant discriminative analysis, pipelining and ensemble learning	Accuracy varied for the categories of the data set used, as the models recorded low accuracy for fraud transactions, and this indicates that the method used to balance the data set is not appropriate.

3. Overview the Proposed System

The proposed methodology for building an artificial intelligence model to track user behavior in a credit card to detect financial fraud using machine learning, which also provides a precise overview of using the GAN algorithm to balance data sets and compares it to traditional methods. When implementing the proposed system, we used Python as a programming

language. Python is considered the most versatile language in the world of programming. With applicability across diverse areas of software development, Python not only simplifies existing programming tasks, but also allows developers to focus on the core functionality of their projects. Being the fastest growing programming language, Python finds its place in every emerging field. It is capable of developing diverse applications such as Natural Language Processing (NLP), which includes creating applications and services that can understand human languages, and traditional spam detection to understand content. The Python language plays an essential role in shaping technological progress and its importance in driving innovations across various fields[28]. The Python environment can be considered one of the best environments for implementing machine learning scenarios. Python offers rich packages and libraries used in machine learning [29]. The proposed model consists of several main stages that will be explained in the following section:

A. Python libraries for machine learning

a) NumPy: NumPy, short for Numerical Python, is the cornerstone of numerical computation in the Python ecosystem, featuring a powerful N-dimensional array object. NumPy is a versatile and indispensable package. Designed as a general-purpose matrix processing toolkit, it provides high-performance multidimensional arrays and accompanying tools for seamless processing. To address performance concerns, NumPy speeds up calculations using efficient multidimensional arrays, along with specialized functions and operators. NumPy finds widespread utility in data analysis, facilitates the creation of robust N-dimensional matrices, forms the base layer for other libraries such as SciPy and scikit-learn, and represents a compelling alternative to MATLAB, especially when combined with SciPy and Matplotlib[30].

b) Keras: Keras, a prominent open source neural network library for Python, was initially designed for ONEIROS (Open-Ended Intelligent Electronic Neural Robot Operating System) by a Google engineer, and seamlessly integrated into the core TensorFlow library. A favorite in the machine learning community, Keras is a high-level neural network API compatible with TensorFlow, CNTK, or Theano, and is scalable for both CPU and GPU environments. Known for its user-friendly approach, Keras makes prototyping easy and fast, making it an excellent choice for beginners venturing into neural network design. The library extends the capabilities of TensorFlow with additional machine learning (ML) and deep learning (DL) programming features, and support for convolutional, recurrent, and standard neural networks. With an active community and a dedicated Slack channel, getting support is easy[31].

c) Pandas: As an indispensable component of the data science lifecycle, Pandas stands out as the most widely used Python library for data science, alongside NumPy and Matplotlib. Pandas plays a crucial role in data analysis and cleaning. Due to its efficiency, Pandas provides fast and flexible data structures, especially the data frame, which is designed for seamless interaction with structured data. Pandas operates at a high level of abstraction and includes advanced data structures and manipulation tools. Its applications extend to general data processing and cleaning, and ETL (Extract, Transform, Load) tasks for data transformation and storage, with strong support for loading CSV files into their data frame format. Pandas is widely used in academic and business fields, including statistics, finance, and neuroscience, and provides specialized functionality for time series analysis, featuring capabilities such as time domain generation, moving windows, linear regression, and date transformation[32].

d)Matplotlib: Matplotlib stands out as a powerful visualization tool, being powerful and elegant in its graphical representations. Functioning as a Python layout library, it has garnered nearly 26,000 comments on GitHub and maintains an active community of about 700 contributors. Popular for its role in data visualization, Matplotlib provides an object-oriented API, enabling seamless embedding of charts into applications. Matplotlib's applications extend to critical tasks such as analyzing correlations between variables, visualizing 95 percent confidence intervals of models, outlier detection through scatter plots, and insightful visualization of data distribution. This multi-faceted tool has proven effective in gaining instant insights from complex data sets[33].

e) The sklearn.metrics module in the Scikit-learn library serves as a powerful tool for evaluating machine learning models, catering to classification and regression tasks providing an accurate understanding of model performance and guiding informed optimization decisions.Frequently used metrics in sklearn.metrics include[34-38]:

1. Accuracy: represents the percentage of correct predictions. For example, if a model predicts financial fraud and is 90% accurate, it correctly identifies the case in 90 out of 100 financial transactions.

$$\text{acuraccy} = \frac{\text{TP+PN}}{\text{TP+PN+FN+FP}} \qquad 1$$

2. Precision: refers to the percentage of positive predictions that were accurate. For example, if the model predicts ad clicks with 80% accuracy, the user actually clicks on 80 out of the 100 predicted ad clicks.

$$precision = \frac{TP}{TP+FP} \qquad 2$$

3. Recall: refers to the fraction of actual positives that are correctly predicted. For example, if a model predicts cancer 70%, it correctly identifies the condition in 70 out of 100 patients who actually have cancer.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \qquad 3$$

4. F1 score: represents the harmonic average of precision and recall, providing a balanced measure of model performance.

$$\text{F1} - \text{score} = \frac{2*\text{Precision}*\text{Recall}}{\text{Precision+Recall}} \qquad 4$$

In addition to these basic metrics, sklearn.metrics provides specialized metrics such as the confusion matrix, which is a tabular representation of the true and predicted labels, which helps in visualizing the performance of the classification model. The confusion matrix for a binary classification model is shown in Figure 2.

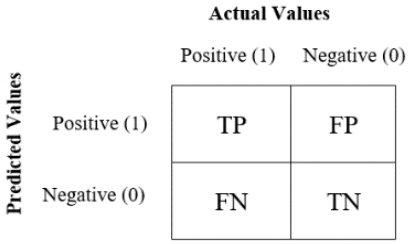

Fig.2: The confusion matrix.

B. Dataset

The data collection stage is one of the most important steps in2 the proposed system. Dealing with credit card fraud detection faces a major scientific challenge due to the limited availability of real data for exploration, which stems primarily from confidentiality concerns [8].In the scenario of financial fraud detection and user behavior analysis, a dataset called "Card Fraud Detection" has been identified. Credit" and can be accessed on the Kaggle platform. This dataset consists of 284,807 credit card transactions, of which 492 were classified as fraudulent. It includes a total of 31 features and has a size of approximately 69 MB. The following table (Table 2) shows a sample of the dataset, explaining its dimensions and specific features. Another critical issue is data imbalance, with fraudulent transactions globally accounting for less than 0.05% of total transactions, creating highly imbalanced categories in credit card fraud datasets [18]:Each row within the table 2 corresponds to an individual credit card transaction, with the attributes showcasing pertinent information such as transaction time, PCA-transformed attributes (V1 to V28), transaction amount, and the class label indicating the legitimacy of the transaction.

Table.2: Sample of the dataset

Time	V1	V2	V3	V4	...	V26	V27	V28	Amount	Class
0.0	-1.35981	-0.07278	2.53635	1.378155	...	-0.18912	0.133558	-0.02105	149.62	0
0.0	1.191857	0.266151	0.16648	0.448154	...	0.125895	-0.00898	0.014724	2.69	0
1.0	-1.35835	-1.34016	1.77321	0.37978	...	-0.1391	-0.05535	-0.05975	378.66	0
1.0	-0.96627	-0.18523	1.79299	-0.86329	...	-0.22193	0.062723	0.061458	123.5	0
2.0	-1.15823	0.877737	1.54872	0.403034	...	0.502292	0.219422	0.215153	69.99	0

C. **Preprocessing**

To conduct experimental tests of the proposed model, the "credit card fraud detection" data set was chosen. It is worth noting that this set is characterized by its unbalanced nature. Figure 3 shows the imbalance in the Credit Card Fraud Detection dataset. In the pre-processing step, multiple techniques are adopted and tested on the data set, and their effectiveness is demonstrated in creating a balanced data set that contributes to tracking user behavior to detect financial fraud in credit cards. There are a set of techniques used to make the "Credit Card Fraud Detection" data set balanced in a way that makes the model's prediction process more accurate. We will not use smooth, oversample and undersample techniques. It will be compared with the adversarial algorithm GAN, and its importance in addressing the problem of imbalance in the "credit card fraud detection" data set will be demonstrated.

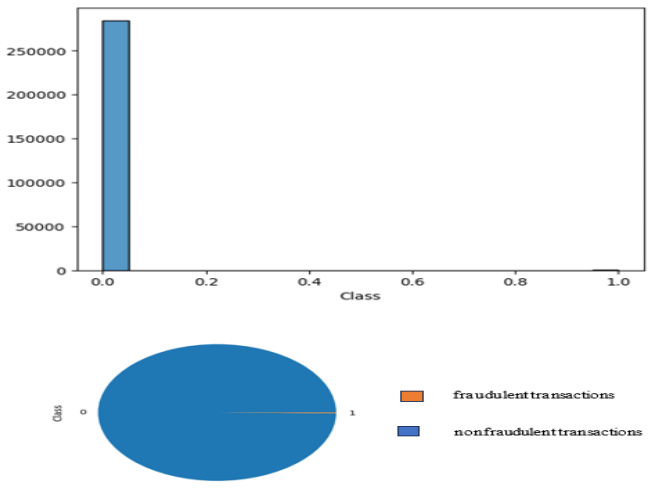

Fig 3: distribution of classes in unbalanced dataset.

1- Smoth technique

Synthetic minority oversampling, known as SMOTE, is a preprocessing technique used to adjust the imbalance in the distribution of classes within a data set. It solves this problem by oversampling the minority class, to avoid overfitting resulting from learning the model from the same Examples. The SMOTE technique examines the difference between the sample and its nearest neighbor, and then multiplies this difference by a random number between 0 and 1. This difference is added to the original sample to create a new synthetic sample in the feature space. This process is repeated with the next nearest neighbor until the number pre-specified by the user is reached. This helps create an effective balance in the data set, enhancing the model's ability to handle categories fairly and accurately. Figure 4 shows Smoth technique on “Card Fraud Detection”.[19]

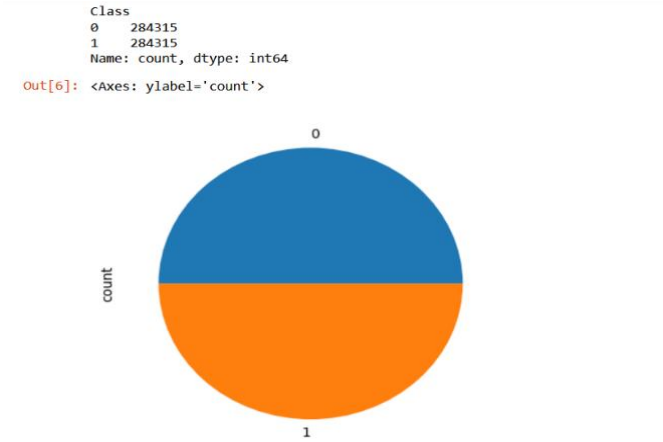

Figure 4: SMOTE technique on dataset

2. Under sampling technique

Under sampling as shown in figure 5 is one of the common techniques to address class imbalance in a data set. The data set is modified by reducing the number of cases in the majority class to balance them with the minority class. Which provides a more equal representation of both categories, making the data set more balanced. Advantages of under sampling include simplicity and computational efficiency. However, under sampling may result in missing potentially valuable information from the majority population.[20]

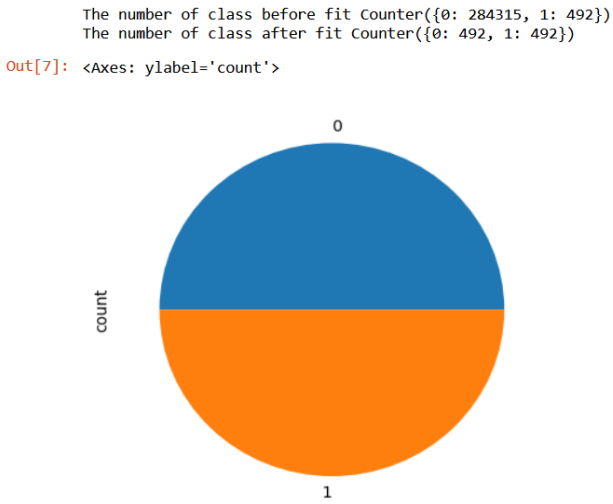

Fig. 5: undersampling technique

3- Over sampling technique

Oversampling is a technique used to address the problem of imbalanced data sets, where one class contains significantly fewer instances than other classes. This technique balances the class distribution by increasing the number of samples in the minority class through the frequency of its examples. While this technique does not provide new information, it does increase the weight of the minority group. However, the disadvantage of oversampling is its ability to exacerbate overfitting, making the model overly specific to the training data.

Thus, although accuracy on the training set may be high, model performance on new datasets may be compromised by overfitting and subsequent failure of the model to train. Figure 6 shows the balancing of Credit Card Fraud Detection using the oversampling technique.[21]

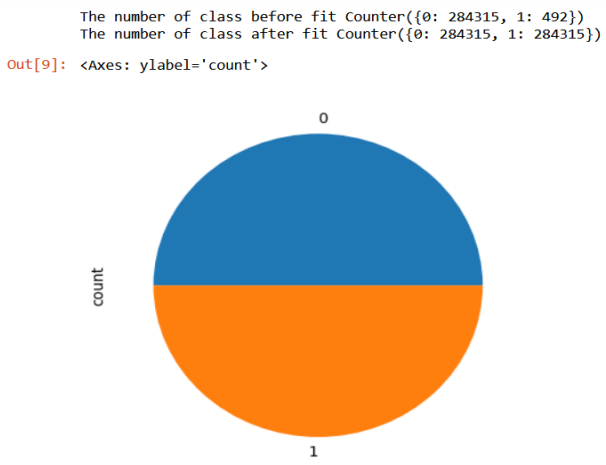

Figure 6: Over sampling technique

4- GAN algorithm

In addition to these common methods, this research presents a new method for aligning the data set. The proposed method is based on the use of the GAN algorithm to balance the Credit Card Fraud Detection data set by determining the size of the samples that are added in order to balance the data set. This technique works by allowing the researcher to determine the final size of the added samples. It is then collected with the basic data and forms a new, balanced data set. Figure 7 shows the process of balancing the data set using GAN technology.[22]

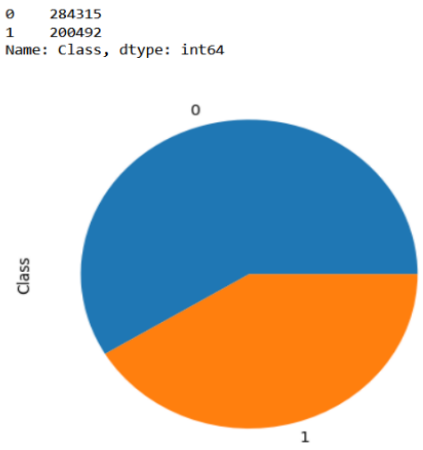

Figure 7: GAN technique

Figure 7 shows the GAN algorithm technique in balancing the data set to detect credit card fraud. It gives flexibility in determining the number of samples to be added for the purpose of balancing the data set.

Also, handling missing values is one of the techniques that is performed in the pre-processing stage, which is shown in Figure 9 Missing data is defined as values or information that are not stored (or do not exist) for some variable in the given data set[40]. Below is a sample of missing data in this group. In addition, measuring the correlation ratio between the features in Figure 8. Figures 8 and 9 show that there are no missing values and there is no correlation between the features.

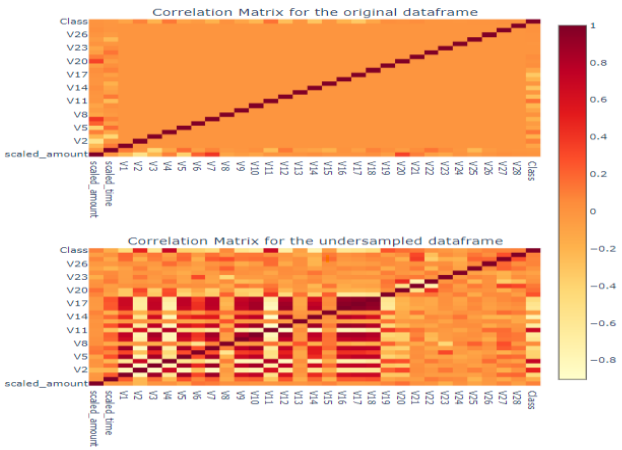

Figure 8: Correlation is a statistical measure[39]

Correlation is a statistical measure that describes the strength and direction of a relationship between two variables. The correlation coefficient, denoted by r, ranges from -1 to 1, where -1 indicates a strong negative correlation, 0 indicates no correlation, and 1 indicates a strong positive correlation. A correlation coefficient of 1 means that the variables are perfectly positively correlated, while a coefficient of -1 means that they are perfectly negatively correlated. A correlation coefficient of 0 means that there is no correlation between the two variables. The correlation coefficient is used to describe the relationship between two variables and can be used in feature selection and other machine learning tasks[39].

```
Out[10]: Time      0
         V1        0
         V2        0
         V3        0
         V4        0
         V5        0
         V6        0
         V7        0
         V8        0
         V9        0
         V10       0
         V11       0
         V12       0
         V13       0
         V14       0
         V15       0
         V16       0
         V17       0
         V18       0
         V19       0
         V20       0
         V21       0
```

Figure 9: missing value technique [40].

D. Splitting dataset

To ensure unbiased evaluation and validation of the machine learning model. The `train_test_split' function is used to split your dataset into subsets, reducing potential biases during the evaluation process. This requires separating the evaluation dataset from the training data, ensuring that the model is evaluated using new, unseen data. This separation is achieved through the strategic use of partitioning the data set before evaluating the model. The ratio of 80% to 20% was adopted when using the train_test_split function.

E. Machine learning

Machine learning, as a fundamental part of the field of artificial intelligence, relies on the use of algorithms to leverage data and make intelligent decisions. This field includes supervised learning where models are trained using labeled data to make predictions, unsupervised learning which discovers patterns in unlabeled data, and reinforcement learning which involves interactions between the agent and the environment to make optimal decisions. Supervised learning includes techniques such as support vector machines, decision trees, and logistic regression, while regression predicts continuous outcomes. This framework offers powerful tools for data analysis, forecasting, and decision-making in a variety of fields. Figure 10 shows different types of machine learning. [23].

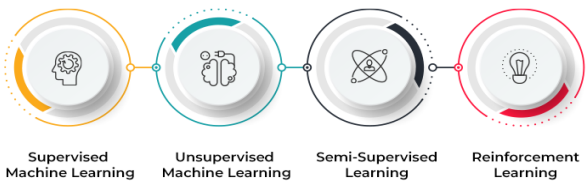

Figure 10: Type of Machine learning [23].

To perform the task of financial fraud detection and user behavior analysis, four machine learning algorithms have been selected, as elaborated in the following section.

1. Decision Tree

Decision tree is one of the well-known supervised learning techniques used to address classification tasks. Creating this type of model involves iteratively dividing data into subsets based on input attributes, and creating a leaf node that represents the expected output. The feature to be used in the partitioning is chosen based on the calculation of the maximum information gain or information gain ratio, which is calculated using target variable entropy measurements before and after each partitioning operation. The Gini measure of impurity is also used for misclassifications.

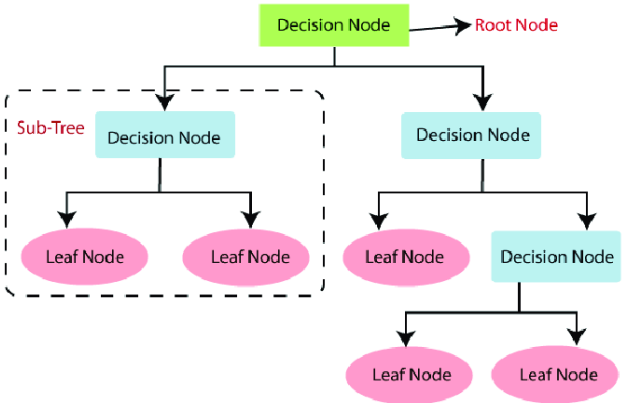

Fig 11: Decision tree algorithm [37]

Entropy, as a measure of randomness, is a guide for decision tree algorithms to evaluate the extent of information inclusion. The entropy calculation includes parts of events at child nodes and aims to maintain simplicity by constructing a tree with uniform branches at each level. The decision tree structure is clearly shown in Figure 11. [24].

2. Naive Bayes

The Naive Bayes algorithm is a probabilistic classification approach that applies Bayes' theorem while assuming attribute independence. It calculates the likelihood of a class having specific attributes and determines the optimal class for a given input. Each attribute is treated as independent and equally important.

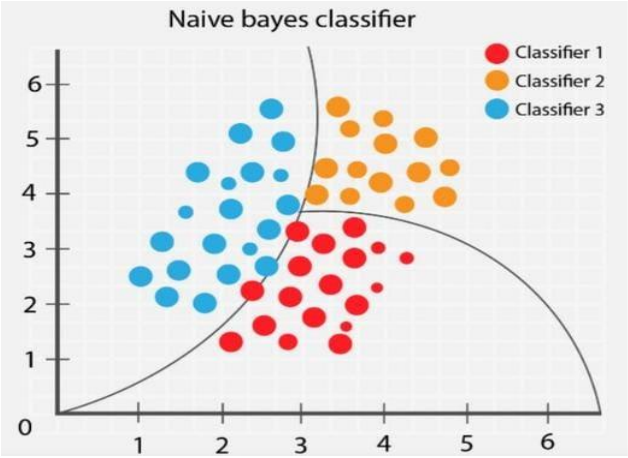

Fig 12: Naive Bayes algorithm.[25]

The Bayes formula serves as the foundation for this method, where P(H|X) represents the probability of hypothesis H being true given the data X, and P(X|H) denotes the likelihood of observing evidence X given hypothesis H. The algorithm calculates the probability that input belongs to a class, assuming no correlation between attributes. It evaluates the likelihood of each category and returns the one with the highest likelihood. An example study using the Naive Bayes algorithm analyzes sentiment in tweets about Indonesian presidential candidates, showing its effectiveness for categorizing data into predefined classes. The behavior of the Naive Bayes algorithm is illustrated in Figure 12.[25].

3. Logistic regression (LR)

Logistic regression is a statistical method used to analyze the relationship between a dependent variable and one or more independent variables. It aims to find coefficients that minimize the difference between predicted and actual values.

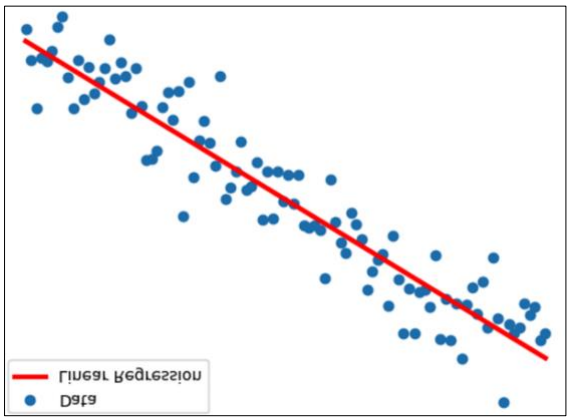

Figure 13 Logistic regression algorithm [26].

For simple linear regression ($y = \beta0 + \beta1x + e$), and multiple linear regression ($y = \beta0 + \beta1x1 + \beta2x2 + ... + e$), it predicts dependent variables based on independent variables. It's valuable for understanding correlations, risk factors, and predicting outcomes.[25] The structure is shown in Figure 13.

F. Ensemble Methods in Machine Learning

Ensemble methods in machine learning are techniques that combine multiple individual models to improve the accuracy and robustness of overall prediction. The idea behind ensemble methods is that by combining multiple models, we can overcome the limitations of any single model and improve the overall performance of the system. Figure 14 shows the group methods[27].

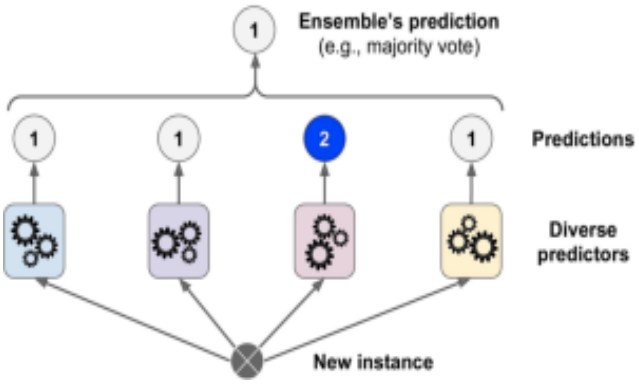

Fig 14 : the ensemble methods [27]

The research presents a hybrid machine learning model using the ensemble technology for machine learning algorithms presented in paragraph C. Conducting tests on the credit card fraud detection data set after addressing the problem of imbalance in the categories of the data set using the four methods mentioned, and conducting a comparison between them to conclude the best path in terms of prediction accuracy.

4. METHODOLOGY

Figure 15 clearly shows the proposed approach.

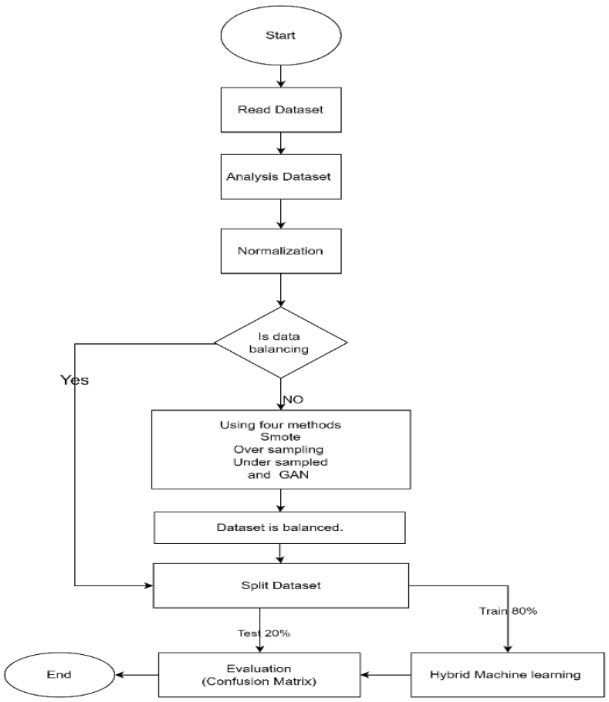

Fig15: Block diagram of main stages for proposed model

This section presents the proposed method for building an artificial intelligence model based on user behavior tracking to detect financial fraud in credit cards. The research focuses on testing the GAN technique to normalize the dataset and comparing it to other methods used.The proposed hybrid automated model consists of a set of four algorithms, namely Decision Tree, Naive Bayes, Logistic Regression, and Support Vector Machines (SVMs). Its performance is evaluated using metrics of precision, recall, precision, and F1-score for a comprehensive evaluation.

5. Experimental Results

In this section, we present the results of the hybrid machine learning implementation used in the proposed system. The proposed hybrid model is tested on the data set after processing it using the four methods for balancing the categories within it. In addition, we apply classification metrics based on confusion matrix distributions.

1. Smote method

Table 3: Performance of hybrid machine

details	Results%
train_score	98.3
test_score	98
Accuracy	98.1
Precision	99.7
Recall	98.1
f1_score	98.1

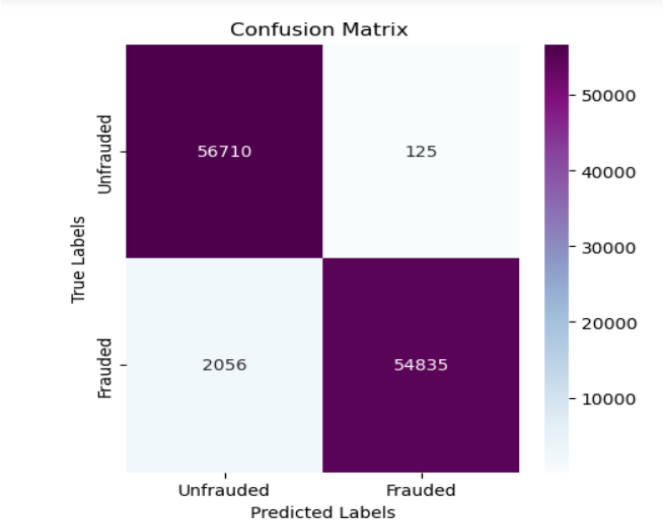

Fig.16: The confusion matrix of hybrid Ml with smote.

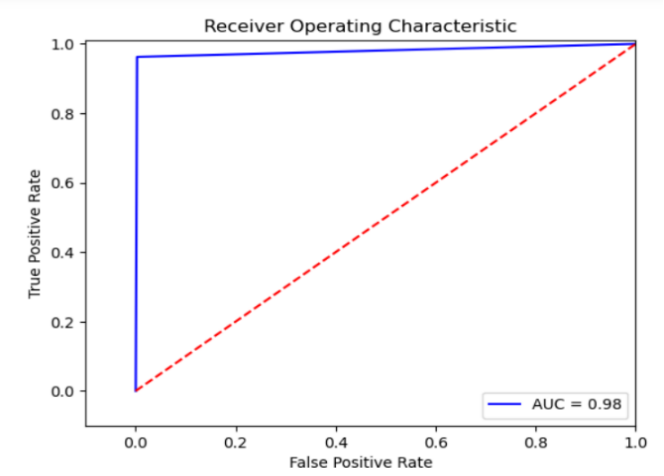

Fig. 17: The ROC curve of hybrid Ml with smote.

2. Under sampling method

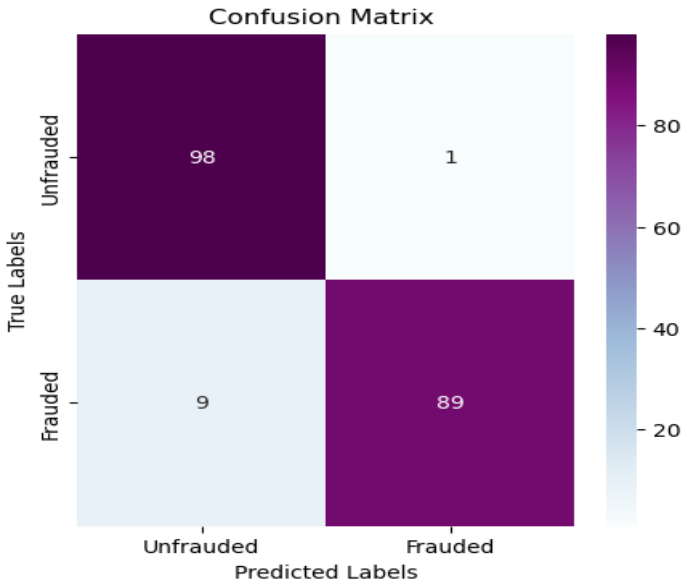

Fig. 18: The confusion matrix of hybrid Ml with under sampling method.

Table 4: Performance of hybrid machine

details	Results%
train_score	97.2
test_score	94.3
Accuracy	94.9
Precision	98.8
Recall	94.9
fl_score	94.7

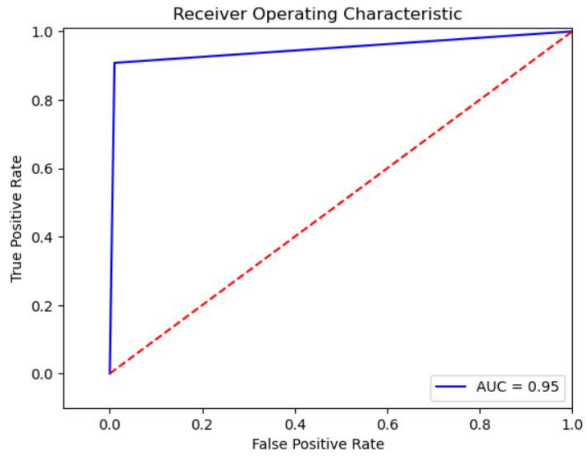

Fig. 19: The ROC curve of hybrid Ml with under sampling method

3. Over sampling method

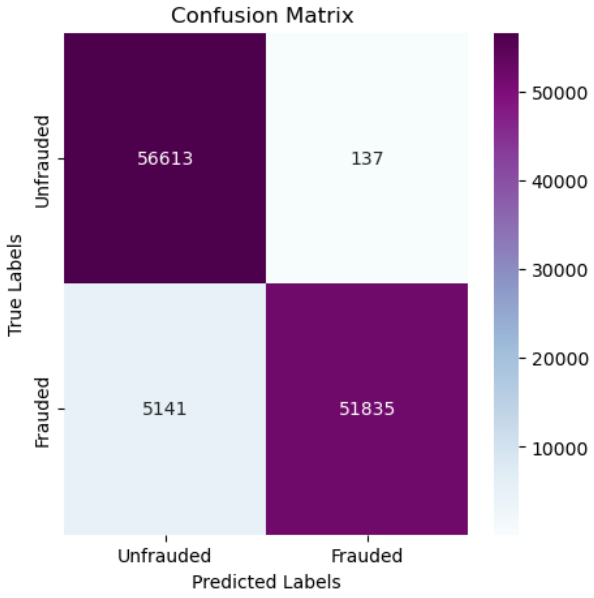

Fig. 20: The confusion matrix of hybrid Ml with Over sampling method

Table 5: Performance of hybrid machine

details	Results%
train_score	95.5
test_score	95.4
Accuracy	95.3
Precision	99.7
Recall	95.2
f1_score	95.1

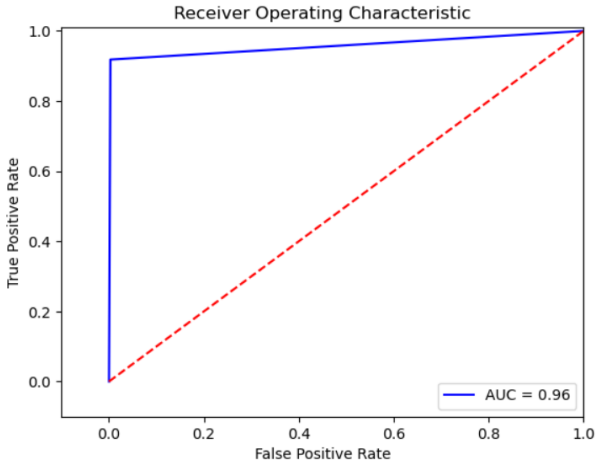

Fig. 21: The ROC curve of hybrid Ml with over sampling method

1. GAN method

Table 5: Performance of hybrid machine

details	Results%
train_score	99.9
test_score	99.9
Accuracy	99.9
Precision	99.9
Recall	99.9
f1_score	99.9

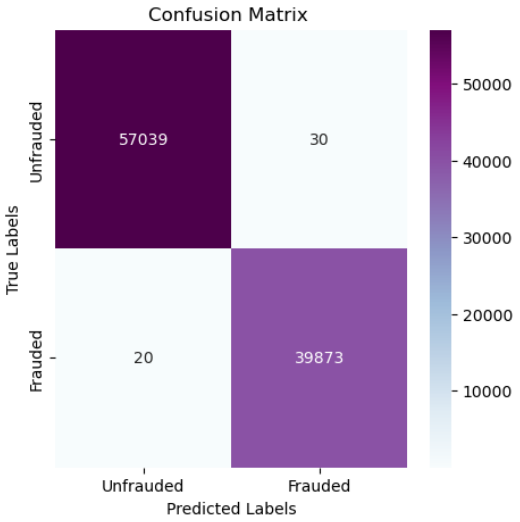

Fig. 22: The confusion matrix of hybrid Ml with under sampling method.

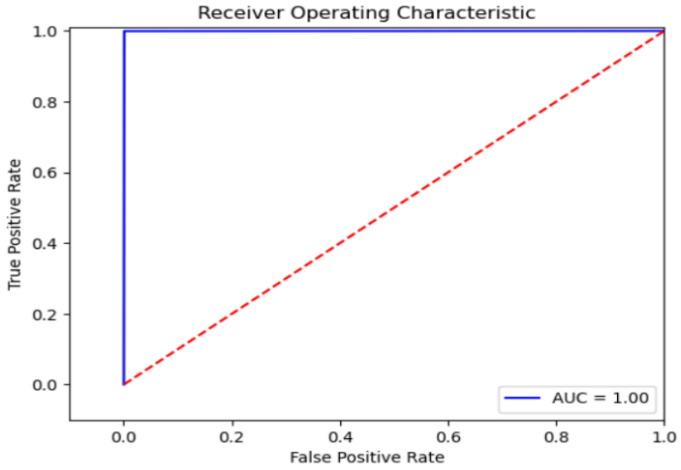

Fig. 23: The ROC curve of hybrid Ml with over sampling method

The execution time of the hybrid machine learning model was recorded in the tests above, as Figure 20 shows that the longest time to train the model when processing the data using the Smote method, then testing when the data was processed using the GAN method, and after that, testing when the data was processed using the Over Sample method, and finally, testing. When processing data using the under sample method.

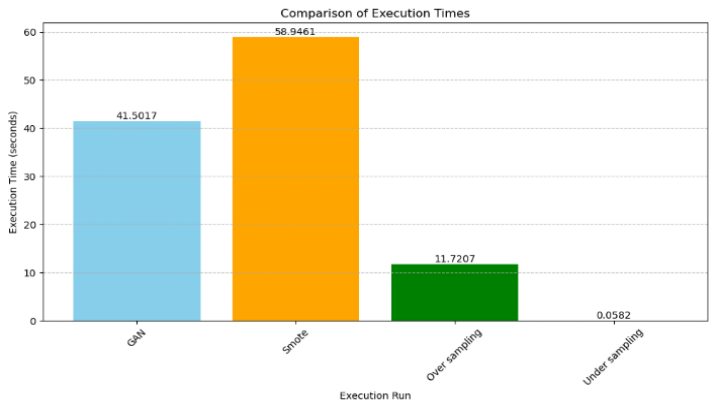

Fig 24: The execution time of hybrid Ml.

6. DISCUSSION

The performance of the hybrid machine learning model was evaluated using different methods to solve the imbalance problem in the credit card financial fraud detection dataset categories, such as SMOTE, Under Sampling, Over Sampling, and GAN. The four tests of the model showed mixed results. Some of them reflect the effectiveness of the model in different aspects. Model testing using the SMOTE method recorded an accuracy of 98.1% and an F-score of 98.3%. On the other hand, the over-sampling and under-sampling testing methods showed close performance, as the two methods recorded an accuracy of 94.3 and 95.3 and an F-score of 94.7 and 95.1, respectively. Finally, the GAN method excelled, showing a test score and accuracy of 99.9%, in addition to exceptional accuracy, recall, and F1 score.also , When examining the receiver operating characteristic (ROC) curve to predict financial fraud, the results also showed the superiority of the GAN algorithm, achieving an accuracy of 100, while the Smot technique achieved an accuracy of 96%, and the under sample and over sample achieved an accuracy of 95 and 95.5, respectively.As a result, we conclude the ability of the GAN method to balance the data set, which in turn is reflected in the model's performance in training and the accuracy of predictions when tested.

7. CONCLUSION

In this research, we apply a hybrid machine learning approach to a credit card fraud detection dataset to evaluate its effectiveness in detecting financial fraud by tracking user behavior. Our focus was on comparing tests of the proposed model on the dataset after balancing it with different methods, namely SMOTE, Under Sampling, and Over Sampling. GAN. The GAN achieved the highest test score and accuracy of 99.9%, as well as exceptional precision, recall, and F1 score.

REFERENCES

1 Al-Faqir, S., & Ouda, O. (2022). Credit Card Frauds Scoring Model Based on Deep Learning Ensemble. Journal of Theoretical and Applied Information Technology, 31st July 2022, Vol.100, No 14, 5223-5224. Little Lion Scientific. ISSN: 1992-8645.

2 Al-Shabi, Mohammed. (2019). Credit Card Fraud Detection Using Autoencoder Model in Unbalanced Datasets. Journal of Advances in Mathematics and Computer Science. 1-16. 10.9734/jamcs/2019/v33i530192.

3 Hassan.N, Ola.A, Ayah.A.A and Mutaz.Y, "Credit Card Fraud Detection Based on Machine and Deep Learning," 2020 11th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 2020, pp. 204-208, doi: 10.1109/ICICS49469.2020.239524.

4 Puh.M and Ljiljana B. "Detecting Credit Card Fraud Using Selected Machine Learning Algorithms." 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (2019): 1250-1255.

5 S P, Maniraj & Saini, Aditya & Ahmed, Shadab & Sarkar, Swarna. (2019). Credit Card Fraud Detection using Machine Learning and Data Science. International Journal of Engineering Research and. 08. 10.17577/IJERTV8IS090031.

6 Tiwari, Pooja & Mehta, Simran & Sakhuja, Nishtha & Kumar, Jitendra & Singh, Ashutosh. (2021). Credit Card Fraud Detection using Machine Learning: A Study.

7 Mijwil, M. M., & Salem, I. E. (2020). Credit Card Fraud Detection in Payment Using Machine Learning Classifiers. Asian Journal of Computer and Information Systems, 8(4), 50. Asian Online Journals. Retrieved from http://www.ajouronline.com.

8 Ileberi, E., Sun, Y. & Wang, Z. A machine learning based credit card fraud detection using the GA algorithm for feature selection. J Big Data 9, 24 (2022). https://doi.org/10.1186/s40537-022-00573-8.

9 Afriyie, Jonathan & Tawiah, Kassim & Pels, Wilhelmina & Addai-Henne, Sandra & Dwamena, Harriet & Emmanuel, Odame & Ayeh, Samuel & Eshun, John. (2023). A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions. Decision Analytics Journal. 6. 100163. 10.1016/j.dajour.2023.100163.

10 Asma Cherif, Arwa Badhib, Heyfa Ammar, Suhair Alshehri, Manal Kalkatawi, Abdessamad Imine,Credit card fraud detection in the era of disruptive technologies: A systematic review,Journal of King Saud University - Computer and Information Sciences,Volume 35, Issue 1,2023,Pages 145-174,ISSN 1319-1578,https://doi.org/10.1016/j.jksuci.2022.11.008.

11 R. Asha, S. K.-G. T. Proceedings, and undefined 2021, "Credit card fraud detection using artificial neural network," Elsevier, Accessed: Mar. 11, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666285X21000066.

12 Rtayli, Naoufal and Nourddine Enneya. "Enhanced credit card fraud detection based on SVM-recursive feature elimination and hyper-parameters optimization." *J. Inf. Secur. Appl.* 55 (2020): 102596.

13 Chang, Victor & Doan, Le Minh Thao & Di Stefano, Alessandro & Sun, Zhili & Fortino, Giancarlo. (2022). Digital payment fraud detection methods in digital ages and Industry 4.0. Computers & Electrical Engineering. 100. 107734. 10.1016/j.compeleceng.2022.107734.

14 Sadineni, Praveen Kumar. (2020). Detection of Fraudulent Transactions in Credit Card using Machine Learning Algorithms. 659-660. 10.1109/I-SMAC49090.2020.9243545.

15 Bryan Ka, Nathaniel P, Stephen R, Dewi S, Julian W, Yudy P,On the benefits of machine learning classification in cashback fraud detection,Procedia Computer Science,Volume 216,2023,Pages 364-369,ISSN 1877-0509,https://doi.org/10.1016/j.procs.2022.12.147.

16 Błaszczyński, J., de Almeida Filho, A. T., Matuszyk, A., Szeląg, M., & Słowiński, R. (2021). "Auto loan fraud detection using dominance-based rough set approach versus machine learning methods." Expert Systems with Applications, 163, 113740.
17 S. Bagga, A. Goyal, N. Gupta, and A. Goyal, "Credit card fraud detection using pipeling and ensemble learning," Procedia Comput. Sci., vol. 173, pp. 104–112, 2020.
18 M. M. Mijwil and I. E. Salem, "Credit Card Fraud Detection in Payment Using Machine Learning Classifiers," Asian J. Comput. Inf. Syst., vol. 8, no. 4, 2020.
19 S. Lakshmi and S. D. Kavilla, "Machine learning for credit card fraud detection system," Int. J. Appl. Eng. Res., vol. 13, no. 24, pp. 16819–16824, 2018.
20 P. Gupta, A. Varshney, M. R. Khan, R. Ahmed, M. Shuaib, and S. Alam, "Unbalanced Credit Card Fraud Detection Data: A Machine Learning-Oriented Comparative Study of Balancing Techniques," Procedia Comput. Sci., vol. 218, pp. 2575–2584, 2023.
21 A. Cherif, A. Badhib, H. Ammar, S. Alshehri, M. Kalkatawi, and A. Imine, "Credit card fraud detection in the era of disruptive technologies: A systematic review," J. King Saud Univ. Inf. Sci., 2022.
22 D. Kajal and K. Kaur, "Credit card fraud detection using imbalance resampling method with feature selection," Int. J., vol. 10, no. 3, 2021.
23 E. Ileberi, Y. Sun, and Z. Wang, "A machine learning based credit card fraud detection using the GA algorithm for feature selection," J. Big Data, vol. 9, no. 1, pp. 1–17, 2022..
24 J. Błaszczyński, A. T. de Almeida Filho, A. Matuszyk, M. Szeląg, and R. Słowiński, "Auto loan fraud detection using dominance-based rough set approach versus machine learning methods," Expert Syst. Appl., vol. 163, p. 113740, 2021.
25 E. N. Osegi and E. F. Jumbo, "Comparative analysis of credit card fraud detection in Simulated Annealing trained Artificial Neural Network and Hierarchical Temporal Memory," Mach. Learn. with Appl., vol. 6, p. 100080, 2021.
26 A. Farhang Ghahfarokhi, T. Mansouri, M. R. Sadeghi Moghaddam, N. Bahrambeik, R. Yavari, and M. Fani Sani, "Credit card fraud detection using asexual reproduction optimization," Kybernetes, vol. 51, no. 9, pp. 2852–2876, 2022.
27 M. Rabbani et al., "A review on machine learning approaches for network malicious behavior detection in emerging technologies," Entropy, vol. 23, no. 5, p. 529, 2021.
28 C. Sudha and D. Akila, "WITHDRAWN: Majority vote ensemble classifier for accurate detection of credit card frauds." Elsevier, 2021.
29 Akshansh Sharma, Firoj Khan, Deepak Sharma, Dr. Sunil Gupta, "Python: The Programming Language of Future"INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN TECHNOLOGY, May 2020 | IJIRT | Volume 6 Issue 12 | ISSN: 2349-6002.
30 alolov, T. S. (2023). PYTHON INSTRUMENTLARI BILAN KATTA MA'LUMOTLARNI QAYTA ISHLASH. Educational Research in Universal Sciences, 2(10), 320-322.
31 Harris, C.R., Millman, K.J., van der Walt, S.J. et al. Array programming with NumPy. Nature 585, 357–362 (2020). https://doi.org/10.1038/s41586-020-2649-2
32 Dürr, O., Sick, B., & Murina, E. (2020). Probabilistic Deep Learning: With Python, Keras, and TensorFlow Probability. Retrieved from Google Books: https://books.google.com/
33 Saabith, A. L. S., Vinothraj, T., & Fareez, M. M. M. (2020). Popular Python libraries and their application domains. International Journal of Advance Engineering and Research Development, 7(11), 18.
34 Sial, A. H., Rashdi, S. Y. S., & Khan, A. H. (2021). Comparative analysis of data visualization libraries Matplotlib and Seaborn in Python. International Journal of Advanced Trends in Computer Science and Engineering, 10(1), Retrieved from http://www.warse.org/IJATCSE/static/pdf/file/ijatcse391012021.pdf.
35 Qin, Jian, et al. "Research and application of machinelearning for additive manufacturing." AdditiveManufacturing (2022): 102691.
36 Scikit-learn. (2023). scikit-learn: Machine Learning in Python. Retrieved from https://scikit-learn.org/stable/index.html.
37 Albadr, Musatafa Abbas Abbood, et al. "Speech emotion recognition using optimized genetic algorithm-extreme learning machine." Multimedia Tools and Applications 81.17 (2022): 23963-23989.
38 Albadr, M. A. A., Ayob, M., Tiun, S., AL-Dhief, F. T., Arram, A., & Khalaf, S. (2023). Breast cancer diagnosis using the fast learning network algorithm. Frontiers in Oncology, 13, 1150840.
39 Albadr, Musatafa Abbas Abbood, et al. "Gray wolf optimization-extreme learning machine approach for diabetic retinopathy detection." Frontiers in Public Health 10 (2022): 925901.
40 Jaadi, Z. (2019, October 15). Everything you need to know about interpreting correlations. Towards Data Science.
41 Tamboli, N. (2023, July 14). Effective Strategies for Handling Missing Values in Data Analysis (Updated 2023).