Al-Rafidain Journal of Computer Sciences and Mathematics (RJCSM)

www.csmj.uomosul.edu.iq

Credit Card Fraud Detection Using Feature Select Method and Improved Machine Learning Algorithm

Mohammed Mansooor Mohammed AL-Hammadi

Institute of Applied Arts, Middle Technical University, Baghdad, Iraq
Email: mohamed_mansour@mtu.edu.iq

Article information

Article history:
Received 24 February ,2025
Revised 24 April ,2025
Accepted 09 May ,2025
Published 26 June ,2025

Keywords:
credit card fraud,
support vector machine,
particle swarm optimization,
hyperparameters

Correspondence:
Mohammed Mansooor
Mohammed AL-Hammadi
Email:
mohamed_mansour@mtu.edu.iq

Abstract

The digital modern era has established credit card fraud as an important security problem which introduces financial vulnerabilities for every member of society including financial institutions and business operations. Financial systems need the detection of such fraud for proper security maintenance along with minimal risk reduction. The research uses an enhanced support vector machine (SVM)-based method that incorporates advanced feature selection techniques for detecting fraudulent transactions. A binary genetic algorithm working with cross-entropy finds the most relevant features by assessing the connection between variables and the target variable. The SVM model performs the classification task following optimization of its hyperparameters through the application of particle swarm optimization (PSO). Experimental trials executed against the Credit Card Fraud Detection dataset proven the proposed method's effectiveness because it delivered 99.99% accuracy. Through the integration of optimization techniques into feature selection algorithms this method improves both security efficiency while maintaining high accuracy rates for credit card fraud detection in contemporary financial settings.

DOI: 10.33899/csmj.2025.157705.1175,

1. Introduction

As the use of e-transactions become common all over the world, so does credit card fraud causing significant concern to the globe. Internet financial operations are getting more significant and thus fraud threats increasing; fraud threats cost tens of billions annually, and there is a requirement for effective fraud identification. The proceeding of using the given data or the information of the card in an unlawful or unauthorised manner is credit card fraud. They employ methods such as identity theft, card skimming, phishing, data breach to acquire the said financial details. Beyond financial losses for individuals and companies, these fraudulent activities erode trust in the entire payment ecosystem. With this rising threat, financial institutions, payment processors and, tech firms have developed elaborate systems to deter fraud. These systems embrace most of the modern technologies such as artificial intelligence-AI, machine learning-ML, and data analytics tools coupled with an ability to detect anomalies in a given pattern, thus helping to prevent fraudulent transactions and enhance the security of financial networks.

1.2. Research background

In study [1], authors have adopted the machine learning techniques on a balanced credit card dataset to assess the performance of the techniques on the imbalanced environment. Through the choice of the Kaggle's balanced dataset containing 568630 transactions Random Forest, Neural Networks, Logistic Regression, and Naive Bayes were compared. The performance yields proved that using Random Forest as well as Neural Networks yields a very high accuracy of 95.9% to detect fraud which can greatly complement the current security of the financial institutions. Such findings indicate the great potential of these advanced methods in improving the detection of the fraud in the financial institutions and offer a good reference for further enhancement of the fraud detection systems. Reference [2] describes a method that utilizes other sophisticated technologies of machine learning for

increasing the rates of deception identification in the banking industry. This study thus highlights the necessity of feature selection in enhancing the detection capacity because of exclusion of the extra features. Considering the provided results, the Brown-Bear Optimization (BBO) algorithm provides the array of features, which is critical for accurate fraud detection. Compared to other methods, this method found that it is 91% accurate for classification on the Australian credit dataset and provided a better way to identify credit cards fraud and manage the problems of dimensionality. In [3], the authors proposed a new approach adopting the SMOTE-KMEANS algorithm with an ensemble in the field of credit card fraud detection. The performance of the proposed model was thus compared with expected models such as logistic regression, decision tree, random forest, support vector machines. The evaluation of the performance was based on measures such as accuracy, the recall rate and the curve of Area Under Curve (AUC). Their results further revealed that the proposed model had better performance and AUC 0.96 with SMOTE-KMEANS, signifying that its effectiveness in identifying fraudulent transactions while ensuring precision and recall values were comparatively higher. Other assessments undertaken encompassed the evaluation of different oversampling techniques to enhance the efficiency of different classifiers. These results indicate that the proposed method has higher accuracy and robustness when following the condition of having a balanced train and test set in classification tasks. The subsequent studies will be targeted at the improvement of the algorithm and using the obtained SMOTE-KMEANS method as an addition to the existing fraud detection models for increasing the action's protection of the financial and consumer spheres. Another study in Reference [4] took a look at the experience of the financial institutions in cases of credit card fraud and more so the number of majorities between the usual and actual fraudulent transaction. This has been the case; making it hard to detect fraud. In this study, different types of sampling techniques have been discussed in order to discover the effectiveness of the sample model in recognizing fraudulent behaviors. It puts into consideration two individual techniques, the Random Undersampling (RUS) and Synthetic Minority Oversampling Technique (SMOTE) out of which the best approach to increase the detection accuracy is discerned and the other three hybrid techniques they are, RUS + Random Oversampling (ROS), RUS + SMOTE, and the last one is RUS + SMOTE Tomek. Using the same data set, six models were used which were namely Random Forest, Logistic Regression, XG Boost and AdaBoost, LightGBM, and Neural Networks with optimization of the hyperparameters of the model being emphasized. Hence, the comparative analysis revealed that all the hybrid sampling techniques performed well in comparison to the individual techniques with an especial emphasis on those of RUS + SMOTE that had an enhanced performance of making fewer false positive and negative values. The model that proved to be the most efficient in this fraud detection task was LightGBM since it yielded an MCC of 0.85. The study contributes to the knowledge of improving fraud detection systems and recommends areas for development for sampling methods and the models in the future. In reference [5], authors examined Random Forest and the K-Nearest Neighbors (KNN) algorithms to be used for the detection of fraud. Bearing this in mind, the problem of fake actions and its prevention is rather important for different industries. In the paper, the authors described the literature on fraud detection after which they discussed Random Forest and KNN methods. An integrated system was presented which enhances both the models for credit card fraud detection, and the architecture as well as the method of the system was also described. It prepares the data and partitions them, builds the models, and assesses its models on the testing data set. It established that the above formulated proposals are accurate, reliable, and scalable, as evidenced by the analysis of the results obtained. Besides, the research described information regarding the most frequent hours of the week for performing fraudulent operations and occupations linked to such processes. The study also focuses on the efficiency of employing Random Forest and KNN approaches in the context of fraud and provides a useful tool in combating fraudulent activities in the fiscal realm. In the particular, to control the uneven distribution of an SVM-based classifier on the data, the firefly optimization approach is integrated with the model in Paper [6]. The feature selection is performed using the firefly algorithm and the classification is done using the SVM and the accuracy achieved is 85.6% and 591 number of fraudulent case is correctly identified Here, the key issues and challenges are mentioned that exist in the fraud detection process such as the imbalance data, noise in labels and inconsistent representation of transaction during the COVID-19 time period as discussed in Reference [7]. The proposed method effectively defines transactions in terms of extracted time and spatial dimensions and uses the self-supervised method for the analysis of card holder action sequences. On real-world datasets, this approach is combined with high F1 such scores are higher compared to the standard methods. In [8], authors present a new approach to credit card fraud detection with regards to the fact that credit card usage is becoming more widespread when used in internet purchases. I think that the owners of credit cards are lucky to have the possibility to perform purchases without the use of actual cash. Nevertheless, it has turned into a disadvantage because with the help of these convenient tools, scammers are able to gain illicit revenue. This paper focuses on Ensemble Learning those mechanims especially gradient boosting algorithms such as LightGBM & LiteMORT; The paper also discussed about importance of accurate fraud detection in details. It is observed that the model efficiency and error decreases if both the methods are used in Simple and Weighted Averaging manner. Weighted averaging LightGBM & LiteMORT has proven efficient as per the obtained scores of 95.20, 90.65, 91.67, 92.79, 99.44 in AUC, Recall, F1-

score, Precision, Accuracy metrics respectively. The problem of card theft has been on the rise in the last few years and has posed a major threat to most electronic transactions hence requiring more advanced techniques in fraud prevention. Another method described in Reference [9] simultaneously addresses both the problem of characterizing fraud and the presence of data imbalance in the dataset. This approach brings out a model which predicts past transaction record for the identification of the two classes of the transactions in order to reduce misclassification as much as is possible. Since this model focuses on achieving high recall rates and, therefore, a minimal number of false positives, the model provides effective fraud detection with acceptable precision. The framework also improves the feature analysis portion using logistic regression, random forest, and XGBoost with SMOTE technique for higher accuracy and preventive measures against credit card fraud Considering in Reference [10], the study aims at identifying the credit card fraud, a significant threat to the financial sector, by implementing an ML model. Pre-processing of the data is done through feature scaling in order to improve the solution, where standard scaler was used. The case study of the experiment entails four algorithms, namely Logistic Regression, Decision Trees, Random Forests, and Support Vector Machines, and the decision is made based on Accuracy, Precision, Recall, and F1-Score. This allows for finding the advantages and disadvantages of every model and, therefore, comes to the conclusion about what approach should be chosen to work with the problem of fraud detection. This is witnessed by the fact that although some models offer higher accuracy other have better recall which is very important in the case of fraud detection due to minimal False negatives. This is associated with enhancing financial security through developing an algorithm that determines the best method of applying machine learning in credit card fraud detection. Regarding uncertainty and bias issues of DNN models especially in unpredictable fraud occurrences, Reference [11] A growing global problem of credit card fraud due to increasing use of online transactions and financial crimes is solved using machine learning techniques on real dataset of European card holder transactions. In order to address the problem of class imbalance in the datasets proper resampling methods are used. Cross validation is performed and the cost sensitive F-beta score is used for model comparison which takes into consideration real life implications of false positives and false negatives. Besides, the effectiveness of the developed model is analyzed depending on the choice of the amount of imbalanced data and its consequences for machine learning algorithm performance, as well as the advantages of ensemble methods. A Linear Regression along with Random Forest model produces the best F- beta in this study, asserting its reliability and versatility. It stresses on the issues of cost-sensitive evaluation, non-standard data resampling and ensemble learning for constructing the improved approaches in cost-constrained fraud detection. In Reference [12], the authors present a different approach that is based on the selected features and measured from transaction data. This method uses an approach of voting mechanism for combining numerous feature ranking subsets as well as using the thresholding method as well as the supervised select subset. The AUC and AUPRC of the proposed approach of classification models such as XGBoost and CatBoost are also measured and from the results the use of the ensemble-selected features improves the accuracy of fraud detection in unbalanced data set.

The performance evaluation of 66 machine learning models is also reviewed in [13] using a real-world dataset from Europe to specify that AllKNN-CatBoost, is the most effective model, as it reaches AUC 97.94 % and recall 95.91 %. The results derived from this work also show that proper selection of the model and the resampling method can lead to highly improved performance in the field of fraud detection.

In Reference [14], credit card fraud is discussed, which ranks as a critical topic today because the number of credit card-related frauds has increased remarkably high in the past ten years due to the growth of international business, e-commerce, and FinTech. Based on the statistic losses are likely to cross $400 billion in the next decade, and therefore there's need for proper fraud detection mechanisms. This research is a work that applies Machine Learning (ML) and presents an attempt to introduce a new method of credit card fraud detection.

In Reference [15], the author also perceives that the adoption of UPI apps, credit, and debit card usage has led to higher cases of credit card fraud. It is not easy to identify such a fraud because the fraudsters make themselves to pass for normal users. To this end, the study puts forward an automated credit card fraud detection approach that primarily uses ensemble learning which is the combination of a set of supervised machine learning algorithms with the aim of enhancing accurate prediction. One of the major issues which require resolution in fraud detection domain is the shortage of equity of transaction data sets, as the number of cheques or credit card transactions containing fraudulent instances is considerably less than that of the normal ones. In order to address this, the study uses two types of sampling techniques at the data level, which are random oversampling and random undersampling coupled with three base classifiers, Random Forest, Logistic Regression, and K-NN classifier. The combination of these models is done using the voting ensemble technique with weighted votes in order to enhance efficiency and accuracy by minimizing cases of wrong classification of fraudulent transactions. This is a common problem that not only costs a lot of financial damages; it is discussed in Reference [16]. Since credit cards are now one of the most common means of payments in the internet as well as traditional stores and with continuous enhancement of e-Commerce, the rate of fraudulent transactions has been on the rise. The main aim

of this study is to establish a model that can be able to predict credit card fraud. In order to this, several methods of supervised machine learning were applied, namely Neural Networks, Naive Bayes, K Nearest Neighbors (KNN), regression models, and Random Forests.

The dataset applied for training and testing of these models was obtained via the UCI Machine Learning Repository. The implementation was done in the Python language to make it easier to achieve the objectives of the study. From the experiment results, all the models have proved to have high capability in the identification of the fraud, but the best among them is the Random Forest with high accuracy in classification.

As reported in Reference [17], the proposed approach of handling the issue of class imbalance in fraud detection datasets is to use undersampling to reduce fraud instances and oversampling to increase them, along with logistic regression. It shows that the features include the transaction size and origin of the transaction that point towards making fraud observations; the model using a logistic regression has an accuracy level of 94%. Logistic regression thus showed its efficiency in dealing with data imbalanced situations where cases of fraud occurrence were limited for this study. In paper [18], authors proposed the credit card fraud detection using various machine learning and deep learning methods. The training features for normal and abnormal transactions involve the use of Naive Bayes, Logistic Regression, K-Nearest Neighbor (KNN), Random Forest, and the Sequential Convolutional Neural Network. The performance of the model is evaluated on information that is available to the public. The prediction findings show an average of 96.1% for the Naive Bayes Classifier, 94.8% for Logistic Regression, 95.89% for the KNN algorithm, 97.58% for Random Forest, and Sequential Convolutional Neural Network (CNN) having the lowest result with an average of 92.3%. The comparative analysis indicates the effectiveness of the KNN algorithm in generating the best performances as opposed to other approaches The use of hybrid methods in the analysis of fraud is presented in Ref [19], which studies seven hybrid machine learning models for the identification of fraudulent transactions. The exploration also shows that improving different algorithms and finding the best combinations serves the need of fraud models very well as evidenced by the performance of the Adaboost + LGBM. Most of it was observed by Reference [20] while using neural network classifiers like Multilayer Perceptron (MLP) and Extreme Learning Machine (ELM) and it considered parameters like recall and classification speed for the assessment of the model. MLP delivered a very good accuracy level of about 97.84% thus showing its vulnerability detection competency towards fraud. Finally, in Reference [21], XGboost model is used to detect fraud in the Ethereum dataset with an accuracy of 99.21% which proves that the model is effective and efficient in dealing with large collections of transactions.

While various papers utilize metrics like accuracy and F1-score for performance evaluation, these metrics can produce varying results, complicating performance comparisons. The computational demands of implementing complex deep learning models also present a challenge, necessitating high-quality data that accurately reflects user behaviors. Lastly, the need for swift data processing and the capability to provide real-time responses to prevent fraud are ongoing challenges that researchers continue to face in this domain

Table 1. Research background

No.	Year	First Author	Method	Dataset Name	Accuracy (%)
1	2025	Feng, X.	Machine Learning Algorithms (Random Forest, Neural Networks, Logistic Regression, Naive Bayes)	Kaggle Credit Card Dataset	95.9%
2	2024	Sorour, S. E.	Brown Bear Optimization + Feature Selection	Australian Credit Dataset	91%
3	2025	Wang, Y.	SMOTE-KMEANS + Ensemble Learning	Kaggle Credit Card Dataset	97.4%
4	2025	Ahmad, I.	Hybrid Sampling (RUS + SMOTE) + Machine Learning	Kaggle Credit Card Dataset	85 (MCC)
5	2025	Alhabib, A. A.	Random Forest + KNN	Kaggle Credit Card Dataset	87.5%
6	2022	Singh, A.	Firefly Optimization + SVM	Kaggle Credit Card Dataset	85.6
7	2024	Chen, C. T.	Self-supervised Learning + Intelligent Sampling	Kaggle Credit Card Dataset	93%
8	2024	Sorour, S. E.	Brown Bear Optimization	Kaggle Credit Card Dataset	96.7%
9	2024	Yan, C.	Adaptive Model Optimization	Kaggle Credit Card Dataset	98.1%
10	2025	Kandpal, H.	Machine Learning Models Comparison	Kaggle Credit Card Dataset	92%
11	2025	Fan, X.	Cost-sensitive F-beta, Borderline SMOTE + Ensemble Learning	Kaggle Credit Card Dataset	90.3%
12	2023	Wang, H.	Ensemble Feature Selection	Kaggle Credit Card Dataset	98.7%

13	2022	Alfaiz, N. S.	Machine Learning Models	Kaggle Credit Card Dataset	97.94 (AUC), 95.91 (Recall)
14	2024	Feng, X.	Machine Learning Methods	Kaggle Credit Card Dataset	97.84%
15	2024	Chhabra, R.	Voting Ensemble Classifiers	Kaggle Credit Card Dataset	94.6%
16	2024	Juyal, P.	Deep and Machine Learning	Kaggle Credit Card Dataset	97.84 %(MLP)
17	2023	Mahajan, A.	Logistic Regression + Resampling	Kaggle Credit Card Dataset	94
18	2021	Mehbodniya, A.	Machine Learning + Deep Learning	Kaggle Credit Card Dataset	96.4%
19	2020	Azhan, M.	Machine Learning + Deep Learning	Kaggle Credit Card Dataset	97.2%
20	2020	Riffi, J.	MLP + ELM	Kaggle Credit Card Dataset	97.84
21	2022	Maurya, A.	Machine Learning	Kaggle Credit Card Dataset	97.84%

In this paper, a new model is proposed for overcoming the existing research limitations and improving the accuracy of the fraud detection by utilizing genetic algorithm, cross-entropy, particle swarm optimization (PSO), and support vector machines (SVM). The process starts with the preprocessing step of the classification of the data set where a binary genetic algorithm using cross entropy has been applied in the process of selecting the features. Therefore, PSO is then used to adjust the hyperparameters of the SVM. The selected features and the optimal parameter then applied in development of the SVM model. The subsequent structure of the rest of this paper is as follows: Section 2 is a brief overview of the algorithms on which the present method is based upon.

In the third section, authors provide information on the simulated dataset adopted in the research for training and assessment.

Section 4 describes the proposed method and how it is going to be applied.Section 5 identifies the method used to assess the outcomes of this research.The result of simulation is presented in section 6 while the comparison of the result with other methods is done in section 7.

Finally, the conclusions of this research study are presented in the last section of the document, section 8.

2. Basic concepts

This section introduces the core concepts utilized in our proposed method to enhance credit card fraud detection.

2.1. Cross-Entropy

The term 'Cross-entropy' is used regularly in information theory and machine learning to compare or assess the similarity of two probabilities, more so the true probability distribution with the predicted one. In classification, cross-entropy works in the capacity of a loss function that makes the model reach the vicinity of the ground fact and enhance the differentiation competence. Cross entropy was used to multiply the probability assigned to it by one distribution by the negative logarithm of the probability given to the same event by the other distribution which compares actual and predicted distributions in the dataset. This metric is one sided, which makes it especially useful in cases where accurate measure of the differences in distribution between the classes is needed. When determining the cross entropy between two vectors, each probability that is assigned by the second vector (Q is subtracted from the logarithm of the corresponding corresponding vector's probability (represented by P. Hence, the works minimize cross entropy to improve the distribution with the actual distribution hence making it easier to predict the data.

Cross-entropy is computed using the following formula:

$$H(p,q) = H(p) + D_{KL}(p \,||\, q) \tag{1}$$

Where $H(p)$ is the entropy and $D_{KL}(p \parallel q)$ is the Kullback–Leibler divergence from p to q.

2.2. Genetic Algorithm

B The Binary Genetic Algorithm (GA) is a form of optimization algorithms based on the natural selection approach and involves binary strings as a solution string. Beginning with a population at random, the program moves through the iterations of the evaluation of the fitness of solutions according to a pre-determined fitness function. This is achieved through the process of selection, crossover and then mutation which result to better fitness solutions in the population.

Selection: People who are fittest have a high likelihood of reproducing hence increasing their gene pool replication which works like the survival of the fittest.

Crossover: The selected two persons interchange some parts of the generated binary strings and create new strings that comprise genetic characteristics of the two parents.

Mutation: Perturbations are made to some of the numbers in the binary representations in the course of the search for a better solution in a space.

Binary GA is particularly suitable in feature selection problems whereby an element is either selected or unselected. In this regard, binary strings well capture the solutions, and thus facilitate the functioning and feature selection optimization of Binary GA.

2.3. Particle Swarm Optimization

Particle Swarm Optimization (PSO), an optimization algorithm based on the sparking of improved performance of individuals in a systematic way, which is based on the metaphor of flocking behavior of a number of particles such birds or fishes. Initially, PSO was introduced by Kennedy and Eberhart in 1995 due to the aspect of simplicity and efficiency in solving optimization problems. In PSO, a population set of particles is searched for a solution within the search space. Every one of these units symbolizes a solution and makes a decision summarizing its experience and the experience of surrounding particles, thereby bringing the swarm to the best solution.

In PSO, particles will change its position and velocity in order to search for improved positions in the search space. The motion of each of the particles is controlled by two fundamental rules namely global best (gbest) as well as the personal best (pbest). While gbest is the best solution known to the whole swarm, pbest denotes the solution best known to each of the particles. In the best experience of every particle and global best, the particles are guided towards the promising areas in the search space. particles tend towards the best solution with every iteration.PSO is well suited in continuous as well as the multi-modal domains and the aim of these domains are to optimise or maximise a particular function towards a given set of parameters.

On the advantages of PSO one can mention that it is easy to use and does not require specific training of the user. It is useful in many cases of optimization because it does not ask for derivative information and consequently does not involve complex mathematical calculations. It also indicates that PSO has great expertise in exploration and exploitation activities. Besides that, to enhance their own answers, the particles try to scope the search space and find new favorable areas. This balance of exploration and exploitation makes sure that PSO is able to escape from the local optima and get closer to the global optima.

2.4. Support vector machine

One of the most effective models for the classification and regression tasks is the Support Vector Machine (SVM). SVM separates the features that are closer to the various classes and aims at finding the best hyperplane that gives the greatest margin between the two classes. It is chosen to pass halfway between the set of marginal points or support vectors in the two classes. SVM has its initial basic idea of using the kernel function to map the input data into a higher feature space. To be able to separate the data points of different classes in this feature space, a hyperplane is constructed.Later, due to the kernel function selection, SVM is capable of dealing with the linearly non-separable data by mapping it into a higher dimension. The kernels that is very often used are the radial basis function (RBF), polynomial, linear, sigmoid. In other words, the SVM method aims to find the right hyperplane in order to maximize the margin while minimizing the classification errror, a task is defined by a quadratic optimization problem. This is done using the building of a number of Lagrange multipliers to solve the Lagrange dual problem. The points that are lying on the margin or are classified wrongly are termed as the support vectors and form the final decision border.

The following is the formula of the SVM:

$$f(x) = sign\left(\sum_{i=1}^{n} \alpha_i y_i k(x, x_i) + b\right) \quad (2)$$

Here, f(x) stands for the predicted value of class label for the new instance, '.&mgr;_i' denotes the values of Lagrange multiplier, y_i is the class label of an instance x_i from the training sample, 'k(x,x_i)' is a kernel function that measures the similarity between the input instance and the training sample, and 'b' is the bias term. 243 proposed to define the sign function to predict the class label as the sign of the given value.

3. Methodology

The suggested feature selection, data preprocessing, and SVM hyperparameter optimization are the three primary components of the suggested approach. This section discusses these components' equations and concepts.

3.1 Preprocessing

To prepare the data for analysis and model training, the preprocessing step is essential. This step is crucial as it transforms raw data into a format that machine learning models can interpret. The credit card dataset, however, has already been processed to address confidentiality concerns, as mentioned in the previous section. As a result, it undergoes most of the main preprocessing steps, including feature extraction, data cleaning, and, if needed, text processing. However, the dataset has not yet been normalized.Normalization is a preprocessing technique that scales numerical features to a common range, preventing any single feature from dominating the learning process. This step is important for machine learning algorithms as it accelerates convergence, avoids bias toward specific features, and enables data comparison and interpretation in a meaningful way. In this research, the dataset undergoes min-max normalization. This technique transforms each feature into a common range, usually between 0 and 1, by subtracting the minimum value and dividing by the difference between the maximum and minimum values:

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{3}$$

where X represents the original feature vector, $X_{\min}$ is the minimum value of the feature, $X_{\max}$ is the maximum value of the feature, and X_{norm} is the normalized feature value.

3.2 Feature selection

Reducing dataset dimensions and selecting the best features are crucial steps following preprocessing. This is particularly important when working with an unbalanced dataset, as it allows the machine learning model to concentrate on solving the actual classification problem rather than identifying the most relevant features. This study introduces a new feature selection method for detecting credit card fraud. In this approach, a subset of features is chosen using a binary Genetic Algorithm (GA), followed by an evaluation of dissimilarities between feature pairs and the similarity between each feature and the target variable using cross-entropy.

First, a binary GA is applied to tackle the challenge of selecting the optimal subset of features from the high-dimensional credit card transaction dataset. The GA encodes each feature as either present or absent, representing a potential solution. Through selection, crossover, and mutation, the GA iteratively evolves and optimizes the feature subset, effectively exploring the feature space to find the most relevant subset for fraud detection.

By iteratively assessing and refining these feature subsets, the GA reduces dimensionality, eliminates redundant or irrelevant features, and enhances the effectiveness and efficiency of the subsequent analysis and modeling stages. After the GA identifies a feature group, cross-entropy is applied to evaluate the dissimilarity between each selected feature pair and the similarity between each feature and the target variable (fraud or non-fraud).

Cross-entropy, an information-theoretic metric, measures the difference between two probability distributions and is widely used in this context. To facilitate this, an entropy matrix is constructed, as outlined below.

$$\forall i \in [1,n],\ \forall j \in [1,n]$$

$$EM_{i,j} = \begin{cases} 1/H(F_i,TA) & i = j \\ \frac{1}{2}H(F_i,F_j) & i \neq j \end{cases} \Rightarrow \tag{4}$$

$$EM = \begin{bmatrix} 1/H(F_1,TA) & \frac{1}{2}H(F_1,F_2) & \cdots & \frac{1}{2}H(F_1,F_n) \\ \frac{1}{2}H(F_2,F_1) & 1/H(F_2,TA) & \cdots & \frac{1}{2}H(F_2,F_n) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2}H(F_n,F_1) & \frac{1}{2}H(F_n,F_2) & \cdots & 1/H(F_n,TA) \end{bmatrix} \tag{5}$$

where Fi is the ith feature vector, Fj is the jth feature vector, TA is the target vector, H is cross-entropy, EM is the suggested entropy matrix, and n is the number of features chosen by GA.

The correlation between chosen features and the target variable can be determined by calculating the reverse cross-entropy in the diagonal components of the Entropy Matrix (EM). Higher diagonal values in the EM matrix indicate a stronger similarity between feature distributions and the target distribution, which is an important goal in feature selection. Features that display high cross-entropy values, on the other hand, signal significant differences between their distributions and that of the target.Another goal of feature selection is to minimize redundant information. This is assessed by analyzing the non-diagonal elements in the EM matrix: higher values in these elements represent greater dissimilarity between feature pairs, suggesting that each feature provides unique information. A more effective feature selection scenario is achieved when the EM matrix values are generally higher, as this implies both strong relevance to the target and minimal redundancy among features.Since each feature pair is evaluated twice in the EM matrix, a coefficient of $\frac{1}{2}$ is applied to each non-diagonal element to balance their weight. However, the objective of the binary Genetic Algorithm (GA) is to minimize the cost function. Therefore, the cost function for GA is designed as follows:

$$Cost = \frac{n}{\sum_{i=1}^{n}\sum_{j=1}^{n} EM_{i,j}} \tag{6}$$

where Cost is the suggested cost function based on cross-entropy. The GA can follow an EM with higher element values by employing the reverse of the sum of all elements. Furthermore, as more picked features result in larger dimensions for the EM matrix and, thus, a smaller reverse value for the cost function, n is used in the numerator of the fraction to balance the number of selected features. As a result, the cost function tends to favor the more characteristics that are chosen. However, by raising the cost of scenarios where n is greater, employing n in the numerator creates a relative equilibrium in this respect.

3.3 classification

The support vector machine (SVM) classifier should be trained to identify credit card fraud after a suitable selection of characteristics has been chosen. Nevertheless, the hyperparameters of the SVM have a significant impact on

both its performance and capacity for generalization. Therefore, SVM hyperparameters are optimized in this research using the particle swarm optimization algorithm (PSO). The kernel function, the primary hyperparameter of SVM, has a direct impact on the classification decision boundaries. The radial basis function (RBF), which only has one parameter and has demonstrated excellent performance in several applications, is chosen as the kernel function of SVM in this work.Therefore, we only concentrate on optimizing two crucial SVM hyperparameters: the kernel parameter σ for the RBF kernel and the regularization parameter C. The trade-off between maintaining a low model complexity and obtaining a modest training error is managed by the regularization parameter C. In the meantime, the RBF kernel's breadth and the decision boundary's flexibility are determined by the kernel parameter σ.

The PSO technique is used to determine the ideal values for these hyperparameters. PSO has been successfully used to solve a variety of optimization issues and is well-suited for optimizing continuous-valued parameters. The first step in optimizing SVM's hyperparameters is to create a cost function that accurately estimates the effects of various hyperparameters. In this sense, an SVM is trained in the cost function using the hyperparameters and the training dataset after obtaining the C and σ parameters.After that, its performance ought to be assessed in order to look into the impact of the learned hyperparameters. Because of the dataset's imbalances, the area under the curve (AUC) of the receiver operating characteristic (ROC) curve is computed for this purpose rather than performance accuracy. As a result, all data is categorized as belonging to the class with more samples, but the optimization process is able to bypass the settings, resulting in excellent accuracy. The final cost function is determined by computing the AUC of each class and is defined as follows:

$$Cost = \frac{1}{n}\sum_{i=1}^{n} AUC_i \qquad (7)$$

where Cost is the final determined cost that needs to be optimized, AUCi is the calculated AUC for the classification of the ith class, and n is the number of problem classes, which is two for credit card fraud detection (two classes of fraud and normal transactions). The global best solution can then be obtained by adjusting the PSO parameters, which include population size, maximum number of iterations, number of decision variables, inertia weight, personal learning coefficient, global learning coefficient, lower bound of decision variables, and upper bound of decision variables. The population size and the maximum number of iterations are the first two parameters of ABC, and they both have a direct impact on how many cost functions are evaluated. Nevertheless, using big values for these parameters adds time and computational strain.Therefore, the maximum number of iterations and population size are thought to be 50 and 30, respectively, in order to have a reasonable trade-off between optimum duration and the number of function evaluations. Other significant parameters are the number of decision variables and their upper and lower bounds; since decision variables are SVM hyperparameters, the number of decision variables is equal to 2. Additionally, every potential condition should be fully covered by the ranges of each choice variable. Therefore, for C, the range of [0,5000] is taken into consideration, and for σ, the range of [0,20] is taken into consideration. The values listed in [23] can be used to adaptively modify the other parameters. The proposed PSO algorithm is used to optimize the hyperparameters of SVM by adjusting the PSO parameters.

According to mentioned steps, the flowchart of the proposed method can be seen in **Figure 1.**

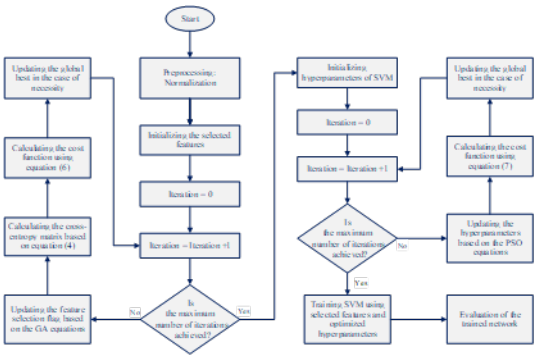

Figure 1. The flowchart of the proposed method.

4. Dataset

In this article, we train and evaluate the suggested model using the Credit Card Fraud Detection dataset. The Kaggle dataset on credit card fraud detection is very useful for developing and testing machine learning models. An extensive collection of anonymized credit card transactions done by European cardholders over a two-day period are included in the dataset. It was created especially to replicate real-world situations, in which most transactions are lawful and very few are fraudulent.

The dataset contains a total of 284,807 transactions, of which only 492 are labeled as fraudulent. This represents approximately 0.172% of the entire dataset, highlighting the highly imbalanced nature of the data.

The following are the salient characteristics of the Kaggle dataset on credit card fraud detection:

Transaction Features: Out of the 31 features in the dataset, the majority are numerical and anonymised for privacy reasons. These characteristics cover a range of transactional elements, including the amount, timing, and nature of the transaction (e.g., purchase, withdrawal).

Class Label: To identify whether a transaction is fraudulent (class 1) or lawful (class 0), the dataset offers a binary class label. There are substantially more genuine transactions than fraudulent ones, resulting in a very unbalanced class distribution. This disparity in class makes it difficult to create reliable fraud detection models.

Data preparation: To protect data privacy, the dataset has already completed a few preparation processes. Principal Component Analysis (PCA) has been used to alter the features, keeping just the original features' numerical representation. To maintain anonymity, most features are therefore designated as V1, V2, V3, etc.
Data Imbalance: As was already established, there is a significant class imbalance in the dataset, with fraudulent transactions making up a very small portion of the total. To guarantee the precise identification of fraudulent activity, this needs particular consideration when developing the model.
Anonymization: All personally identifying information (PII), including cardholder names and billing addresses, has been eliminated from the dataset in order to safeguard cardholder privacy. Analysis is limited to numerical features that capture the transaction characteristics.
For the purposes of model development and evaluation, the dataset was split into training and testing subsets using an 80/20 ratio. Stratified sampling was applied to ensure that both subsets preserved the original class distribution, allowing for a fair assessment of the model's performance across both majority and minority classes.

5. Evaluation Metrics

The performance of the suggested approach is assessed in this article using evaluation measures for accuracy, precision, recall, and F1 score. Additionally, we plot the Receiver Operating Characteristic (ROC) curve and determine the surface beneath it in order to compute AUC.
The parameters in the confusion matrix must be used in order to compute the assessment criteria. The confusion matrix is a table that displays the numbers of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions in order to provide an overview of a classification model's performance. It serves as the foundation for computing additional assessment measures and offers a thorough analysis of the model's performance for every class. The following definitions and calculations apply to each of the evaluation criteria used in this study:
Accuracy: A key evaluation parameter that gauges the general correctness of credit card fault detection is accuracy. It shows the proportion of cases that were correctly classified—both true positives and true negatives—to all instances. Accuracy in credit card fault detection refers to the system's ability to differentiate between fraudulent and non-fraudulent transactions while taking into account both accurate positive and negative predictions.

$$Accuracy = \frac{TP_y + TN_y}{TP_y + TN_y + FN_y + FP_y} \quad (8)$$

Precision: Precision, sometimes referred to as the positive predictive value, calculates the percentage of accurately detected credit card errors among all anticipated positive occurrences. Precision measures the system's capacity to precisely detect real defects while reducing false positives in the context of credit card fault detection. A low percentage of transactions that are mistakenly reported as fraudulent is indicated by a high precision value.

$$Precision = \frac{1}{n_c}\sum_y \left(\frac{TP_y}{TP_y + FP_y} \right) \quad (9)$$

Recall: A credit card fault detection system's recall, also known as sensitivity or true positive rate, gauges its capacity to accurately identify all real positive cases, or fraudulent transactions. It measures the percentage of frauds that are accurately identified out of all fraudulent transactions. A high recall value shows how well a system captures the majority of fraudulent activity.

$$Recall = \frac{1}{n_c}\sum_y \left(\frac{TP_y}{TP_y + FN_y} \right) \quad (10)$$

F1 Score: The F1 score is a balanced indicator of the effectiveness of the credit card fault detection system since it integrates precision and recall into a single statistic. With a range of 0 to 1, it is the harmonic mean of precision and recall. The F1 score can be used to assess how well credit card fault detection systems work because it takes into account both false positives and false negatives. An improved balance between recall and precision is shown by a higher F1 score, which denotes a more

dependable system for detecting credit card errors.

$$F_1 Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \quad (11)$$

6. Simulation results

In this section, the results of simulating the proposed method are presented.
Generalizability
To thoroughly evaluate the proposed method's outcomes, all stages discussed in previous sections were implemented in the MATLAB environment. One critical parameter in this process is the proportion of data allocated to test and training datasets, which directly impacts the generalizability of the credit card fraud detection approach. The test dataset, in particular, helps assess how well the method can detect fraud in new, unseen data. Due to the limited size of the training dataset, this approach can accurately detect credit card fraud even with a small number of training samples, making it practical for real-world applications.
To explore the generalizability of the proposed method, we evaluated four different data division scenarios, allocating 20%, 30%, 40%, and 50% of the data as the test set, with the remaining 80%, 70%, 60%, and 50% used for training, respectively. We used the Holdout method for data splitting.

The accuracy results across these data partitions are shown in **Figure 2**. As observed, the model learns patterns more effectively and shows a slight accuracy increase as the training dataset size increases. Notably, all configurations yielded accuracies above 99.9%, indicating that the method can generalize well even with a smaller training set. However, the performance approaches near-perfect accuracy when 70% or 80% of the data is used for training. Thus, for subsequent sections, 70% of the original dataset is used for training.

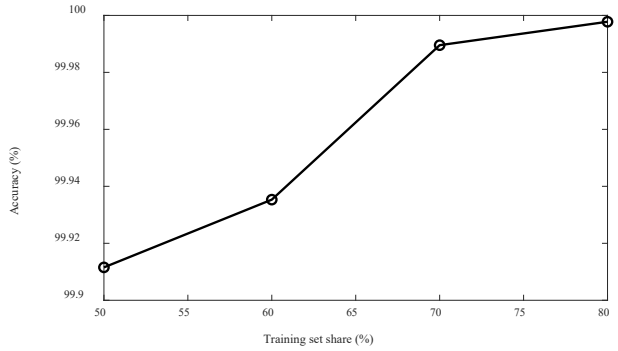

Figure 2. The accuracy of the proposed method using different data partitioning proportions.

6.2 Comprehensive evaluation

This section presents the results from a comprehensive evaluation of the proposed approach. Feature selection is the first critical aspect assessed in this approach. The binary Genetic Algorithm (GA) and a cost function are applied to the training dataset as part of the feature selection process, with the GA parameters set accordingly. The two main factors that directly influence the number of cost function evaluations are the population size and the maximum number of iterations. Selecting sufficiently high values for these parameters can reduce computation time while increasing the likelihood of finding the global optimal feature subset.

Thus, for an effective balance between evaluation points and computational efficiency, we set the population size to 30 and the maximum iterations to 50. The GA's convergence curve for feature selection is shown in **Figure 3**. As indicated, the curve converged after approximately 300 cost function evaluations, suggesting that this number of evaluations was sufficient to reach the global best solution. Ultimately, the best feature subset was determined to include only the last and seventeenth features.

Following feature selection, the SVM hyperparameters were optimized using Particle Swarm Optimization (PSO) with similar configurations to the GA. The optimized hyperparameters are displayed in **Table 2**, where the σ value is notably high at 15.74. This high value implies that, in cases where the Radial Basis Function (RBF) kernel is not feasible, a linear kernel might serve as a viable alternative. Moreover, the effectiveness of the feature selection process is evident, as the large σ value helps the model reduce the risk of overfitting.

With the optimal features and hyperparameters, the SVM model was trained, and the model's performance was then evaluated. The trained SVM was used to predict labels for the test dataset, with the predicted values compared against the true target labels. To comprehensively assess the method's effectiveness, we computed the confusion matrix, ROC curve, and other evaluation metrics, as detailed in Section 5. **Figure 4** presents the results, with all metrics exceeding 99.99%. Additionally, **Figure 5** shows that only six fraudulent samples and three normal samples were misclassified.

Finally, the ROC curve in **Figure 6** illustrates the relationship between the true positive and false positive rates. The near-perfect area under the ROC curve (AUC) of almost 1.0 demonstrates the exceptional performance of the proposed approach.

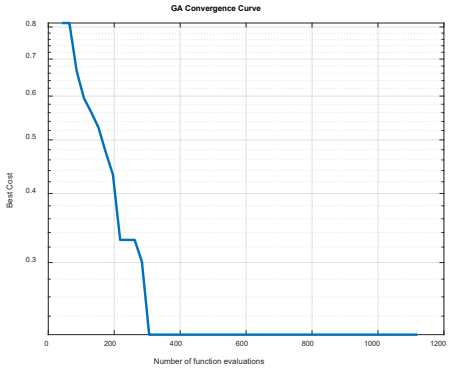

Figure 3. The convergence curve of GA in determining the best features.

Table 2. The obtained optimal hyperparameters for SVM

Hyperparameter	Value
σ	15.7425
C	2842.70

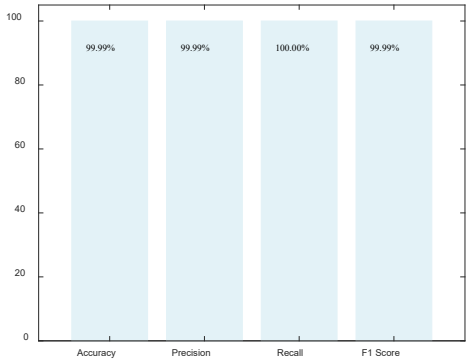

Figure 4. Accuracy, precision, recall, and F1 Score of the proposed method

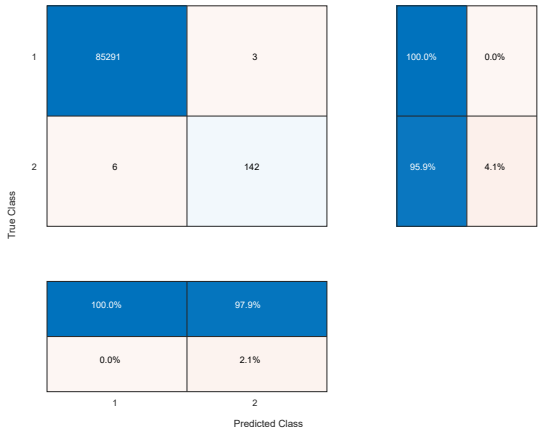

Figure 5. The confusion matrix of the proposed method.

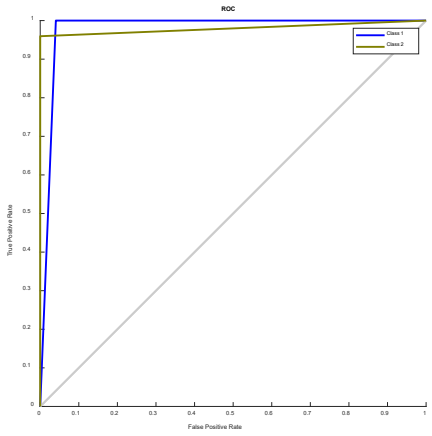

Figure 6. The ROC curve of the proposed method.

7. Comparison

Transaction fraud is on the rise due to the increased use of credit cards, driven by the growth of e-commerce and communication technology [24]. Altab Althar Taha and Sareef Jameel Malbery developed a technique to detect credit card fraud by utilizing an enhanced Light Gradient Boosting Machine (LightGBM) that combines parameter tuning with Bayesian-based hyperparameter optimization. They tested this approach on two publicly available, real-world datasets containing both fraudulent and non-fraudulent transactions, achieving 98.40% accuracy, 92.88% AUC, 97.34% precision, and an F1-score of 56.95% [25].

In another study [26], a neural network (NN)-based unsupervised learning algorithm was proposed to detect credit card fraud. This method outperformed several existing techniques, including autoencoder (AE), separation forests, local outlier factors, and K-Means clustering, achieving a high accuracy rate of 99.87%.

Similarly, in reference [27], Esenogho, Ebenezer, et al. proposed a hybrid data resampling technique combined with a neural network ensemble classifier as an effective approach for credit card fraud detection. This system utilized the Adaptive Boosting (AdaBoost) technique to build the ensemble classifier, using a Long Short-Term Memory (LSTM) neural network as the base learner. Additionally, hybrid resampling was achieved with the Edited Nearest Neighbor (SMOTE-ENN) method and the Synthetic Minority Oversampling Technique (SMOTE). The LSTM ensemble method achieved an accuracy of 99.6%.

A comparative summary of the proposed approach and referenced methods is presented in **Table 3**, clearly indicating the potential of the proposed approach to improve the effectiveness of credit card fraud detection.

Table 3: comparison of the proposed method with some previous methods

References	Methodology	Accuracy (%)
[22]	enhanced light gradient boosting machine	98.40
[23]	neural network based unsupervised learning	99.87
[24]	AdaBoost (LSTM)	99.8
Proposed method	GA / Cross-entropy / PSO / SVM	99.99

Conclusion

Credit card fraud results in serious consequences and substantial financial losses for individuals, businesses, and financial institutions alike. Effectively addressing these losses requires an accurate and dependable fraud detection method. In this paper, the critical challenge of detecting credit card fraud is addressed by proposing an advanced approach based on Support Vector Machines (SVM) integrated with an improved feature selection technique. In the proposed method, cross-entropy is combined with a binary genetic algorithm to select the most relevant features. This hybrid approach enables the evaluation of each feature's relationship with the target variable, thereby identifying those that are most indicative of fraudulent activity. By isolating these key features, the overall accuracy of the fraud detection system is significantly enhanced. For the classification stage, the SVM model is employed due to its strong performance in complex

classification tasks. To further improve the model's effectiveness and optimize computational resources, particle swarm optimization (PSO) is applied for fine-tuning the SVM hyperparameters. Rigorous testing on the Credit Card Fraud Detection dataset has demonstrated the method's effectiveness, with a high accuracy rate of 99.99% achieved. These results underscore the method's ability to accurately detect fraudulent transactions while minimizing false positives—an essential aspect for maintaining trust among consumers and institutions. A key strength of the proposed approach lies in its comprehensive strategy for fraud detection. Through the integration of a feature selection process that quantitatively assesses each feature's distinction and relevance to the classification target, a more precise and efficient system is developed. Furthermore, the classification capabilities of the system are strengthened by the optimized SVM model enhanced through PSO, resulting in a robust tool for accurate and reliable fraud detection.

Acknowledgement

None.

Conflict of interest

None.

References

[1] Feng, X. (2025). Credit card fraud detection using machine learning algorithms (Master's thesis, University of California, Los Angeles).

[2] Sorour, S. E., AlBarrak, K. M., Abohany, A. A., & Abd El-Mageed, A. A. (2024). Credit card fraud detection using the brown bear optimization algorithm. Alexandria Engineering Journal, 104, 171-192.

[3] Wang, Y. (2025). A Data Balancing and Ensemble Learning Approach for Credit Card Fraud Detection. arXiv preprint arXiv:2503.21160.

[4] Ahmad, I., Waleed, M., Ullah, A., & Jamil, M. U. (2025, February). Enhancing Credit Card Fraud Detection with Hybrid Sampling and Machine Learning Models. In 2025 6th International Conference on Advancements in Computational Sciences (ICACS) (pp. 1-7). IEEE.

[5] Alhabib, A. A., Alasiri, A. F., Alharbi, M. B., Ahmad, S., & Eljialy, A. E. M. (2025). Credit Card Fraud Detection Using Random Forest and K-Nearest Neighbors (KNN) Algorithms. In International Conference on Cognitive Computing and Cyber Physical Systems (pp. 383-395). Springer, Singapore.

[6] Singh, A., Jain, A., & Biable, S. E. (2022). Financial Fraud Detection Approach Based on Firefly Optimization Algorithm and Support Vector Machine. Applied Computational Intelligence and Soft Computing, 2022.

[7] Chen, C. T., Lee, C., Huang, S. H., & Peng, W. C. (2024). Credit Card Fraud Detection via Intelligent Sampling and Self-supervised Learning. ACM Transactions on Intelligent Systems and Technology, 15(2), 1-29.

[8] Sorour, S. E., AlBarrak, K. M., Abohany, A. A., & Abd El-Mageed, A. A. (2024). Credit card fraud detection using the brown bear optimization algorithm. Alexandria Engineering Journal, 104, 171-192.

[9] Yan, C., Wang, J., Zou, Y., Weng, Y., Zhao, Y., & Li, Z. (2024, July). Enhancing credit card fraud detection through adaptive model optimization. In 2024 IEEE 7th International Conference on Big Data and Artificial Intelligence (BDAI) (pp. 49-54). IEEE.

[10] Kandpal, H., Usmani, T., Khan, A., Khan, B., & Srivastava, A. (2025). Integrating Credit Card Fraud Detection with Machine Learning Algorithms.

[11] Fan, X., & Boonen, T. J. (2025). Machine Learning Algorithms for Credit Card Fraud Detection: Cost-Sensitive and Ensemble Learning Enhancements. Available at SSRN.

[12] Wang, H., Liang, Q., Hancock, J. T., & Khoshgoftaar, T. M. (2023, August). Enhancing credit card fraud detection through a novel ensemble feature selection technique. In 2023 IEEE 24th International Conference on Information Reuse and Integration for Data Science (IRI) (pp. 121-126). IEEE.

[13] Alfaiz, N. S., & Fati, S. M. (2022). Enhanced credit card fraud detection model using machine learning. Electronics, 11(4), 662.

[14] Feng, X., & Kim, S. K. (2024). Novel machine learning based credit card fraud detection systems. Mathematics, 12(12), 1869.

[15] Chhabra, R., Goswami, S., & Ranjan, R. K. (2024). A voting ensemble machine learning based credit card fraud detection using highly imbalance data. Multimedia Tools and Applications, 83(18), 54729-54753.

[16] Juyal, P. (2024, March). Using Deep and Machine Learning Techniques to Spot Credit Card Fraud. In 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 752-758). IEEE.

[17] Mahajan, A., Baghel, V. S., & Jayaraman, R. (2023, March). Credit card fraud detection using logistic regression with imbalanced dataset. In 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 339-342). IEEE.

[18] Mehbodniya, A., Alam, I., Pande, S., Neware, R., Rane, K. P., Shabaz, M., & Madhavan, M. V. (2021). Financial fraud detection in healthcare using machine learning and deep learning techniques. Security and Communication Networks, 2021, 1-8.

[19] Azhan, M., & Meraj, S. (2020, December). Credit card fraud detection using machine learning and deep learning techniques. In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS) (pp. 514-518). IEEE.

[20] Riffi, J., Mahraz, M. A., El Yahyaouy, A., & Tairi, H. (2020, June). Credit card fraud detection based on multilayer perceptron and extreme learning machine architectures. In 2020 International Conference on Intelligent Systems and Computer Vision (ISCV) (pp. 1-5). IEEE.

[21] Maurya, A., & Kumar, A. (2022, June). Credit card fraud detection system using machine learning technique. In 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom) (pp. 500-504). IEEE.

[22] Taha, A. A., & Malebary, S. J. (2020). An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine. IEEE Access, 8, 25579-25587.

[23] Rai, A. K., & Dwivedi, R. K. (2020, July). Fraud detection in credit card data using unsupervised machine learning based scheme. In 2020 international conference on electronics and sustainable communication systems (ICESC) (pp. 421-426). IEEE.

[24] Esenogho, E., Mienye, I. D., Swart, T. G., Aruleba, K., & Obaido, G. (2022). A neural network ensemble with feature engineering for improved credit card fraud detection. IEEE Access, 10, 16400-16407.