

Original Article

Intelligent Credit Card Fraud Detection based on Sea Horse Optimization and Deep Sequential Learning

Moosa Swarnalatha^{1*}, Neelakantappa M²

¹Department of Computer Science & Engineering, Bharatiya Engineering Science and Technology Innovation University (BESTIU), Andhra Pradesh, India.

²Department of Information Technology, Vasavi College of Engineering, Telangana, India.

^{1*}Corresponding Author : moosaswarnalatha@gmail.com

Received: 17 October 2025

Revised: 25 July 2026

Accepted: 26 August 2026

Published: 30 September 2026

Abstract - Recently, credit card fraud has increased a significant threat to financial institutions and consumers, thus a robust and intelligent detection system is essential to prevent fraudulent activities efficiently. In this research, an optimized deep learning model, namely Sea Horse Optimization (SHO), based on feature selection and Long Short-Term Memory (LSTM) with an attention mechanism, based detection model to efficiently capture the temporal sequence in credit card transactions. The proposed SHO-LSTM framework is trained and evaluated on the Kaggle Credit Card Fraud Detection (CCFD) dataset, comprising 284,807 transactions. Additionally, a sampling technique, namely Synthetic Minority Oversampling Technique, is used in this research for solving the class imbalance issue effectively and enhance detection performance. Experimental results of the proposed model demonstrate that the SHO-LSTM model achieves superior performance with an accuracy of 99.98%, which is higher than the existing fraud detection methods such as CNN-SVM, CSO-DNN, and CSO-GRU. These findings underscore the effectiveness of combining metaheuristic optimization with deep learning for fraud detection in highly imbalanced datasets.

Keywords - Credit card transaction, Fraud detection, Long Short-Term Memory, Sea-Horse Optimization, Synthetic Minority Oversampling Technique.

1. Introduction

Recently, the substantial growth of the internet, smartphones, e-commerce, and online shopping has become more popular worldwide. Similarly, the online payment or e-payment for online shopping has replaced the conventional training payment techniques or systems for several companies and organizations [1, 2]. For online payments and transactions, the credit card is most widely used by people in e-payment, which has increased; that also increases the issue of fraudulent activities during transactions [3, 4]. These fraudulent activities and other malicious attacks during e-payment by credit card lead to losses in the financial field; thus, an efficient Credit Card Fraud Detection (CCFD) is essential to prevent losses [5].

The credit card fraud activities are referred to as the unauthorized access during online transactions that acquires the cardholder's information through cyber hacking, telephone communication, and text messaging that enables the unauthorized financial transactions [6-8]. To detect credit card fraudulent activities, many researchers have used artificial intelligence approaches such as Machine Learning (ML) algorithms to analyze and identify if a transaction is legitimate

or fraudulent. Various ML approaches and other correlated methods are used to identify the fraud activities, but the existing models face challenges due to skewed distribution and dataset insufficiency [9, 10]. The increasing sophistication of fraudulent techniques necessitates intelligent, adaptive detection systems capable of identifying subtle patterns in transactional data. The conventional machine learning models often struggle with high-dimensional data, and severe class imbalance greatly impacts the detection model's performance in real-world scenarios.

The dataset utilized for CCFD, which contains important classes like default payment activities and fraudulent activities, is underrepresented. Due to imbalanced classes and missing values, the machine learning algorithms failed to identify and learn the intricate details from these classes, which resulted in inaccurate detection results. This limitation makes the model more challenging to generalize and to determine the decision boundaries precisely. However, the ML algorithms mainly depend on the historic data that fails to adapt to new fraud attack patterns, which results in inaccurate results. To overcome these limitations, an optimized Deep Learning (DL) model, namely Sea Horse Optimization-based



Long Short-Term Memory (SHO-LSTM), is proposed to accurately detect the fraudulent activities. The primary contributions of this research are:

- The proposed SHO algorithm employed to select the best set of features helps in detection performance to reduce the inaccurate results, false alarm rate and enhance the detection performance.
- A DL-based LSTM model is employed for extracting temporal and sequential patterns in transaction data, while SHO ensures the model focuses on the most relevant features and uses optimal configurations.

The remaining part of this manuscript is structured as follows: The literature review is discussed in Section 2. The methodology is clearly explained in Section 3. Experimental results and discussion are illustrated in Section 4. Research findings of this research are summarized as a conclusion in Section 5.

2. Literature Review

In this section, existing ML and DL based detection models utilized for CCFD are analyzed and discussed as follows:

N. Prabhakaran and R. Nedunchelian [16] developed an Oppositional Cat Swarm (OCS) optimization algorithm-based feature selection model with Bidirectional-Gated Recurrent Unit (Bi-GRU) for CCFD. The developed Bi-GRU model was applied to identify the fraudulent activities during transactions by credit card holders, along with significant features that were selected by the OCS algorithm. The main advantage of the developed Bi-GRU model was that parameters were finetuned by a chaotic krill herd optimization algorithm to enhance the fraud detection performance. However, the developed OCS-based feature selection has limited feature information that affects the accurate prediction of fraudulent activities.

Tesfahun Berhane et al. [17] explored a hybrid model for CCFD based on Convolutional Neural Network and SVM (CNN-SVM) algorithms. The explored hybrid CNN-SVM method was integrated by replacement of CNN final output layer with an SVM classifier to detect fraudulent activities during transactions efficiently. The main advantage of the SVM model with the CNN model is that it has enhanced the binary classification of normal fraudulent transactions by generating the hyperplanes that help to differentiate normal and fraudulent activities. However, the explored hybrid CNN-SVM model failed to detect the fraudulent activities due to the presence of irrelevant features with redundant information.

T. Karthikeyan et al. [18] represented a Deep CNN (DCNN) based detection model with Competitive Swarm Optimization (CSO) for fraud detection. The introduced

model was used to analyze the anomalies in credit card transactions that differ from regular patterns. An advantage of the introduced model was that the DCNN learn and extracts the hierarchical features automatically, which helps to improve detection performance. However, the introduced DCNN-CSO model did not detect the fraudulent activities accurately due to an imbalance of data that affected the detection performance.

Lina Ni et al. [19] suggested a Light Gradient Boosting Machine (LightGBM) model-based fraud detection model that incorporated a Fraud Feature-Boosting mechanism. The suggested Light GBM model was a for-layer fraud detection model which utilized a Spiral Oversampling Balancing Technique (SOBT) to address the class imbalance issue by balancing the data in the minority class to the majority class. An advantage of the suggested detection model was the integration of the Compound Grouping Elimination (CGE) technique that eliminates highly redundant features to improve CCFD performance. However, the suggested Light GBM model failed to adopt the evolving nature of fraudulent activities because it often relies on the historic data.

Zahra Salekshahrezaee et al. [20] presented a CCFD model based on ensemble classifiers, which was an integration of random forest, CatBoost, LightGBM and XGBoost algorithms. The presented ensemble detection model was utilized to detect the fraudulent activities in credit card transactions effectively. The main advantage of the presented model was the incorporation of preprocessing techniques, such as Synthetic Minority Oversampling Technique (SMOTE) and Random Under-sampling (RUS), which were used for the minority class to balance the data.

Iman Akour et al. [21] designed a hybrid CNN-LSTM model to enhance CCFD by integrating an attention layer. The combination of a designed CNN-LSTM model efficiently captured the specific behavioral representations, where the CNNs focused on extracting spatial features while the LSTM handled temporal sequences. Additionally, the attention mechanism was incorporated with the approach to focus on the most relevant features, which improved the detection performance. However, the presence of irrelevant features made the designed hybrid model difficult to differentiate between normal and fraudulent transactions accurately.

Si Shi et al. [22] developed an attention-based Balanced Variational AutoEncoder (Bal-VAE) based detection model to identify fraudulent activities in cred card transactions. The developed Bal-VAE model with a Multi-Head Attention (MHA) mechanism was employed to detect credit card fraud efficiently by addressing the data imbalance issues using an augmentation technique, which greatly impacts fraud detection. However, the imbalanced ratios in the standard loss function make the model difficult to capture the minority fraud class, resulting in low precision.

From the above related works, it is observed that the existing methods face limitations in accurate detection, which are mentioned as follows: the hybrid models, such as CNN-SVM, CNN-LSTM, effectively capture spatial and temporal patterns, but it often struggles with irrelevant features and redundant information, which affect accurate detection. The advanced model, such as optimized DL, integrated with an attention mechanism and explainability artificial intelligence, improved feature selection and data balancing; however, it failed to adopt the evolving fraud tactics. To overcome this, an improved CCFD model is proposed in this research, namely SHO-LSTM, which effectively captures temporal and sequential patterns in transaction data that help in distinguishing between normal and fraudulent transactions.

3. Methodology

In this research, to detect credit card fraudulent activities using a deep learning model, the SHO-LSTM model is used. Figure 1(a) and (b). illustrates the overall framework for the proposed CCFD model. Initially, the transaction data are acquired and preprocessed by the SMOTE technique to address the data imbalance problem. After that, the preprocessed data are fed to the proposed optimization-based feature selection to remove redundant features, and then finally, the LSTM-based classifier is utilized for the detection of fraudulent activities efficiently.

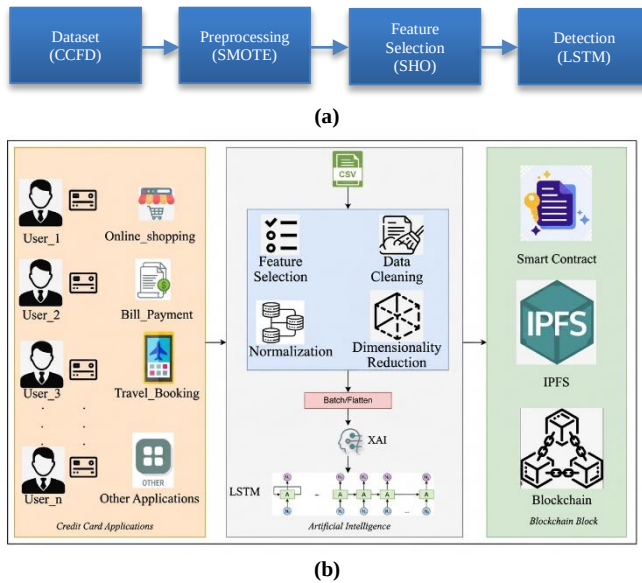


Fig. 1 (a) Block diagram of the proposed CCFD framework, and (b) Utilization of an LSTM Model for the classification of fraudulent patterns in credit card transactions.

The proposed SHO algorithm is utilized to discern and preserve the most pertinent features, hence minimizing noise and enhancing model efficiency. The optimized features are fed as input into the LSTM model, which is proficient at identifying temporal patterns and dependencies in transaction sequences. The SHO-LSTM framework improves the

accuracy, reliability, and scalability of fraud detection in financial systems through the integration of intelligent feature selection with deep learning.

3.1. Dataset

The CCFD is a publicly available benchmark dataset that is obtained from the Kaggle repository [22]. This dataset comprises 284807 data points, which are further divided into two classes: Fraud and Non-fraud /Normal. The description of the CCFD dataset is represented in Table 1. This dataset is acquired from two-day credit card transactions that were made in September 2013 by the cardholders in Europe countries. The CCFD dataset involves 31 numerical features, which are based on Time and Amount (V1, V28). In the final column, the transaction type is represented by the values such as 1 and 0, which indicate a fraudulent transaction and a non-fraudulent transaction.

Table 1. Description of the CCFD dataset

Description	Values
Attributes	31
Instances	58016
Number of Classes	2
Positive Samples	57879
Negative Samples	137

3.2. Preprocessing

The acquired data is fed into the preprocessing phase to convert the raw feature data into a useful format for further CCFD process. In this preprocessing stage, the SMOTE and Normalization techniques are utilized to transform the raw data.

Initially, the SMOTE method, which is one of the widely used sampling techniques to address the data imbalance issue, selects a point at random from the minority class. The K-nearest neighbor algorithm in SMOTE estimates K-nearest neighbors for this random point [23]. Artificial points are integrated among selected and neighbor point. Since the dataset consists of 57,879 positive samples and 137 negative samples, this leads to bias that affects the detection results. Hence, SMOTE is employed in this research to balance the class distribution. By using this oversampling process, the dataset consists of 57,879 samples per class. The configuration of this SMOTE technique is specified in Table 2.

Table 2. Details of SMOTE configuration for class imbalance

Configuration	Parameter/Value
Sampling strategy	Equal class balancing (1:1)
Number of Neighbors	3
Random State	42

Min-Max Normalization: the sampled data are further passed to normalization process by min-max normalization to

rescale the input values of data into a uniform range between [0,1] or [-1,1]. The mathematical representation of min-max normalization is given in Equation (1):

$$z_{\text{normalize}} = \frac{(z - z_{\min})}{(z_{\max} - z_{\min})} \quad (1)$$

Where, z and $z_{\text{normalize}}$ represents original and normalized value; $z_{\min}$ and $z_{\max}$ denotes the minimum and maximum values of z . These normalized data are further fed to the proposed feature selection process for selecting significant featured that help to enhance CCFD. This balanced dataset is then split into training and testing in a ratio of 80:20, while preserving the class distribution. Table 3 summarizes the configurations of preprocessing techniques used in the based CCFD framework.

Table 3. Preprocessing configuration summary

Step	Technique	Parameters / Notes
Data Imbalance Handling	SMOTE	k = 5 nearest neighbors; applied only to minority (fraud) class
Normalization	Min-Max Scaling	Rescaled features to range [0, 1];
Missing Values	Not explicitly mentioned	Assumed none or handled during SMOTE preprocessing
Feature Selection	Sea Horse Optimization (SHO)	Applied after normalization; reduced dimensionality by ~35%
Data Split	Train-Test Split	80: 20 for training and testing using a stratified sampling technique

3.3. Proposed SHO-based Feature Selection

In this research, a recent metaheuristic optimization algorithm is utilized for optimal feature selection to enhance CCFD. The traditional algorithms, such as PSO and the genetic algorithm [25], have limitations that have uniform and linear search patterns and suffer from premature convergence, which results in suboptimal results. Additionally, the Grey Wolf Optimization (GWO) algorithm is often limited by a fixed hierarchical structure, which leads to poor exploration activity that affects the selection of relevant features. Thus, the SHO algorithm is proposed in this research to overcome the above-mentioned limitations and enhance optimal feature selection. The SHO is the kind of population-based optimization that is inspired by the natural behavior of sea horses in the ocean. This SHO involves three primary phases, namely: Movement behavior, Predation behavior and Breeding behavior. In the movement phase, the sea horse includes spiral and Brownian motions, in the predation phase, which includes success and failure with a probability of 90% success: 10% failure, while searching and attacking behavior of the sea horse. During the breeding phase, the male seahorse has the main responsibility and forwards the genetic data to

their offspring, which helps to enhance the genetic diversity of the population.

3.3.1. Initialization

In this initialization phase of the SHO algorithm, the population is distributed across the search space, where each individual signifies a unique solution vector. The mathematical formulation of initialization is mentioned in Equation (2):

$$\text{Seahorses} = \begin{bmatrix} x_1^1 & \dots & x_1^{\text{Dim}} \\ \vdots & \ddots & \vdots \\ x_{\text{pop}}^1 & \dots & x_{\text{pop}}^{\text{Dim}} \end{bmatrix} \quad (2)$$

Where, the i^{th} The individual is initialized using the given Equations (3) and (4):

$$X_i = [x_i^1, \dots, x_i^{\text{Dim}}] \quad (3)$$

$$x_i^j = \text{rand} \times (Ub^j - Lb^j) + Lb^j \quad (4)$$

Where Ub and Lb denote upper limit and lower limit; rand represents a random number; i and j indicate positive integers.

3.3.2. Movement Behavior Phase

The sea horse utilizes a spiral and Brownian movements for location update in the search space. Spiral motion is primarily utilized in the local exploitation stage. The Levy flight is applied to obtain the step length of movement. Similarly, Brownian movement is utilized in the exploration stage. The detailed information on spiral movement and Brownian motion is explained as follows:

Spiral Movement

This motion behavior focuses on the local exploitation stage, when the random number is $r_1 > 0$, the sea horse moves towards an X_{elite} which denotes an elite individual corresponding to spiral motion.

Then, the Levy flight technique is utilized to update the movement step size of the sea horse; thus, the position of the sea horse is carried out with high probability in the prior iteration, which helps to mitigate excessive local exploitation. Moreover, the spiral movement changes its angle of rotation continuously, which expands the nearby present local solution. Mathematical representation of spiral motion is given in Equations (5)-(9):

$$X_{\text{new}}^1(t+1) = X_i(t) + \text{Levy} \cdot ((X_{\text{elite}}(t) - X_i(t)) \times x \times y \times z + X_{\text{elite}}(t)) \quad (5)$$

$$x = \rho \times \cos(\theta) \quad (6)$$

$$y = \rho \times \sin(\theta) \quad (7)$$

$$z = \rho \times \theta \quad (8)$$

$$\rho = \mu \times e^{\theta v} \quad (9)$$

Where, X_{elite} represents the position of elite individuals; x , y , and z that denotes 3 dimensional component of spiral movement; θ indicates a random value, ρ indicates the length of the stem; μ and v represent the logarithmic spiral constant. The Levy flight distribution in spiral motion is determined from Equation (10)

$$Levy = s \times \frac{\omega \times \sigma}{|k|^{\frac{1}{\lambda}}} \quad (10)$$

Where s denotes a constant 0.01; k and ω represent random value.

Brownian Movement

When the random value $r_1 \leq 0$, then the sea horse trajectory is determined by Brownian motion, which is specifically integrated to facilitate global exploration. By utilizing Brownian-distributed step sizes, the algorithm ensures a more diverse traversal of the search space. This mechanism is vital to maintain population diversity and prevent the model from premature convergence in local optima. Brownian motion is mathematically formulated by giving in Equation (11):

$$X_{new}^1(t+1) = X_i(t) + rand.l.\beta_t.(X_i(t) - \beta_t.X_{elite}(t)) \quad (11)$$

Where parameter l is fixed at a value of 0.05. The formula for β_t The random walk coefficient is mathematically expressed in Equation (12):

$$\beta_t = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \quad (12)$$

The movement update rule for the sea horse is determined by is given in Equation (13):

$$X_{new}^1(t+1) = \begin{cases} X_i(t) + Levy.(X_{elite}(t) - X_i(t)) \times x \times y \times z + X_{elite}(t) & r_1 > 0 \\ X_i(t) + rand.l.\beta_t.(X_i(t) - \beta_t.X_{elite}(t)) & r_1 \leq 0 \end{cases} \quad (13)$$

3.3.3. Predation Behavior

The random variable r_2 is used to determine the result of the predation behavior of the sea horse, when $r_2 > 0.1$, it represents the seahorse hunted and approached the food successfully, representing the development ability of SHO. If it is not, then it indicates the failed predation, and the sea horse

continued the process of exploration. This predation behavior is mathematically represented as in Equation (14):

$$X_{new}^2(t+1) = \begin{cases} \alpha.(X_{elite} - rand.X_{new}^1(t)) + (1-\alpha).X_{elite} & \text{if } r_2 > 0.1 \\ (1-\alpha).(X_{new}^1(t) - rand.X_{elite}) + \alpha.X_{new}^1(t) & \text{if } r_2 \leq 0.1 \end{cases} \quad (14)$$

Where, r_2 and $rand$ denotes a random number; $X_{new}^1(t)$ represents the new location of the sea horse after the t^{th} iteration; t stands for the present iteration. α indicates the weight of the influence factor.

3.3.4. Breeding Behavior

In this breeding behavior, the population is categorized into male and female sets based on the rank of fitness values. Moreover, the population is categorized into two equal subsets, where the top-performing half of the individuals represent the males and the low-performing half represent females. This categorization helps to improve the genetic features of the previous generation in the new offspring's. The mathematical formulation for this sea horse breeding behavior of individuals is expressed in Equation (15):

$$X_i^{offspring} = r_3.X_i^{father} + (1-r_3).X_i^{mother} \quad (15)$$

Where i represents a positive integer; r_3 indicates a random variable. In this breeding phase, the distribution pattern of male and female sea horses transfers their genetic characteristics from the prior generation and acquires better sea horse offspring. To enhance the feature selection, an elite reverse learning strategy is integrated, which explores a new set of features from the unexplored regions. This new set of features enhanced the CCFD, which adopts the evolving nature of the fraudulent activities. The mathematical formulation of this reverse learning strategy is given in Equation (16):

$$\bar{X}_i = k(lb + ub) - x_i \quad (16)$$

Where lb and ub denote lower and upper boundaries; x_i represents the present sea horse individual position information. The reverse individuals are obtained through the incorporation of elite reverse learning efficiently.

3.3.5. Fitness Function

The SHO approach with an elite reverse learning strategy utilized a fitness function that reaches increased classifier efficacy. Equations (17) and (18) represent the fitness criteria, where the optimal execution of a candidate solution corresponds to the minimum classification error rate. The most informative feature, which is known as the best solution that obtains the minimum error rate and the worst solution has a high error rate.

$$\text{Fitness}(x_i) = \text{classifier error rate}(x_i) \quad (17)$$

$$(x_i) = \frac{\text{number of misclassified instances}}{\text{Total number of instances}} \times 100 \quad (18)$$

The above-mentioned fitness equation is utilized to identify the subset of variables that yields the lowest possible value. x_i . The classification error rate is determined by the ratio of misclassified instances to the total number of samples. Feature combinations that minimize classification error and subsequently enhance predictive accuracy are referred to as the optimal solutions.

Additionally, the selection of appropriate features in the SHO model eliminates the noise and redundancy as “worst solutions” that prevent the model from the overfitting issue.

In CCFD, the selected features help to differentiate between normal and fraudulent activities efficiently from the high-dimensional transaction data. Algorithm 1 represents the feature selection process using the proposed SHO algorithm.

Algorithm 1:

Begin

Initialize population of sea horses (candidate solutions)

For each sea horse:

Randomly generate a position within the search space

Evaluate fitness using the classification error rate

While the termination condition is not met:

For each sea horse in the population:

Generate a random number r_1

If $r_1 > 0$:

Perform Spiral Movement toward elite solution

Update position using Levy flight

Else:

Perform Brownian Movement for global exploration

Update position using a random walk

Generate a random number r_2

If $r_2 > 0.1$:

Successful predation: move toward elite

Else:

Failed predation: continue exploration

Rank sea horses by fitness

Select the top 50% as males, the rest as females

For each offspring:

Generate a new solution using a weighted average of male and female

Apply Elite Reverse Learning to enhance diversity

Evaluate the fitness of the new population

Update the elite solution if a better solution is found

Return the best feature subset with minimum classification error

End

3.4. Detection

After feature selection, the selected significant features are fed as input to LSTM based fraud detection model. The LSTM is a type of RNN model that is extensively utilized for sequential data, which has the capability to learn long-range dependencies [3]. Unlike standard RNNs, the LSTM model involves a gating mechanism that preserves information over time and mitigates vanishing gradient issues, which enables the model to learn complex fraud patterns efficiently. However, the CNN models are excelling at capturing spatial features but lack temporal information, which is crucial to identifying fraudulent activities. The LSTM model excels at capturing this temporal information by maintaining an internal cell state, which acts as a continuous memory that allows the model to interpret new transactions within the context of past actions. Figure 2 represents the LSTM architecture used for fraud detection in credit card detection.

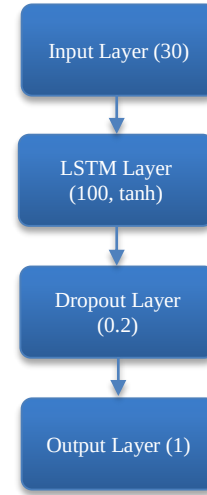


Fig. 2 The architecture diagram of the LSTM model

In LSTM, there are three important gates (input, forget and output) and one memory cell, which leverage the data processing from the input layer to the output layer [23]. The mathematical representation of the gating mechanism in LSTM and the memory unit is formulated in Equations (19) to (22):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (19)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (20)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (21)$$

$$C_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (22)$$

Where W and b denote weights and bias of the layers.; i_t , f_t , o_t , and C_t represents input, forget, output gates and a memory cell unit. However, the LSTM model has certain limitations, which have a fixed-size memory that often forces

the model to forget previous layer information in long transaction sequences. Hence, a self-attention mechanism is incorporated with the LSTM model to dynamically capture the previous time steps to improve the detection performance. This self-attention mechanism assigns varying importance weights to the significant historical transactions, which helps the LSTM model to maintain previous layer information and adapt to the evolving transaction behavior, which leads to accurate detection results.

The selected 31 preprocessed features are given as input to the LSTM's input layer. These optimized features are then processed through two sequential LSTM layers, where the first layer contains 128 units and the second layer consists of 64 units, respectively. Then, a Tanh-based activation function is used in both layers with a dropout rate of 0.2 to capture the temporal dependencies while preventing overfitting. Finally, the data passes through a dense layer, which involves 32 units with Rectified Linear Unit (ReLU) activation function. Additionally, a sigmoid activation function is utilized to perform the final binary classification between fraudulent and legitimate transactions.

4. Results and Discussion

4.1. Experimental Settings

Experimental results of the SHO-LSTM approach, which utilizes the CCFD dataset, are described in this section, Table 3. The SHO-LSTM is simulated using Python 3.9 tool with system configuration of Windows 10, Intel i5 processor and 16 GB RAM, respectively. The experimental setting for the proposed SHO-LSTM model is meticulously described to guarantee reproducibility and transparency. The implementation utilized Python 3.9, employing essential libraries including TensorFlow 2.9 for deep learning model development, scikit-learn 1.1 for preprocessing and evaluation metrics, and imbalanced-learn 0.9 for the application of the SMOTE. The experiments were conducted in an isolated environment devoid of GPU acceleration, with all random seeds standardized to guarantee uniform results across trials. This setup establishes a solid basis for reproducing the study and verifying the efficacy of the suggested framework. Table 4 depicts the hyperparameter details of the proposed SHO-LSTM model-based fraud detection model for CCFD.

Table 4. Hyperparameter settings of SHO-LSTM in CCFD

Parameter	Value
Learning rate	0.0001
Epoch	100
Activation function	Sigmoid
Batch size	64
Optimizer	Adam
Loss function	Binary_crossentropy

4.2. Evaluation Metrics

Performance measures like recall, F1-score, precision and accuracy are considered for evaluating the proposed model with previous DL and optimization models. Mathematical representation of the above-mentioned evaluation metrics is represented in Equations (23) to (26):

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} * 100 \quad (23)$$

$$Precision = \frac{(TP)}{(TP+FP)} * 100 \quad (24)$$

$$Recall = \frac{(TP)}{(TP+FN)} * 100 \quad (25)$$

$$F1 - score = 2 \times \frac{(Precision \times Recall)}{(Precision+Recall)} * 100 \quad (26)$$

Where, *TP*- True Positive, *TN*- True Negative, *FP*- False Positive, and *FN*- False Negative.

4.3. Performance Analysis

Performance evaluation of the proposed SHO-LSTM model with various classifiers based on feature selection is depicted in Table 5. The existing DL approaches, such as CNN, LSTM, RNN, and Gated Recurrent Unit (GRU), are considered for the comparative analysis with the proposed method in CCFD.

The LSTM approach with SHO-based feature selection achieved higher accuracy than one without integration of feature selection. Due to the selection of relevant transaction features from the dataset, the SHO algorithm and the efficient learning of temporal information by the LSTM model enhanced the detection performance over existing models.

Table 5. Performance analysis of the proposed method utilized for CCFD based on feature selection

Methods	Feature Selection	Precision (%)	Accuracy (%)	Recall (%)	F1-score (%)
CNN	No	86.41	86.42	86.42	86.41
RNN		89.82	89.83	89.82	89.83
GRU		92.93	92.94	92.93	92.94
Proposed LSTM		96.46	96.47	96.47	96.47
SHO-CNN	Yes	96.36	96.35	96.34	96.35
SHO-RNN		97.54	97.56	97.55	97.55
SHO-GRU		98.65	98.64	98.63	98.64
Proposed SHO-LSTM		99.97%	99.98%	99.96%	99.96%

Figure 3 illustrates the performance evaluation of the SHO algorithm with various state-of-the-art algorithms for feature selection in CCFD. The existing algorithms, such as PSO, GWO, Salp Swarm Optimization (SSO) and Jellyfish Search Optimization (JSO), are utilized and retrained on a similar experimental setup for comparative evaluation with the proposed method.

The results confirm that the proposed SHO algorithm-based feature selection method identified the most important transaction-related features. These selected features help the detection model to focus on critical fraud indicators and detect fraudulent activities accurately.

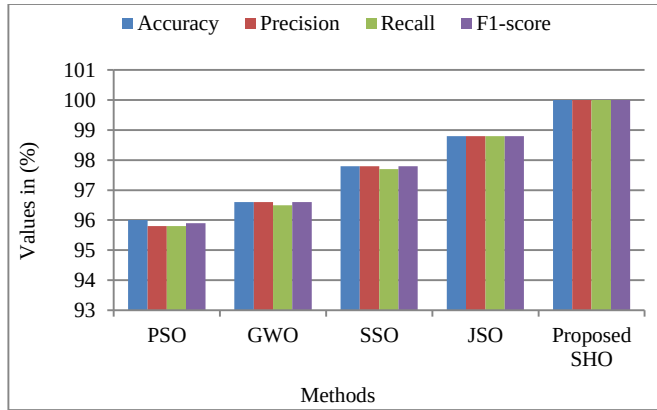


Fig. 3 Evaluation of the proposed feature selection method with the existing optimization algorithm in CCFD using the benchmark dataset

Figure 4 (a) and (b) represent the analysis of the SHO-LSTM model performance in terms of the confusion matrix and ROC curve. From the results, it is clear that the SHO algorithm's ability to select the most discriminative features reduces noise and enhances the LSTM model's capacity to capture complex temporal dependencies in transaction data.

This leads to achieving high AUC and minimal classification errors, which highlights the model's robustness and effective separation between fraud and non-fraud/normal classes.

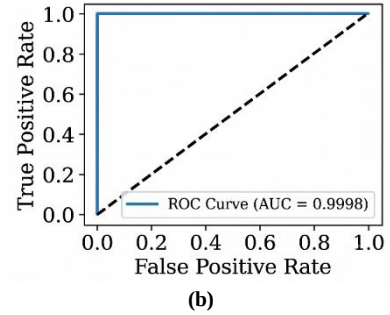
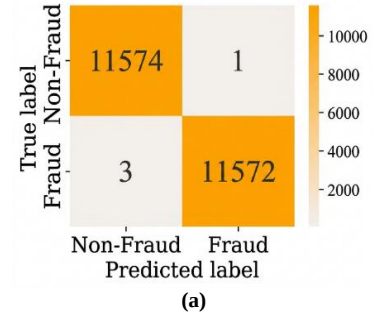
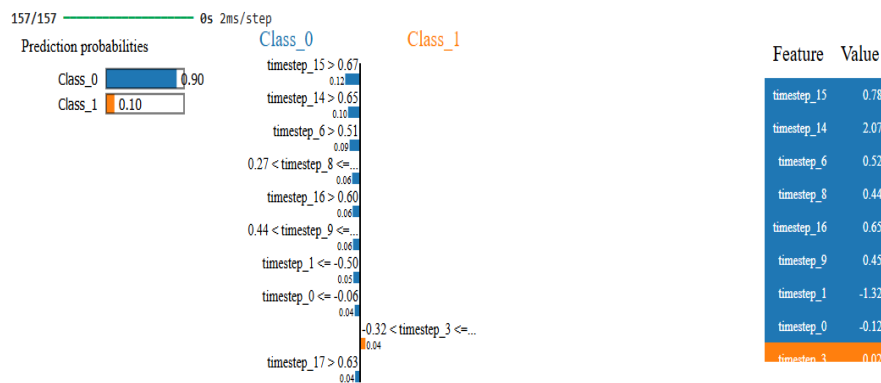


Fig. 4 (a) Confusion matrix, and (b) ROC curve of the proposed SHO-LSTM model utilized for fraud detection during credit card transactions.

Figure 5 (a) and (b) represent the error analysis of the proposed SHO-LSTM model employed for CCFD. From Figure 5 (a), the SHO algorithm selects high-impact temporal features and filters noise, which allows the model to focus on significant sequential dependencies. This relevant selection ensures that only statistically relevant patterns are utilized for detection, which results in high prediction accuracy and reduced MSE in CCFD. Also, in Figure 5 (b), the prediction graph illustrates that for correct predictions, the transaction feature values, which are plotted in light blue, consistently follow a stable flow with the learned distribution with moderate fluctuations. In contrast, the false negative, which is plotted in a red line, occurs because its feature pattern closely overlaps with the distribution of legitimate transactions due to a lack of effective discrimination between normal and fraudulent transactions.



(a)

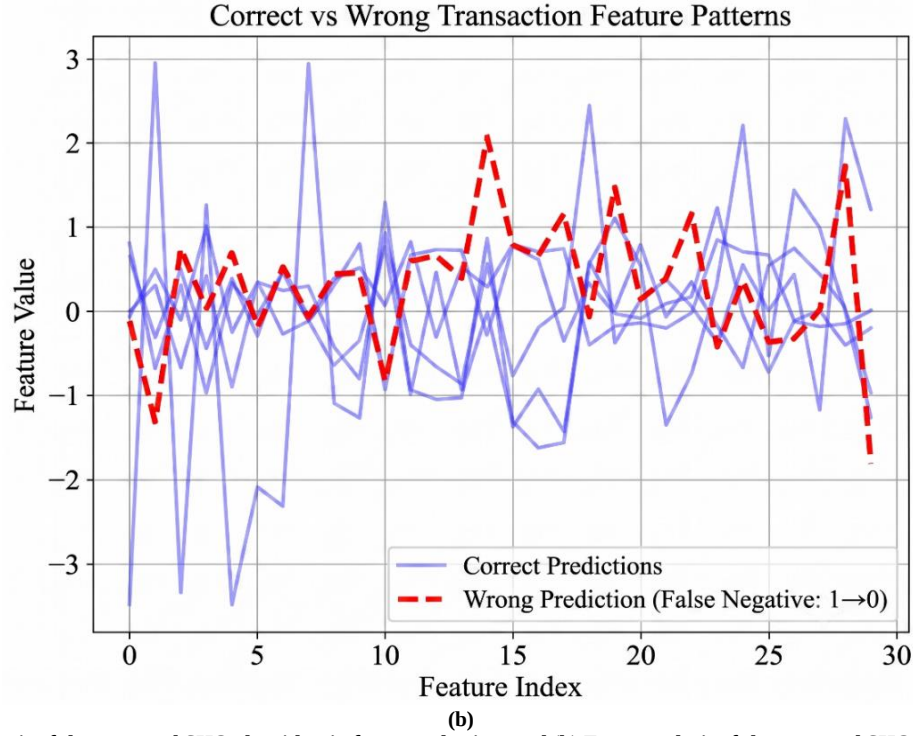


Fig. 5 (a) Error analysis of the proposed SHO algorithm in feature selection, and (b) Error analysis of the proposed SHO-LSTM model for the detection of fraudulent activities in credit card transactions.

Table 6 represents an ablation study of the SHO-LSTM model in CCFD with different configurations using the Kaggle CCFD dataset. From this ablation study, the combination of SMOTE+SHO+LSTM achieved superior

performance where the SHO algorithm dynamically focus on the most relevant feature. Also, the integration of the self-attention mechanism with LSTM addresses the limitations of the LSTM model and leads to accurate detection performance.

Table 6. Ablation study of the proposed CCFD model in terms of various configurations

Configuration	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
LSTM	86.42	86.41	86.42	86.41
SMOTE+LSTM	89.83	89.82	89.82	89.83
SHO+LSTM	92.94	92.93	92.93	92.94
SMOTE+SHO+LSTM	99.26	99.27	99.26	99.27
Proposed SMOTE+SHO+LSTM with attention mechanism	99.98	99.97	99.96	99.96

Table 7 illustrates computational complexity analysis of the SHO-LSTM model in CCFD with existing algorithms in terms of computational time, memory usage and inference time. From this analysis, it is clear that the proposed model attained a computational time higher than RNN and GRU models due to the architectural complexity of LSTM. However, the LSTM is more computationally expensive, but

the improved accuracy justifies the additional processing time. To quantify the impact of the proposed SHO on the baseline LSTM model in CCFD performance, the proposed SHO algorithm is evaluated based on five different metrics, which are represented in Table 8. The results clearly demonstrate that integration of SHO for feature selection significantly enhances detection accuracy and robustness.

Table 7. Computational complexity analysis of the SHO-LSTM model using the Kaggle CCFD dataset

Methods	Computational time (s)	Memory usage (MB)	Inference time (s)
CNN	452.3	840	0.042
RNN	310.5	420	0.070
GRU	350.2	510	0.062
LSTM	425.8	680	0.078
Proposed SHO-LSTM	360.4	395	0.080

Table 8. Performance comparison: LSTM vs. SHO-LSTM

Metric	LSTM (Without SHO)	SHO-LSTM (With SHO)
Accuracy	93.90 %	99.98 %
Precision	92.60 %	99.97 %
Recall	91.00 %	99.96 %
F1-Score	91.80 %	99.96 %
ROC-AUC	0.973	0.998

To ascertain that the observed performance enhancements are not attributable to random chance, various statistical significance tests are performed comparing the SHO-LSTM model with baseline models, including CNN-SVM, CSO-DNN, and CSO-GRU.

1. Paired t-Test

A paired t-test is performed based on the accuracy ratings derived from 10-fold cross-validation of each model. The null hypothesis posited no significant difference between SHO-LSTM and each baseline model. The p-values for SHO-LSTM compared to CNN-SVM, CSO-DNN, and CSO-GRU were all less than 0.01, signifying statistically significant enhancements.

2. Wilcoxon Signed-Rank Test

Due to the non-normal distribution of precision and recall scores, the Wilcoxon signed-rank test was employed to compare SHO-LSTM with other models over various folds. The SHO-LSTM consistently surpassed baseline models with p-values < 0.05, affirming its resilience across several criteria.

3. Confidence Intervals

The proposed SHO-LSTM model achieved 95% confidence intervals for each statistic through bootstrapping over 1,000 resamples.

4.4. Comparative Analysis

A comparative analysis is conducted on the CNN-SVM, CSO-DNN, and CSO-GRU models under uniform conditions in Table 9. Existing CCFD detection models such as CSO-GRU [16], CNN-SVM [17], CSO-DNN [18], CNN-LSTM [21] and Bal-VAE + attention [22] are utilized for comparative evaluation in this research. The results obtained by CNN-LSTM [21] and Bal-VAE + attention [22] models are converted into percentages to provide uniform comparative results.

The SHO-LSTM model surpassed all baseline models, especially in recall and ROC-AUC, demonstrating its efficacy in detecting fraudulent transactions. The application of SHO lowered feature dimensionality by 35%, enhancing training efficiency while maintaining accuracy.

Table 9. Comparative study of SHO-LSTM with existing fraud detection models

Model	Precision (%)	Recall (%)	Accuracy (%)	F1-Score (%)
CSO-GRU [16]	99.30	99.25	99.35	99.27
CNN-SVM [17]	98.65	98.42	98.76	98.53
CSO-DNN [18]	99.10	98.95	99.12	99.02
CNN-LSTM [21]	90.51	89.27	99.93	89.88
Bal-VAE + attention [22]	99	99	N/A	N/A
Proposed SHO-LSTM	99.97	99.96	99.98	99.96

4.5. Discussion

From the performance analysis and comparative analysis results of the proposed method, it is clear that the SHO-LSTM method attained better results than state-of-the-art methods and existing detection models for CCFD. The SHO-LSTM model attained better results, which is greater than existing approaches, which have several limitations, such as OCS-based feature selection [16], which has limited feature information that affects the accurate prediction of fraudulent activities. The hybrid CNN-SVM model [17] failed to detect the fraudulent activities due to the presence of irrelevant features with redundant information. DCNN-CSO model [18] faces challenges in detecting fraudulent activities accurately due to the imbalance of data that affects the detection performance. The proposed SHO-LSTM model enhanced the CCFD by adopting the evolving nature of fraudulent activities

and identify the anomalies efficiently through the spiral and Brownian motions. Moreover, the LSTM model utilized the previous layer information in a sequence order that enhanced the detection performance. This research introduces a hybrid SHO-LSTM system for detecting credit card fraud, integrating meta-heuristic feature selection with deep sequence modeling. The model attains nearly flawless accuracy and recall, surpassing current norms. The approach exhibits robustness and scalability by mitigating class imbalance using SMOTE and enhancing feature selection via SHO.

5. Conclusion

The SHO-LSTM is proposed to detect fraudulent activities during credit card transactions. The proposed LSTM model efficiently captures temporal and sequential patterns in transaction data, whereas the based feature selection method

ensures the LSTM model focuses on the most relevant features and uses optimal configurations. By leveraging SHO for feature selection, the model reduces inaccurate results, lowering false alarms and enhancing the detection performance. The details about the credit card transactions of users are obtained from the CCDF benchmark dataset. This study presents a hybrid SHO-LSTM framework for credit card fraud detection, combining metaheuristic feature selection with deep sequence modeling. The model achieves near-perfect accuracy and recall, outperforming existing benchmarks. By addressing class imbalance with SMOTE and optimizing feature selection through SHO, the framework

demonstrates robustness and scalability. In future, advanced DL approaches will be integrated to enhance the efficiency of fraud detection by training these models on diverse and large-scale datasets to improve the models' ability for identifying complex fraudulent patterns.

Conflicts of Interest

The author declares that there is no conflict of interest.

Funding Statement

The author declares that there is no Funding.

References

- [1] Honghao Zhu et al., "NUS: Noisy-Sample-Removed Undersampling Scheme for Imbalanced Classification and Application to Credit Card Fraud Detection," *IEEE Transactions on Computational Social Systems*, vol. 11, no. 2, pp. 1793-1804, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [2] Dijana Jovanovic et al., "Tuning Machine Learning Models using a Group Search Firefly Algorithm for Credit Card Fraud Detection," *Mathematics*, vol. 10, no. 13, pp. 1-30, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [3] Ibomoiye Domor Mienye, and Yanxia Sun, "A Machine Learning Method with Hybrid Feature Selection for Improved Credit Card Fraud Detection," *Applied Sciences*, vol. 13, no. 12, pp. 1-17, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [4] Aya Abd El-Naby, Ezz El-Din Hemdan, and Ayman El-Sayed, "An Efficient Fraud Detection Framework with Credit Card Imbalanced Data in Financial Services," *Multimedia Tools and Applications*, vol. 82, no. 3, pp. 4139-4160, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [5] Daniele Lunghi et al., "An Adversary Model of Fraudsters' Behavior to Improve Oversampling in Credit Card fraud Detection," *IEEE Access*, vol. 11, pp. 136666-136679, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [6] Kun Zhu et al., "An Adaptive Heterogeneous Credit Card Fraud Detection Model based on Deep Reinforcement Training Subset Selection," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 8, pp. 4026-4041, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [7] Shaymaa E. Sorour et al., "Credit Card Fraud Detection using the Brown Bear Optimization Algorithm," *Alexandria Engineering Journal*, vol. 104, pp. 171-192, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [8] Honghao Zhu et al., "A Self-Adapting and Efficient Dandelion Algorithm and its Application to Feature Selection for Credit Card Fraud Detection," *IEEE/CAA Journal of Automatica Sinica*, vol. 11, no. 2, pp. 377-390, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [9] Mustafa Abdul Salam et al., "Federated Learning Model for Credit Card Fraud Detection with data Balancing Techniques," *Neural Computing and Applications*, vol. 36, no. 11, pp. 6231-6256, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [10] Hamed Tabrizchi, and Jafar Razmara, "Credit Card Fraud Detection using Hybridization of Isolation Forest with Grey Wolf Optimizer Algorithm," *Soft Computing*, vol. 28, no. 17, pp. 10215-10233, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [11] Md Amirul Islam et al., "An Ensemble Learning Approach for Anomaly Detection in Credit Card data with Imbalanced and Overlapped Classes," *Journal of Information Security and Applications*, vol. 78, pp. 1-21, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [12] Mohamed Amine MEDDAOUI et al., "Deep Learning Optimization Conception: Less data, Less Time, more Performance," *International Journal of Advanced Computer Science and Applications*, vol. 16, no. 7, pp. 324-332, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [13] Bogdan Milicevic, and Zoran Babovic, "A Systematic Review of Deep Learning Applications in Database Query Execution," *Journal of Big Data*, vol. 11, no. 1, pp. 1-38, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [14] Yu Xie et al., "Learning Transactional Behavioral Representations for Credit Card Fraud Detection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 4, pp. 5735-5748, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [15] Sheo Kumar et al., "Credit Card Fraud Detection using Support Vector Machine," *Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT and Smart Cities Applications (ICMISC 2021)*, Springer, Singapore, vol. 237, pp. 27-37, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [16] N. Prabhakaran, and R. Nedunchelian, "Oppositional Cat Swarm Optimization-based Feature Selection Approach for Credit card Fraud Detection," *Computational Intelligence and Neuroscience*, vol. 2023, no. 1, pp. 1-13, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [17] Tesfahun Berhane et al., "A Hybrid Convolutional Neural Network and Support Vector Machine-based Credit Card Fraud Detection Model," *Mathematical Problems in Engineering*, vol. 2023, no. 1, pp. 1-10, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] T. Karthikeyan, M. Govindarajan, and V. Vijayakumar, "An Effective Fraud Detection using Competitive Swarm Optimization based Deep Neural Network," *Measurement: Sensors*, vol. 27, pp. 1-12, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [19] Lina Ni et al., "Fraud Feature Boosting Mechanism and Spiral Oversampling Balancing Technique for Credit Card Fraud Detection," *IEEE Transactions on Computational Social Systems*, vol. 11, no. 2, pp. 1615-1630, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]

- [20] Zahra Salekshahrezaee, Joffrey L. Leevy, and Taghi M. Khoshgoftaar, "The Effect of Feature Extraction and data Sampling on Credit Card Fraud Detection," *Journal of Big Data*, vol. 10, no. 1, pp. 1-17, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [21] Iman Akour, Nour Mohamed, and Said Salloum, "Hybrid CNN-LSTM with Attention Mechanism for Robust Credit Card Fraud Detection," *IEEE Access*, vol. 13, pp. 114056-114068, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [22] Si Shi, Wuman Luo, and Giovanni Pau, "An Attention-based Balanced Variational Autoencoder Method for Credit Card Fraud Detection," *Applied Soft Computing*, vol. 177, pp. 1-12, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [23] Kaggle, Credit Card Fraud Detection Dataset, Kaggle, 2024. [Online]. Available: <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud>
- [24] Hanen Karamti et al., "Improving Prediction of Cervical Cancer using KNN Imputed SMOTE Features and Multi-Model Ensemble Learning Approach," *Cancers*, vol. 15, no. 17, pp. 1-19, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [25] S. Jhansi Ida et al., "Enhancing Credit Card Fraud Detection Through LSTM-based Sequential Analysis with Early Stopping," *2024 2nd International Conference on Networking and Communications (ICNWC)*, Chennai, India, pp. 1-6, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]