


An improved machine learning technique for credit card fraud detection

 Vinay Kumar Kasula^{1*}, Mounica Yenugula², Akhila Reddy Yadulla³, Bhargavi Konda⁴, Supraja Ayyamgari⁵

^{1,2,3,4,5}Department of Information Technology, University of the Cumberlands, KY, USA; vinaykasula.phd@ieee.org (V.K.K.)
ymounica.phd@ieee.org (M.Y.) akhilareddyadulla@ieee.org (A.R.Y.) bhargavikonda@ieee.org (B.K.)
Sayyamgari32587@ucumberlands.edu (S.A.)

Abstract: Credit Card Fraud (CCF) detection is a major challenge in financial security, especially in detecting unauthorized transactions. As fraudsters' patterns increase, traditional methods face challenges in detecting them. Hence, this paper proposes a Sea Lion - Self-Supervised Network (SL-SSNet) to improve detection accuracy. This research study aims to enhance performance by optimizing data quality feature extraction and achieving better results in fraud detection. The innovative approach for CCF detection using a hybrid optimization model combines the strengths of Sea Lion optimization and Self-Supervised Networks to improve both model accuracy and performance. Initially, the CCF dataset is collected from Kaggle. Then the data goes through a pre-processing phase where irrelevant data points, noise, and low-quality data are removed. The relevant data is selected in the next phase. Feature extraction is performed to select the most important and influential features related to fraud detection. The final phase is prediction and performance evaluation. The results show that the SL-SSNet model performs better than other methods in fraud detection. Specifically, the model achieved 99.98% accuracy, 82.46% precision, 97.23% recall, and 89.97% F1-score. These results prove the effectiveness and robustness of SL-SSNet in detecting fraudulent transactions.

Keywords: CCF detection, Feature extraction, Pre-processing, Sea Lion optimization, Self Supervised Network.

1. Introduction

People will increasingly rely on Internet transactions as the globe moves toward a cashless future. The presence of fraudsters at crime sites is no longer necessary [1]. They can hide their identities in a number of ways while committing their heinous crimes in the convenience of their own homes [2]. These identity-concealing methods, which include utilizing a VPN and directing the victim's communications over the Tor network, are difficult to detect [3]. Online financial losses have a significant impact that should not be understated. Once card data are stolen, fraudsters may use or sell the cards to others [4]. This is the situation in India, where the dark web sells the card details of almost 70 million individuals [5]. One of the worst CCF cases in the USA recently led to losses of GBP 17 million. In the mid-2000s, a gang of international scammers banded together to steal the credit card details of over 32,000 people, which led to the event [6]. Many people believe that this is the largest card scam in history. Thus, when security measures are ineffective, billions of dollars are lost due to credit card theft [7]. Cardholders are comforted that all transactions are harmless while they use their cards, and card issuers are reassured throughout transaction processing [8]. Fraudsters, conversely, want to trick cardholders and financial institutions into thinking that the fraudulent transactions are authentic [9].

Additionally, some fraudulent transactions are perpetuated to profit financially without the cardholders' or card issuers' awareness [10]. The worst part about using a credit card is this: It's possible that cardholders and approved institutions are unaware of fraudulent transactions all the time

[11]. As a result, it is extremely difficult to identify fraudulent activity amid thousands of normal transactions, particularly when the percentage of fraudulent transactions is much lower compared to the number of legitimate transactions [12]. The financial sector may employ data mining and predictive analytics, particularly modelling algorithms that employ anomaly detection and clustering procedures, as fraud detection strategies [13]. All of these tactics need the use of supervised and unsupervised ML techniques, which are helpful in categorizing CCF [14]. However, when such machine learning methods attempt to identify every instance of Fraud, they run into infinite problems.

The optimal machine learning model requires the highest values of the regularly used evaluation metrics. Numerous changes are needed in this area in order to approach this ideal model [15]. Detecting CCF is difficult and depends on several criteria, including resampling, cross-validation, and ML algorithms. The model's performance can be improved by considering these elements, and the assessment metrics can confirm this [16]. Balanced datasets to tackle real-world problems are scarce. Therefore, the classification approach often minimizes the significance of the dataset's minority class [17]. The most important categorization class, particularly for detecting CCF, is the minority class [18]. Following the selection of the optimal ML algorithms, the suggested method employs a variety of resampling strategies to highlight the imbalance class issue caused by the dataset's unequal class distribution [19, 20]. This research considers both resampling and enhanced cross-validation (CV) approaches. Key contribution to this study is discussed as flows,

- The CCF data is collected and implemented on the Python platform.
- The SL-SSNet model is developed with the predictive features to enhance model accuracy
- It is performed by refining the input data during pre-processing and selecting key fraud-related features in feature extraction
- Hence the fraudulent transactions are predicted and classified.
- The performance was evaluated and compared to traditional methods.

The study examines CCF detection Section 2 discusses the literature on previously published works. The methodology of this investigation is covered in Section 3. Results and Discussion are included in Section 4. Conclusion is covered in Section 5

2. Literature Review

Some of the recent literature is defined as follows.

Alarfaj, et al. [20] and Benchaji, et al. [21] developed experiments to identify such frauds, including public data accessibility, statistics on the disparity in wealth, shifts in the type of Fraud. Numerous ML based techniques for credit card recognition are presented, including the ELM, DT, RF, SVM, LR, and XG Boost. Modern DL algorithms are still required to reduce fraud losses, even if they are not very accurate. For this, the most current advancements in deep learning algorithms have been the primary emphasis.

Benchaji, et al. [21] and Karthik, et al. [22] developed a novel approach to detecting CCF that sequentially represents data using LSTM deep recurrent neural networks and attention processes. As opposed to earlier research, the proposed method considers the sequential structure of transactional data and enables the classifier to determine which transactions in the input sequence are the most significant and which have the highest accuracy in detecting fraudulent transactions.

Karthik, et al. [22] and Berhane, et al. [23] proposed a new model for detecting CCF that employs ensemble learning strategies like bagging and boosting. Our model uses the key elements of both bagging and boosting ensemble classifiers to create a hybrid model. Tests using data from Brazilian banks and UCSD-FICO demonstrate that our model is more robust than the most advanced ones in identifying hidden fraudulent transactions because a hybrid approach was used.

Berhane, et al. [23] and Abdul Salam, et al. [24] created a method for detecting CCF using a CNN-SVM hybrid model. They used real credit card transaction data from the public to assess how well our proposed hybrid CNN-SVM model identified CCF. Their hybrid CNN-SVM model was created by

substituting an SVM classifier for the CNN model's final output layer. A support vector machine is stacked after the first classifier, followed by a fully connected layer with a learnt end-to-end softmax. In contrast, the second classifier is created by deleting the final fully linked softmax layer.

Abdul Salam, et al. [24] and Malik, et al. [25] proposed multiple frameworks for CCFD federated learning. Across all institutions, there is a discernible disparity in credit card transactions, and a very small proportion of fraudulent transactions exceed the majority of legitimate ones. To demonstrate the urgent need to analyze class imbalance management techniques in detail in order to develop a powerful model to identify fraudulent transactions, the dataset has to be balanced.

Malik, et al. [25] and Ileberi, et al. [26] proposed that seven hybrid machine-learning algorithms use a real-world word dataset to detect fraudulent behaviour. The hybrid models were developed using contemporary ML approaches to detect CCF. From the first step, the best single algorithm was then used to build hybrid methods. Ileberi, et al. [26] and Nguyen, et al. [27] implemented an ML system for employing distorted real-world datasets generated by European credit card customers to identify CCF. They resampled the dataset using the SMOTE to address the class imbalance problem.

Nguyen, et al. [27] and Jiang, et al. [28] conducted Vesta Corporation's IEEE-CIS Fraud Detection Dataset. Through the use of labelling logic, they steer the research process to forecast fraudulent credit cards rather than fraudulent occurrences by setting the whole account to "Fraud=1" once the credit card has been unlawfully charged. Jiang, et al. [28] and Mienye and Sun [29] experimented with identifying credit card theft, which have been extensively researched using classic ML techniques. However, these techniques frequently struggle to prove their efficacy despite unidentified attack patterns.

Mienye and Sun [29] and Alfaiz and Fati [30] proposed an MLP that serves as the meta-learner in this resilient deep learning method; this employs a stacking ensemble design using LSTM and GRU neural networks as foundation learners. This hybrid SMOTE-ENN approach balances the dataset's class distribution.

2.1. Problem Statement

A variety of difficulties are examined in the study on the development of machine learning algorithms for CCF detection, including the requirement for real-time detection with fewer FPs, the difficulty of balancing fraudulent and legitimate transactions for effective detection, and the complexity of some types of Fraud. Existing systems cannot usually learn, adapt, or adjust to ever-changing fraud technologies, so their detection tends to be delayed or inaccurate. Furthermore, because the transaction records consist of many variables and changing user behaviors, the same features tend to make traditional models less effective. This study is intended to develop highly adaptive, accurate, and efficient algorithms capable of learning evolving patterns of Fraud and available with little disruption for legitimate users. The Problem statement is displayed in Figure 1.

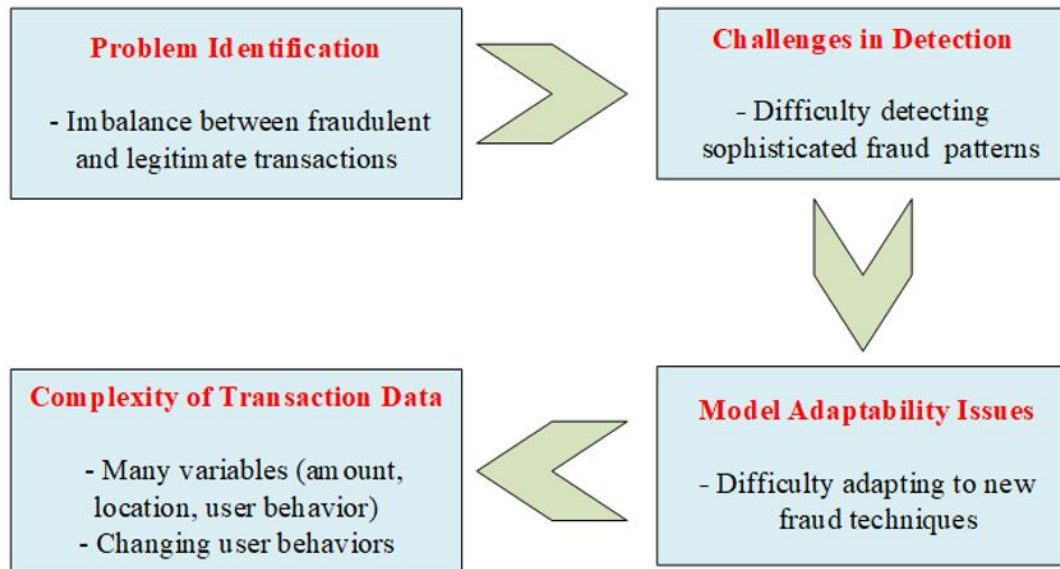


Figure 1.
Problem statement.

3. Proposed Methodology

The proposed method begins with input data followed by hybrid optimization of Sea Lion - Self Supervised Network (SL-SSNet) Optimisation. This hybrid has the advantages of Sea Lion and Self-Supervised Networks to boost model accuracy and performance. The data points remaining after hybrid optimization then enter into a pre-processing phase where irrelevant data points, noise points, and low-quality data are discarded so that only clean and efficient information is acknowledged for the next analysis phase. The next stage is feature extraction, where selected and extracted features are the most important in fraud detection. In order to help the system distinguish between authentic and fraudulent transactions, the dataset's dimensionality should be decreased while maintaining the necessary information. The Proposed Architecture is displayed in Figure 2.

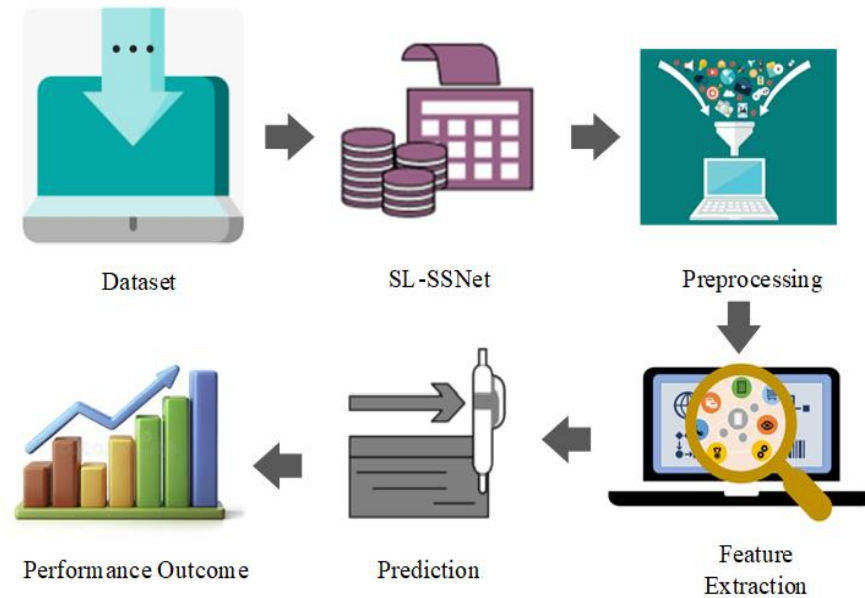


Figure 2.
Proposed Architecture.

3.1. Proposed SL-SSNet

The phase of exploration can include peculiarities in the Methodology by integrating Contrastive Loss in Self-Supervised Learning (SSL) with Sea Lion, a very powerful tool or framework. This work is a method that learns representations through contrasting positive and negative samples, guiding the model in understanding the contrasts between normal and fraudulent transactions. It is to employ SSL so that the system can access unlabelled data and, hence, requires reduced amounts of labeled fraud data. Sea Lion potentially works as a development tool in model training or data analysis, enhancing the approach through functionalities to optimize model accuracy and generalization. This might increase the efficacy of the whole CCF detection system by making it possible to detect fraud patterns more successfully, especially in dynamic and extremely complex datasets. The Equation (1) displays the hybrid Equation.

$$Y^{(g+1)} = Y_{rand}^g - C | 2rX_{rand}^3 - X^3 | + (-\log \left(-\log \frac{\exp(z_i \cdot z_j / T)}{\sum_{K=1}^N \exp(z_i \cdot z_k / T)} \right)) \quad (1)$$

C depends on behaviour where Y_{rand}^t is a random sea lion randomly selected from the current population. r denoted random value in the range $[0, 1]$. This is typically used in self-supervised contrastive learning, where z_i, z_j represent embedding of data points, and T is a temperature parameter controlling the sharpness of the softmax distribution.

3.1.1. Pre-processing

The data points will go under a very important pre-processing phase. It will tend to remove all the irrelevant, noisy data, or low-quality data that will prejudice the Result of the following analysis. These data points were deleted with the intention of preserving just the essential, practical, and efficient information needed to increase the fraud detection system's accuracy and efficacy. In general, the pre-processing followed by missing value treatment, data normalization, outlier removal, and data

formatting will prepare the data in a way that is usable by the algorithm for machine learning purposes. This is the step that provides quality raw data to be used in the analysis phase, which directly affects the efficiency of detecting fraudulent transactions. The data initialization is shown in Equation (2).

$$D_i = X_{1,2,23,\dots,n} \quad (2)$$

The data initialization is denoted as D_i , the credit card data is denoted as X and the pre-processing of Normalisation, Standardisation, and Handling missing values is shown in Equation (3).

$$X_{processed} = \frac{X(\mu) - X_{\min}}{X_{\max} - X_{\min}} \times \frac{1}{\sigma} \quad (3)$$

X_{\min} and X_{\max} being the minimum and maximum values of a feature, respectively. While μ and σ are the mean and standard deviation of a feature, used for standardization and handling missing values that need to be imputed using methods like the mean or median to complete the dataset.

3.1.2. Feature Extraction

Finding and choosing the necessary features from a sizable amount of raw data while lowering the data's dimensionality without sacrificing the most important details that might determine whether a transaction is authentic or fraudulent is known as feature extraction in fraud detection. In order to prevent repetition resulting from spotting fraud with irrelevant data, the focus is on criteria such transaction amount, frequency, location, and user behaviour patterns. Both accuracy and training time are improved, as is the fraud detection model's overall performance and efficiency. The feature selection is performed in Equation (4).

$$X_{features} = X_{processed} \cdot W(F) \quad (4)$$

$X_{features}$ denotes the selected features, $X_{processed}$ projected onto a lower-dimensional space, while W is the matrix of eigenvectors representing the directions of maximum variance for the data and the relevant features denotes as F

3.1.3. Prediction

Prediction of Fraud in credit cards is based on the identification of anomalies through monitoring the selected features with unusual patterns or behavior. The Sea Lion Fitness algorithm is implemented in order to optimize this process and enhance the misgivings in the detection of the anomaly by analyzing improved features and the region of the trail of the sea lion observed. The program would change settings to increase how well it could distinguish between a legitimate transaction and a fraudulent one, which increases the accuracy meant for detection. The model's Sea Lion Fitness technique improves the detection of anomalies and odd trends in transaction data; it increases credit-card fraud detection systems' accuracy and reliability. Equation (5) performs the prediction.

$$I_p = \ln \left(\frac{P \cdot X_{features}}{1 - P} \right) \quad (5)$$

I_p denotes the credit card prediction variables P is the probability of the transaction being fraudulent and it's classification is established in Equation (6).

$$C = \begin{cases} \text{If } (I_p = 0) & \text{Nonfraud} \\ f(I_p = 1) & \text{fraud} \end{cases} \quad (6)$$

Here C denoted the classification Variable. The classification is based on non-fraud and Fraud. The proposed SL-SSNet model in instruction prediction demonstrates robustness, efficiency, and effectiveness. The entire workflow is shown in Figure 3.

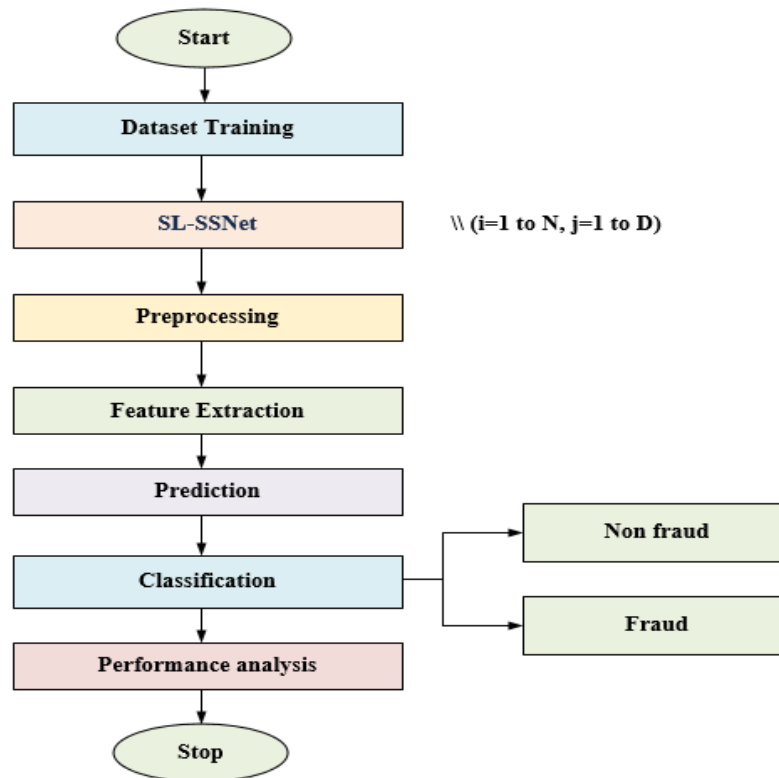


Figure 3.
Flow chart of the developed SL-SSNet.

SL-SSNet Pseudo code

Start

{

Initialization

\\ dataset initialization

Data Pre-processing

{

Normalize the data

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

Standardize the data

$$X_{std} = \frac{X - \mu}{\sigma}$$

Handle missing values

$$X_{missing} = \mu$$

}

```

Feature Extraction
{
  Extracting relevant features ():
  features = Xprocessed, W
}
Prediction
{
  Prediction = trained model predict ():
  Prediction = Xnew, trained model
  return prediction
Classification Based on Prediction
classify prediction (Ip):
  Ip = 0
  return "Non-fraud"
  Ip = 1
  return "fraud"
}
}
Stop

```

4. Results

Table 1 displays the parameters that have been established for this investigation. The operating system is Windows 10, which is a stable and modern system that runs applications. Python was selected for the required programming environment since it is a very strong and flexible programming language. Python has been mainly used for scientific computing and algorithm development. Python 3.7.14 was especially applicable for this study to allow proper interaction with various libraries and frameworks executing the algorithm. The model in use is the Sea Lion-Self-Supervised Network (SLSN), which is an extremely complex machine-learning framework that leverages the Sea Lion optimization algorithm alongside self-supervised learning techniques. These unique trajectories allow the model to learn from unlabelled data and facilitate improvement in other tasks.

Table 1.
Parameters Execution.

Metrics	Specification
Operating System	Windows 10
Program platform	Python
Version	3.7.14
Optimization	Sea Lion Optimization
Machine Learning	Self-Supervised Network

4.1. Case Study

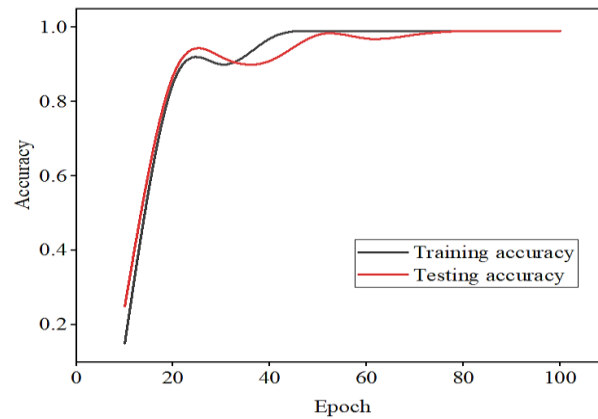
This dataset is from the Kaggle repository. It contains a total of 284,807 entries. The database further compartmentalizes the information into two categories: Fraud and no-fraud transactions. Of the total number of samples, 284,315 samples are labeled as non-fraud, while 492 samples are cases of Fraud. To verify the models performance in prediction the dataset is split at an 80:20 ratio. 80% is robbed for non-fraudulent transactions, while the remaining 20% is made up of fraudulent transactions.

Table 2.

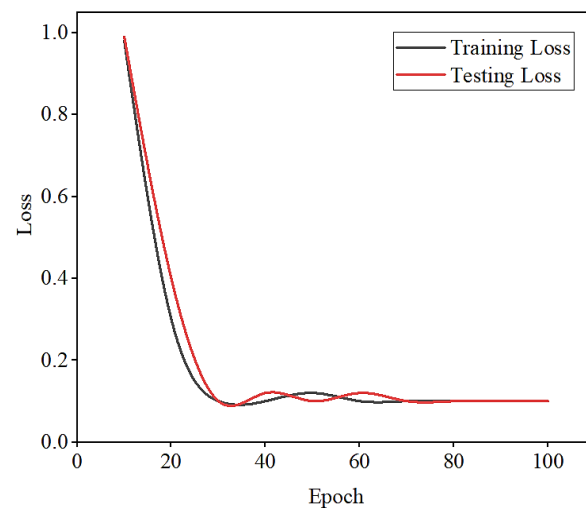
Division of dataset ratio 80:20.

Total samples =284807 (100%)	
Non Fraud	2,84,315
Fraud	492
Training = 227845 (80%)	
Non Fraud	2,27,452
Fraud	393
Testing=56962(20%)	
Non Fraud	56,863
Fraud	99

There are 393 instances of Fraud and 227,452 non-fraud samples in the training set, which makes up 80% of the total. As for the testing set, it contains 56,962 samples that is 20% of the total instances. Of these 99 are fraud cases and 56,863 of which are non-fraud case samples. Such a representation ensures a larger proportion of non-fraud cases in both training and testing datasets while keeping the fraud cases as a minority, thus maintaining an imbalance between the two classes.



(A)



(B)

Figure 4.

(A) Training and testing accuracy and (B) Training and testing Loss.

The proposed models accuracy and loss graph during training and testing for 100 epochs is shown in Figure 4. The confusion matrix is provided in Figure 5.

The proposed SL-SSNet framework properly predicts CCF detection according to the confusion matrix shown in Figure 5. TP, TN, FP, and FN situations are used to classify it. Both real positive and negative outcomes indicate the precisely predicted detection of Fraud i.e., 56859 instances as non-fraud and 98 instances as fraud. Similarly, FPs and negatives show that the detection and normal cases were not accurately predicted, i.e, 4 non-fraud instances misclassified as fraud and 1 fraud instance misclassified as Non-fraud.

Non-Fraud	56859	4
Fraud	1	98
	Non- Fraud	Fraud

Figure 5.
Confusion matrix for CCF detection.

4.2. Performance Evaluation

The effective function of the developed SL-SSNet model for predicting credit fraud detection cards is valid using the CCF detection Dataset and Python program. To evaluate the model's performance using measures for fraud detection, including F1 score, accuracy, Recall, and precession. A few current approaches are contrasted with the suggested strategy in order to assess the usefulness of the proposed framework. The Accuracy, Precession, Recall, and F1- scores are compared with LR, KNN, DT, NB, RF, GBM, Light GBM (LGBM), XG Boost (XGB), and Cat Boost (CB).

4.2.1. Accuracy

Error rate, often known as accuracy, is a measure of how often a classifier correctly classifies a piece of data Equation (7) displays the accuracy, which is determined by dividing the total number of occurrences by the number of false positives (TP) and false negatives (TN) that were properly identified.

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (7)$$

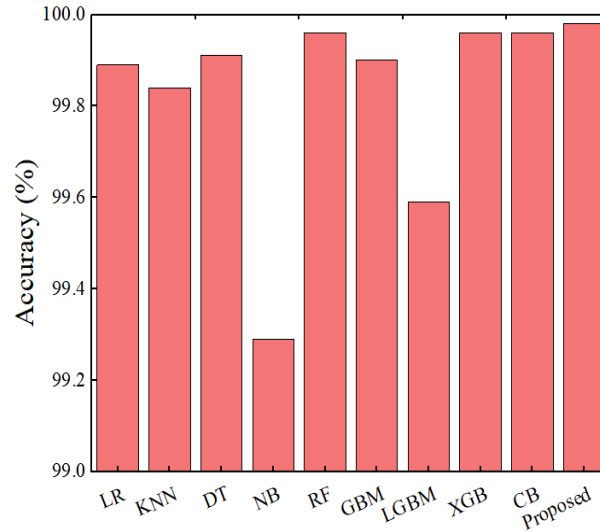


Figure 6.
Accuracy comparison graph.

Figure 6 shows the percentage accuracies of the various classifiers express the performance of the classifiers on a given task. Logistic Regression achieved an accuracy rate of 99.89%, while KNN was slightly lower at 99.84%. Decision Tree is rated at 99.91%, its accuracy being marginally better than that of Naive Bayes, whose accuracy was rated at 99.29%. Models such as Radio Frequency, XG Boost, CatBoost, and proposed SL-SSNet have given a noteworthy status to their performance through accuracy rates, with SL-SSNet clearly outperforming all others at 99.98%. GBM and Light GBN are also quite competitive at accuracies of 99.90% and 99.59%, respectively.

4.2.2. Precision

Evaluation parameters called Precision and Recall operate differently and produce distinct outcomes. Precision and Recall are frequently traded off. Recall decreases with increasing Precision and increases with decreasing Precision. Equation (8) indicates that the Positive Predictive Value, or simply the Positive Predictive Value, assesses how accurate the forecasts were for the positive occurrences out of all the positive cases.

$$\text{Precision} = \text{Postive predicted value} = \frac{TP}{TP + FP} \quad (8)$$

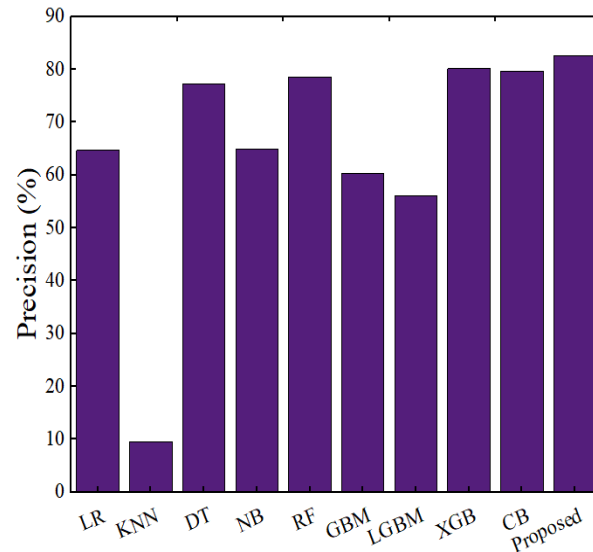


Figure 7.
Precision comparison graph.

The percentage of Precision of the various classifiers when performing a particular task is shown in Figure 7. Out of the classifiers mentioned, the Proposed Model (SL-SSNet) stands high above any other classifier, recording the top statistics of 82.46%, thus implying that the model exhibits good capability in predicting correct results. Next, in order, are the XG Boost and Catboost models, at 80.08% and 79.67% respectively. Following these is Radio Frequency at 78.46%. The Decision Tree, Naive Bayes, and Logistic Regression classifiers perform reasonably, respectively, scoring 77.24%, 64.85%, and 64.62% accuracy; however, these three do not compare to the top three mentioned above. The KNN (9.56%) and the Light GBN (56.12%) feature remarkably below expectations.

4.2.3. Recall

In order to identify fraudulent credit card transactions, recall also referred to as sensitivity and TP rate (TPR) is one of the most crucial assessment criteria. Its ability to recognize affirmative cases determines its relevance. Fraudulent conduct is detected more frequently when the recall value is higher. In order to prevent any instances of Fraud from being missed, it is crucial to acquire a greater recall value as much as possible. While the suggested model could achieve a respectable level of FP, it should not come at the price of FN to the greatest extent feasible. Equation (9) represents Recall:

$$Recall = \frac{TP}{TP + FP} \quad (9)$$

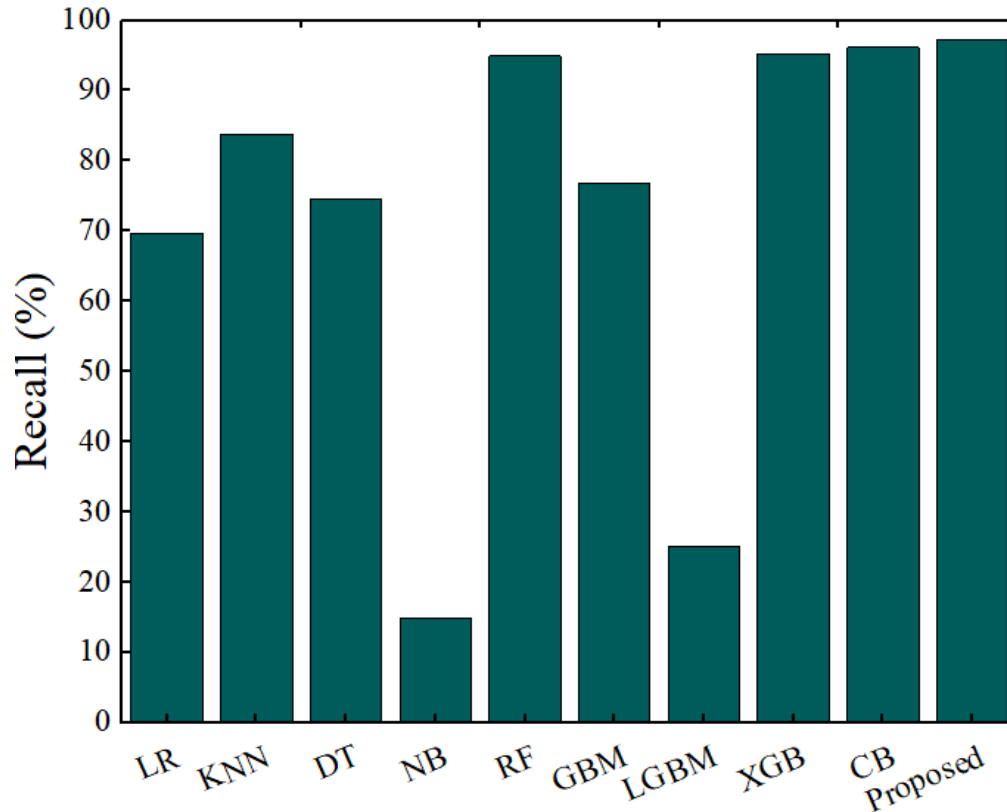


Figure 8.
Recall comparison graph.

This proposed SL-SSNet classifier stands out among all models by being a class apart with a recall value of 97.23%, showing its ability to detect positive instances. Other good models are Catboost and XGBoost, with recall values of 96.12% and 95.23%. Radio Frequency KNN can then be followed with recalls of 94.87% and 83.75%. Models such as Naive Bayes and LightGBM measure much lower Recall at 14.78 and 24.99, respectively is displayed in Figure 8. This implies that SL-SSNet proves to be quite promising in gathering relevant patterns in the data rather than traditional classifiers.

4.2.4. F1-Score

Recall has a more important role in detecting CCF than Precision. To evaluate the model's performance, the F1-Score, on the other hand, combines the Precision and Recall values. Among other assessment metrics, the F1-Score is taken into account when comparing two or more models; hence, the classifier with the highest F1-Score. The expression for the F1-Score is Equation (10).

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (10)$$

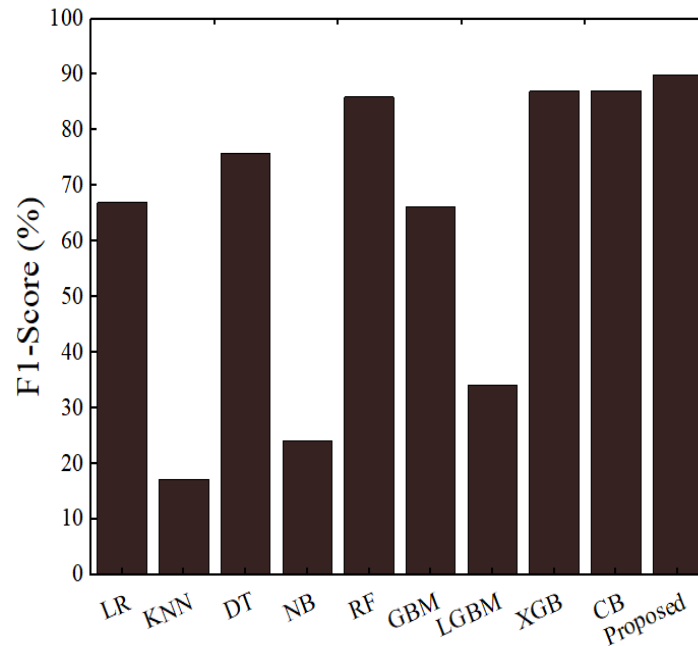


Figure 9.
F1-Score.

Figure 9 displays that SL-SSNet model boasts the highest F1 score, 89.97%, showing a good balance between Precision and Recall. Next would be Catboost and XGBoost, which deliver good average performances with F1 Scores of 87.11% and 86.98%, respectively. Radio Frequency follows closely next at 85.88%, while Decision Tree and Logistic Regression put up fair performances at only 75.81% and 66.88%, respectively.

Table 3.
Entire comparison.

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
LR	99.89	64.62	69.71	66.88
KNN	99.84	9.56	83.75	17.11
DT	99.91	77.24	74.49	75.81
NB	99.29	64.85	14.78	24.05
RF	99.96	78.46	94.87	85.88
GBM	99.90	60.34	76.88	66.15
LGBM	99.59	56.12	24.99	34.10
XGB	99.96	80.08	95.23	86.98
CB	99.96	79.67	96.12	87.11
Proposed (SL-SSNet)	99.98	82.46	97.23	89.97

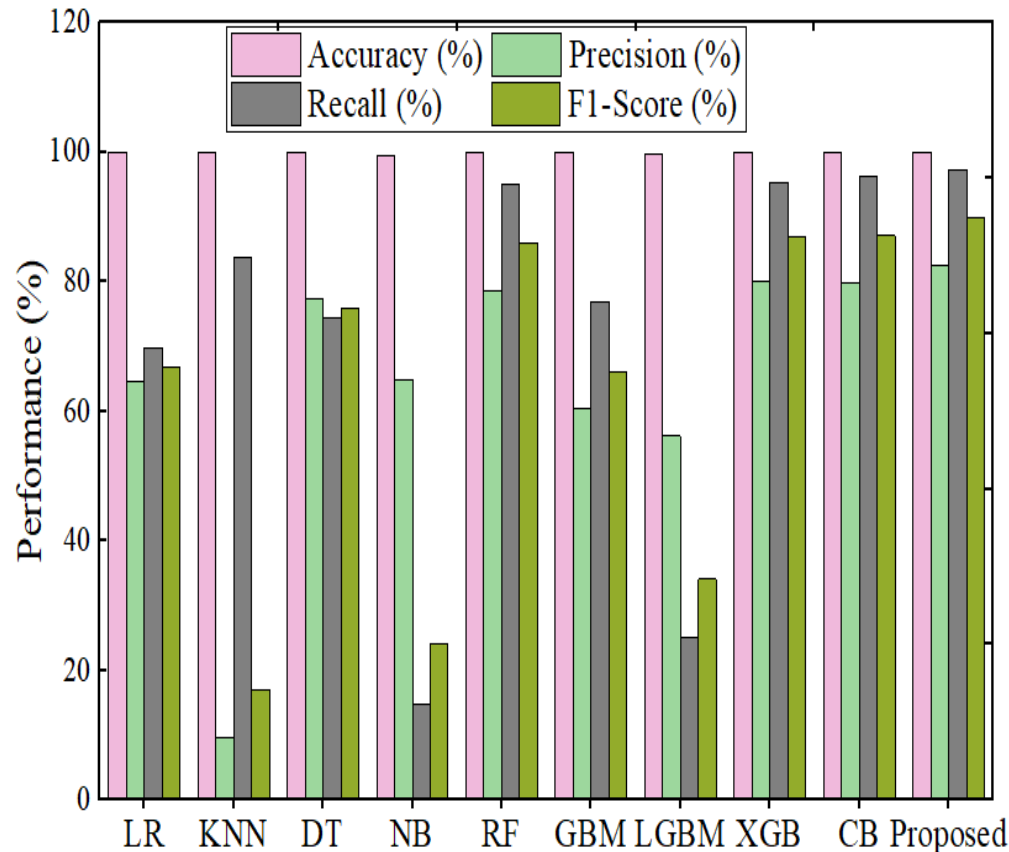


Figure 10.
Overall Comparison graph.

The worst performances include KNN and Naive Bayes, which have F1 Scores of 17.11 and 24.05, respectively, to emphasize their poor ability to balance FPs with false negatives. So, it follows that SL-SSNet is best in maintaining precision-recall balance for this classification. The overall comparison of the developed framework with the prevailing model is depicted in Table 3 and Figure 10.

4.3. Discussion

SL-SSNet was proven to outperform the exclamation point in CCF prediction. Accordingly, with this proposed model, using the ability of Sea Lion to select and track its optimal features from this model, machine learning employs prediction on whether a credit card transaction is fraudulent or not based on these optimized features. Collectively, the model attains high Accuracy, Precision, Recall, and F1 scores, thereby showcasing the effectiveness of the model in wider parameters. The results of full performance in detail, along with the developed model, are in Table 4.

Table 4.
Performance of SL-SSNet.

Metrics	Performance
Accuracy	99.98
Precision	82.46
Recall	97.23
F1-Score	89.97

5. Conclusion

In conclusion, the hybrid optimization model of SL-SSNet has shown considerable improvement in detecting CCF by using Sea Lion Optimization and Self-Supervised Networks. Hence the data collected is refined and the features are selected by the sea lion optimization. With the selected features the prediction and classification is performed. The systematic process of quality enhancement and selecting features brings about a model achievement of 99.98% accuracy, 82.46% precision, 97.23% recall, and 89.97% F1-score-for CCF detection. These perform extremely well, and metric values indicate effectiveness and robustness in creating this model as a promising tool for detecting fraudulent transactions. Another major possibility of SL-SSNet is its applicability in the real world, as it promises to provide an efficient means of combating the menace of CCF.

Transparency:

The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.

Copyright:

© 2025 by the authors. This open-access article is distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

References

- [1] I. Mekterović, M. Karan, D. Pintar, and L. Brkić, "Credit card fraud detection in card-not-present transactions: Where to invest?," *Applied Sciences*, vol. 11, no. 15, p. 6766, 2021. <https://doi.org/10.3390/app11156766>
- [2] D.-G. Beju and C.-M. Făt, *Frauds in banking system: Frauds with cards and their associated services* (Economic and financial crime, sustainability and good governance). Cham: Springer, 2023, pp. 31–52.
- [3] L. Razaq, T. Ahmad, S. Ibtasam, U. Ramzan, and S. Mare, "We even borrowed money from our neighbor" understanding mobile-based frauds through victims' experiences," *Proceedings of the ACM on human-computer interaction*, vol. 5, no. CSCW1, pp. 1–30, 2021.
- [4] J. Besenyő and A. Gulyas, "The effect of the dark web on the security," *Journal of Security & Sustainability Issues*, vol. 11, no. 1, pp. 1–19, 2021.
- [5] Y. Takefuji, "Case report on enormous economic losses caused by fraud from Japan to the world," *Journal of Economic Criminology*, vol. 1, p. 100003, 2023. <https://doi.org/10.1016/j.jeconc.2023.100003>
- [6] A. Gupta, M. Lohani, and M. Manchanda, "Utilizing mathematical concepts of heat map for an intelligent and secure approach to efficiently detect credit card fraud," *Journal of Interdisciplinary Mathematics*, vol. 26, no. 8, pp. 1837–1854, 2023. <https://doi.org/10.47974/JIM-1761>
- [7] M. Garba, "Adoption of information and communications technology for cashless economy in Nigerian banking sector," *Kaduna Journal of Postgraduate Research*, vol. 2, no. 2, pp. 1–13, 2019.
- [8] E. Strelcenia and S. Prakoonwit, "Improving classification performance in credit card fraud detection by using new data augmentation," *AI*, vol. 4, no. 1, pp. 172–198, 2023. <https://doi.org/10.3390/ai4010008>
- [9] Z. Morić, V. Dakic, D. Djekic, and D. Regvart, "Protection of personal data in the context of e-commerce," *Journal of Cybersecurity and Privacy*, vol. 4, no. 3, pp. 731–761, 2024. <https://doi.org/10.3390/jcp4030034>
- [10] S. Xiang, G. Zhang, D. Cheng, and Y. Zhang, "Enhancing attribute-driven fraud detection with risk-aware graph representation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 37, no. 5, pp. 2501–2512, 2025. <https://doi.org/10.1109/TKDE.2025.3543887>
- [11] S. K. Hashemi, S. L. Mirtaheri, and S. Greco, "Fraud detection in banking data by machine learning techniques," *Ieee Access*, vol. 11, pp. 3034–3043, 2022. <https://doi.org/10.1109/ACCESS.2022.3232287>
- [12] A. K. Lin, "The AI revolution in financial services: Emerging methods for fraud detection and prevention," *Jurnal Galaksi*, vol. 1, no. 1, pp. 43–51, 2024.
- [13] J. K. Afriyie *et al.*, "A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions," *Decision Analytics Journal*, vol. 6, p. 100163, 2023. <https://doi.org/10.1016/j.dajour.2023.100163>
- [14] B. Lebichot, T. Verhelst, Y.-A. Le Borgne, L. He-Guelton, F. Oble, and G. Bontempi, "Transfer learning strategies for credit card fraud detection," *IEEE Access*, vol. 9, pp. 114754–114766, 2021.
- [15] M. K. R. Mallidi and Y. Zagabathuni, "Analysis of credit card fraud detection using machine learning models on balanced and imbalanced datasets," *International Journal of Emerging Trends in Engineering Research*, vol. 9, no. 7, pp. 846–852, 2021. <https://doi.org/10.30534/ijeter/2021/02972021>

- [16] O. A. Bello, A. Folorunso, O. E. Ejiofor, F. Z. Budale, K. Adebayo, and O. A. Babatunde, "Machine learning approaches for enhancing fraud prevention in financial transactions," *International Journal of Management Technology*, vol. 10, no. 1, pp. 85-108, 2023. <https://doi.org/10.37745/ijmt.2013/vol10n185109>
- [17] R. K. Kennedy, F. Villanustre, T. M. Khoshgoftaar, and Z. Salekshahrezaee, "Synthesizing class labels for highly imbalanced credit card fraud detection data," *Journal of Big Data*, vol. 11, no. 1, p. 38, 2024. <https://doi.org/10.1186/s40537-024-00897-7>
- [18] E. Ileberi, Y. Sun, and Z. Wang, "A machine learning based credit card fraud detection using the GA algorithm for feature selection," *Journal of Big Data*, vol. 9, no. 1, p. 24, 2022. <https://doi.org/10.1186/s40537-022-00573-8>
- [19] D. Jovanovic, M. Antonijevic, M. Stankovic, M. Zivkovic, M. Tanaskovic, and N. Bacanin, "Tuning machine learning models using a group search firefly algorithm for credit card fraud detection," *Mathematics*, vol. 10, no. 13, p. 2272, 2022. <https://doi.org/10.3390/math10132272>
- [20] F. K. Alarfaj, I. Malik, H. U. Khan, N. Almusallam, M. Ramzan, and M. Ahmed, "Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms," *Ieee Access*, vol. 10, pp. 39700-39715, 2022. <https://doi.org/10.1109/ACCESS.2022.3166891>
- [21] I. Benchaji, S. Douzi, B. El Ouahidi, and J. Jaafari, "Enhanced credit card fraud detection based on attention mechanism and LSTM deep model," *Journal of Big Data*, vol. 8, pp. 1-21, 2021. <https://doi.org/10.1186/s40537-021-00499-1>
- [22] V. S. Karthik, A. Mishra, and U. S. Reddy, "CCF detection by modeling behavior patterns using a hybrid ensemble model," *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 1987-1997, 2022. <https://doi.org/10.1007/s13369-021-06038-3>
- [23] T. Berhane, T. Melese, A. Walelign, and A. Mohammed, "A hybrid convolutional neural network and support vector machine-based credit card fraud detection model," *Mathematical Problems in Engineering*, p. 8134627, 2023. <https://doi.org/10.1155/2023/8134627>
- [24] M. Abdul Salam, K. M. Fouad, D. L. Elbably, and S. M. Elsayed, "Federated learning model for credit card fraud detection with data balancing techniques," *Neural Computing and Applications*, vol. 36, no. 11, pp. 6231-6256, 2024. <https://doi.org/10.1007/s00521-023-09410-2>
- [25] E. F. Malik, K. W. Khaw, B. Belaton, W. P. Wong, and X. Chew, "Credit card fraud detection using a new hybrid machine learning architecture," *Mathematics*, vol. 10, no. 9, p. 1480, 2022. <https://doi.org/10.3390/math10091480>
- [26] E. Ileberi, Y. Sun, and Z. Wang, "Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost," *IEEE Access*, vol. 9, pp. 165286-165294, 2021. <https://doi.org/10.1109/ACCESS.2021.3059119>
- [27] N. Nguyen *et al.*, "A proposed model for card fraud detection based on Catboost and deep neural network," *IEEE Access*, vol. 10, pp. 96852-96861, 2022. <https://doi.org/10.1109/ACCESS.2022.3201234>
- [28] S. Jiang, R. Dong, J. Wang, and M. Xia, "Credit card fraud detection based on unsupervised attentional anomaly detection network," *Systems*, vol. 11, no. 6, p. 305, 2023. <https://doi.org/10.3390/systems11060305>
- [29] I. D. Mienye and Y. Sun, "A deep learning ensemble with data resampling for credit card fraud detection," *IEEE Access*, vol. 11, pp. 30628-30638, 2023. <https://doi.org/10.1109/ACCESS.2023.3262020>
- [30] N. S. Alfaiz and S. M. Fati, "Enhanced credit card fraud detection model using machine learning," *Electronics*, vol. 11, no. 4, p. 662, 2022. <https://doi.org/10.3390/electronics11040662>