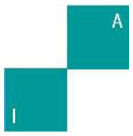
Inteligencia Artificial 26(72) (2023), 202-222
doi: 10.4114/intartif.vol26iss72pp202-222



INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

An intelligent approach for anomaly detection in credit card data using bat optimization algorithm

Haseena S [1], Saroja S [2], Suseandhiran N [3], Manikandan B [4]

[1,3,4] Department of Information Technology, Mepco Schlenk Engineering College, India
[2]Department of Computer Applications, National Institute of Technology, Trichy, India
[2]activeroja@gmail.com

Abstract. As technology advances, many people are utilising credit cards to purchase their necessities, and the number of credit card scams is increasing tremendously. However, illegal card transactions have been on the rise, costing financial institutions millions of dollars each year. The development of efficient fraud detection techniques is critical in reducing these deficits, but it is difficult due to the extremely unbalanced nature of most credit card datasets. As compared to conventional fraud detection methods, the proposed method will help in automatically detecting fraud, identifying hidden correlations in data and reducing the time for the verification process. This is achieved by selecting relevant and unique features by using Bat Optimization Algorithm (BOA). Next, balancing is performed in the highly imbalanced credit card fraud dataset using the Synthetic Minority over-sampling technique (SMOTE). Then finally the CNN model for anomaly detection in credit card data is built using a full center loss function to improve fraud detection performance and stability. The proposed model is tested with the Kaggle dataset and yields around 99% accuracy.

Keywords: Credit card anomaly detection, imbalanced data, feature selection, optimization, neural network, loss function

1 Introduction

Credit card fraud happens when a scammer uses your credit card number and pin, or a stolen credit card, to make financial transactions on your account without your permission. Any individual, who uses a credit card that has been revoked, cancelled, reported lost, or stolen with the purpose to deceive to gain something of value is committing credit card fraud. Credit card fraud can also be committed by using a credit card number without actually having the card. Because it is used in conjunction with identity theft, stealing a person's identity to obtain a credit card is a more dangerous type of credit card fraud. The problem of credit card theft impacts the whole consumer credit business. It's one of the most prevalent sorts of fraud, as well as one of the most difficult tasks to detect [1].

Credit card fraud is a type of identity theft that involves the unlawful use of another person's credit card information to make transactions or withdraw money from their account. When someone, a fraudster or thief, uses your stolen credit card or the information from that card to make unlawful purchases in your name or take out cash advances on your account, this is known as credit card fraud. There are two types of credit card fraud: application fraud and transaction fraud [2]. Application fraud is related to identity fraud in that it occurs when someone obtains a new card by using the personal information of another individual [3]. When a card is stolen or a lost card is obtained to make fraudulent transactions, this is known as transaction fraud [4].

To address this issue, we need a system that can stop a transaction if it detects something suspicious. This necessitates the development of a system that can track the pattern of all transactions and, if any pattern is aberrant, abort the transaction. We now have a plethora of machine learning techniques that can assist us in classifying unusual transactions. The only requirements are historical data and an appropriate algorithm that can better fit our data [5].

Any credit card fraud detection system's main goal is to identify suspicious events and report them to an analyst while allowing normal transactions to be handled automatically [6]. Financial institutions have been

ISSN: 1137-3601 (print), 1988-3064 (on-line)

delegating this responsibility to rule-based systems that use expert-written rule sets for years. For the financial business, detecting credit card fraud is a critical issue. These industries have suffered a great deal as a result of fraudulent operations aimed at increasing income and losing customers' trust. As a result, these businesses must identify fraudulent transactions before they become a major issue. Unlike other machine learning issues, the target class distribution in credit card fraud detection is not evenly distributed. It's also known as the unbalanced data problem or the class imbalance problem. However, they are increasingly turning to a machine learning method, which can significantly improve the process [7].

The goal of this article is to develop a credit card fraud detection model that performs feature selection, data balancing and deep representation learning to create effective representations of transaction behaviours. At the same time, we hope that our model will be stable. There are numerous approaches to dealing with the issue of class disparity. This study focuses on a new learning representation that can improve fraud detection performance while also maintaining performance stability. As our representation learning model, we provide a deep neural network that maps the original attributes of transactions into deep representations for reliably recognising fraud transactions [8]. The acquired deep representations should, intuitively, optimise both intraclass compactness and interclass separability at the same time. As the loss function of our deep representation learning model, we create a novel function called Full centre loss (FCL).

Feature selection is performed to remove the unwanted features present in the dataset and to improve the classification accuracy. The proposed method uses a Bat optimization algorithm (BOA) for selecting the features. The echolocation behaviour of microbats with variable pulse rates of emission and loudness is the basis for this bat algorithm. Xin-She Yang[9] created the Bat-inspired algorithm, a metaheuristic optimization method.

SMOTE (Synthetic Minority Over-sampling Technique) [10] is an oversampling technique that produces synthetic minority class samples. It has the potential to outperform basic oversampling and is commonly utilised. When the balanced dataset is trained and tested, by comparing the under-sampling and oversampling methods, it was seen that oversampling techniques provided the best results [11]. SMOTE provided better results than other sampling methods where it gave a 50% of fraud transactions equal to legal transactions.

The structure of this paper is as follows. Related research is reviewed in Section II. Section III describes the existing credit card fraud detection model, in Section IV, we discuss the proposed approach for credit card fraud detection, in Section V, experiments are made and in Section VI, a comparison and analysis of experimental results are discussed. Finally, Section VII summarizes the conclusions of the study.

2 Related Works

Based on the divide-and-conquer concept, Zhenchuan Li et al., [12] presented a novel hybrid strategy to solve the problem of class imbalance with overlap. A unique assessment criterion, Dynamic Weighted Entropy (DWE), is proposed to evaluate the quality of the overlapping subset to attain good attributes. The Credit Card Fraud Detection dataset from Kaggle was used in this experiment. The goal of DWE is to increase model efficiency while minimising information loss and conserving time.

Javad Forough and Saeedeh Momtazi [13] proposed an ensemble model based on sequential modelling of data using deep recurrent neural networks, Long short term memory (LSTM), and Gated Recurrent Unit (GRU), as well as a novel voting mechanism based on artificial neural networks to detect fraudulent actions. Two datasets are used: data from European cardholders in September 2013 and data from a Brazilian bank. On the Brazilian dataset, LSTM beats GRU and ensemble model l as baseline models, with a recall of 71.44 percent.

In their study, Akila S and Srinivasulu Reddy [14] offers a cost-sensitive Risk Induced Bayesian Inference Bagging model (RIBIB) for credit card fraud detection. A constrained bag formation method, a Risk Induced Bayesian Inference method as a base learner, and a cost-sensitive weighted voting combiner are all part of RIBIB's bagging architecture. Experiments are carried out on data from Brazilian banks. The proposed approach has an accuracy of 86.9% and is less expensive.

A convolutional neural network (CNN) based method to identify fraudulent transactions was put out by Fu et al. [15]. The input characteristics were turned into matrices, which were then processed into pictures. It was also suggested to include a new function termed trading entropy to identify more intricate fraud patterns and boost classification precision. A real-time fraud detection technique based on recurrent neural networks was suggested by Wang et al. [16]. Many researchers use CNN in various applications [17-19].

Suresh Kumar et al., [20] in their study seek to forecast the occurrence of fraud using different machine learning algorithms such as support vector machine (SVM), k-nearest neighbour (KNN), and artificial neural network (ANN). The Credit Card Fraud Detection dataset from Kaggle is used by the system. The proposed technology has a 99.92 percent accuracy rate.

Honghao Zhu et al., [21] wanted to assess the effectiveness of several intelligent optimization approaches for optimising a WELM in an unbalanced classification. On 14 imbalanced datasets, experimental results reveal that the three optimised WELMs outperform the comparison algorithms in classification.

The fraud detection problem is phrased as a sequence classification task in Subudhiet al., [22], and LSTM networks are used to incorporate transaction sequences. The research is based on a credit-card transaction dataset collected between March and May 2015. The results of the experiments reveal that LSTM has higher prediction accuracy than RF. There is much research that uses LSTM network in various application [23].

To overcome the problem of imbalanced data, Huang Tingfei et al., [24] presented an oversampling strategy based on variational automatic coding (VAE) mixed with standard deep learning techniques. The VAE-based oversampling strategy appears to help deal with imbalanced classification problems, according to the findings of the experiments. There are additional drawbacks, such as the fact that this technology cannot be used in an unsupervised environment. To detect fraud, Sikdar et al., [25] developed the IFDT C4.5 decision tree, which uses intuitionistic fuzzy logic and the C4.5 decision tree. In this case, intuitionistic fuzzy logic aids in the consideration of cognitive qualities of attributes, ensuring that valid transactions are not misclassified as fraudulent and fraudulent transactions are not misclassified as legitimate. A Singaporean bank's data set was employed, as well as a synthetic data set for evaluation purposes.

Fabrizio Carcilloet al.[26] presented SCARFF, a Scalable Real-time Fraud Finder that combines Big Data tools (Kafka, Spark, and Cassandra) with a machine learning approach that deals with imbalance, non-stationarity, and feedback latency. The results suggest that the system can withstand an incoming rate of 200 transactions per second, which is a stunning outcome when compared to the present pace of 2.4 transactions per second.To detect credit card fraud, Sanjeev Jha et al., [27] used a transaction aggregation technique in their study.

Before each transaction, transactions are aggregated to capture consumer buying behaviour, and these aggregations are used for a model estimate to identify fraudulent transactions. Credit card fraud detection has become one of the most popular topics in the fraud detection field in recent years, thanks to the steady growth of e-commerce transactions. In general, fraud detection is difficult due to two major issues: Changes in the data dynamic and class imbalance.

Credit card fraud detection has long been researched as a problem of class inequality. Resampling is a well-known approach [28] [29]. A training data set can be balanced by eliminating some samples from the majority class (i.e. under-sampling) or by creating some examples for the minority class (i.e., over-sampling). Ensemble approaches, like bagging, boosting, and stacking, are also frequently employed to address the problem of class imbalance. Another approach is to use cost-sensitive learning, in which different misclassification error costs are applied to different classes, with a higher cost being assigned to a minority class in general. Aside from the quantity imbalance, the spatial distribution of cases from various classes has a significant impact on the classification results. Samples at the crossroads of majority and minority, for example, are more significant for a classifier because they are more difficult to identify effectively. As a result, the Gaussian mixture under-sampling approach is offered as a way to sample more informative cases and improve classifier performance.

The reason for the problem of transaction data dynamic change is that users' transaction habits are changing and evolving. The distortion of the original transaction data distribution is caused by variations in consumption seasonality and fraud trends. However, most fraud detection algorithms, such as Support vector machines (SVMs), Random forests (RFs), and CNNs, assume that classes are balanced and data distribution is constant. From 1990 to 2017, the literature gives a fairly thorough examination of credit card fraud detection technologies. The majority of these strategies focus on merging several class imbalance processing methods to improve a classifier's performance, but they ignore the problem of data dynamic change.

3 Methodology

This section describes the proposed system and its components in detail. Figure 1 depicts the proposed system's schematic diagram.

3.1 Feature engineering

Feature engineering [30-31] is the process by which knowledge of data is used to construct explanatory variables, features that can be used to train a predictive model. Engineering and selecting the correct features for a model will not only significantly improve its predictive power, but will also offer the flexibility to use fewer complex models that are faster to run and more easily understood.

Before the card is recognised and suspended, a fraudster will try to abuse it as much as possible in a short period. As a result, we should notice unusual transactions within a short amount of time. We will be able to detect abrupt changes if we aggregate transactions [32] [33] over time to achieve this goal. The amount of the transactions, whether it's the minimum, maximum, mean, or sum, can disclose a lot of information. Magnetic stripe transactions are more vulnerable to fraud than chip or pin transactions. As a result, we may calculate an aggregated sum per card based on the transaction type.

3.2 Feature selection-BAT

Engineers lead machine learning systems toward an objective through feature selection, which is a discriminating process.

Algorithm: Bat optimization

Create the a_i and b_i bat populations (i = 1, 2,..., n).
Set the frequencies fi, pulse rates p_i and loudness Ai to their default values.
While (n< Maximum number of iterations)
    By adjusting the frequency, individuals may come up with different solutions.
    [(10.1) to (10.3)] Update velocities and locations/solutions
    If (rand >p_i)
        Choose a solution from the list of the best options.
        Create a local solution based on the best option that has been chosen.
    end if
    Create a new solution by flying around at random.
    if (rand <A_i& $f(a_i) < f(a*)$)
        Accept the new approaches.
        Increase p_i and reduce A_i
    end if
    Determine the current best a* by ranking the bats.
end while

Feature selection can be important in optimising portions of what experts call the "bias-variance trade-off" in machine learning, in addition to the idea of eliminating complexity from systems at scale. In our methodology, mutual information feature selection method. The application of information gain to feature selection is mutual information from the discipline of information theory. Mutual information is measured between two variables and quantifies the reduction in uncertainty for one variable when the other variable's value is known.

The following is a summary of the idealisation of microbat echolocation [34]: Each virtual bat flies at a random velocity v_i at point x_i, with a variable frequency or wavelength and loudness A_i. It alters frequency, loudness, and pulse emission rate r as it seeks and discovers its victim. A local random stroll amplifies the search. The best candidates are chosen till specific requirements are reached. Use the following approximations in addition to these reduced assumptions to keep things simple. In general, a range of wavelengths [min, max] corresponds to a frequency f in a range [f_{min}, f_{max}].

A. Virtual Bats' Movement

In the simulation, virtual bats [35] must be utilised. In a d-dimensional search space, there must be specified rules for updating their positions x_i and velocities v_i. At time step t, the new solutions $\boldsymbol{x_i^{t+1}}$ and velocities $\boldsymbol{v_i^{t+1}}$ are provided in equ (1-3) as shown below:

$$f_i = f_{min} + (f_{max} - f_{min})\beta \quad (1)$$

$$v_i^{t+1} = v_i^t + (x_i^t - x_*)f_i \quad (2)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3)$$

where $\beta \in [0, 1]$ is a randomly generated vector from a uniform distribution. Here, x_* represents the current global best location (solution), as determined by comparing all of the solutions among all of the n bats. Because the product $\lambda_i f_i$ is constant, it can adjust the velocity change while correcting the other factor λ_i (or f_i), depending on the type of problem. Depending on the domain size of the problem of interest, we utilise $f_{min} = 0$ and $f_{max} = O(1)$ in implementation. Each bat is given a frequency that is drawn equally from [f_{min}, f_{max}] at first.

Once a solution is chosen from among the existing best solutions for the local search, a new solution for each bat is generated locally using a random walk.

$$x_{new} = x_{old} + \sigma \in_t A^{(t)} \quad (4)$$

Where t currently comes from a Gaussian normal distribution N(0, 1), and σ is a scaling factor. The scalings of the design variables of an optimization problem under σ consideration should be linked. Because fi essentially regulates the pace and range of movement of the swarming particles, the technique for updating bat velocities and positions may resemble that of typical particle swarm optimization. BA, on the other hand, can be more effective because it influences exploration and exploitation through frequency tuning and parameter control.

B. Loudness and Pulse Emission

Furthermore, as the iterations go, the loudness A_i and the rate r_i of pulse emission must be changed proportionately. Because the loudness of a bat's call drops as it approaches its prey, and the rate of pulse emission rises, the loudness can be set to any convenient value. We can also use $A_0 = 1$ and $A_{min} = 0$ for simplicity, assuming that $A_{min} = 0$ indicates that a bat has just discovered the prey and has temporarily stopped generating any sound. Now we have

$$A_i^{t+1} = \alpha A_i^t, \qquad r_i^{t+1} = r_i^0[1 - exp(-\gamma t)] \quad (5)$$

where α and γ are constants.

For any 0 <α< 1 and γ > 0, we have

$$A_i^t \to 0, r_i^t \to r_i^0 \text{, as } t \to \infty. \quad (6)$$

We can use α = γ in the simplest scenario, and we used α= γ=0.9 in our simulations. The sample code, however, does not include the modifications of A and r, which is primarily to demonstrate the importance of frequency tuning in the bat method. The selection of settings necessitates some trial and error. Randomization can be used to achieve varied values of loudness and pulse emission rate for each bat at first.

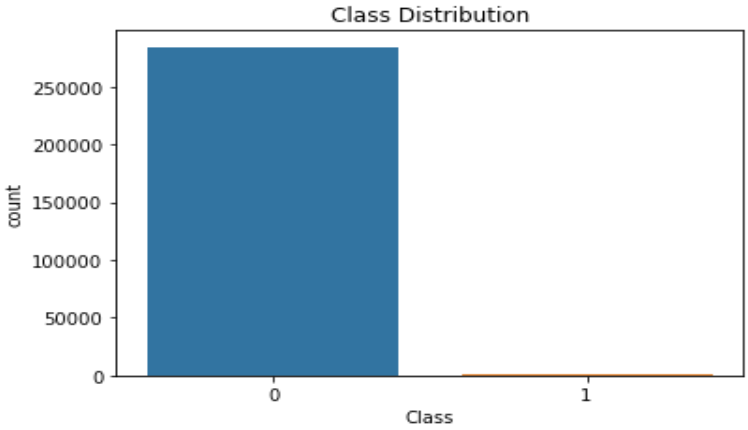


Figure 2: Distribution of transactions per target variable

Algorithm: SMOTE algorithm

Input: Raw training dataset **D**
Output: Oversampled training dataset D^+

1. Calculate the oversampling data size $S^+$and initialize $D^+ = D$
2. Identification of noise, borderline and edge samples
3. Sample importance calculation
Calculate borderline sample importance
Calculate noisy sample importance
Calculate edge sample importance
4. Synthetic minority samples generation
for j from 1 to S^+
Select one sample from the borderline, edge and noise minority samples as a^+ with respect to their samples importance:
Select a nearest neighbor of a^+in majority and minority classes as a^+_{knn} with respect to theirsample balance;
Calculate α value with respect to the sample importance of a^+ and a^+_{knn};
Generate the synthetic minority samples and add it to D^+;
5. Export the oversampled training dataset D^+ to the classification model

For example, the initial loudness A_i^0can be set to 1, whereas the initial emission rate r_i^0 can be set to zero or any value between r_i^0 (0, 1]. Only if the new solutions improve will their loudness and emission rates be changed,

indicating that these bats are on their way to finding the best answer. We can observe that BA can capture many characteristics of other algorithms by closely examining it. BA essentially becomes the standard PSO when the changes in the frequency fi are replaced by a random parameter and $A_i = 0$ and $r_i = 1$. If the velocities are not used,

we utilise fixed loudness and rate: A_i and r_i. For example, when $A_i = r_i = 0.7$, this algorithm can be reduced to a simple harmony search (HS), because the frequency/wavelength change is essentially pitch adjustment, and the rate of pulse emission is similar to the harmonic acceptance rate in the HS algorithm. HS and PSO, in other words, can be thought of as specific examples of BA. As a result, it's no surprise that BA is effective.

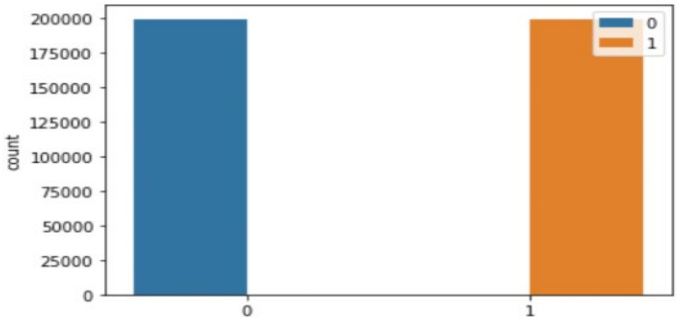


Figure 3: Balancing data after using SMOTE

3.3 SMOTE

The credit card dataset is quite unbalanced, which has a negative impact on how well ML models perform. The unbalanced class problem is frequently solved using the synthetic minority oversampling method (SMOTE) [36]-[38]. By adding synthetic samples to the minority class, this oversampling strategy balances the distribution of classes in the dataset. By eliminating certain samples from the majority class, undersampling techniques provide a balanced dataset.

Figure 2 depicts the dataset graph before applying the SMOTE. Here legal data is more than the fraud data.

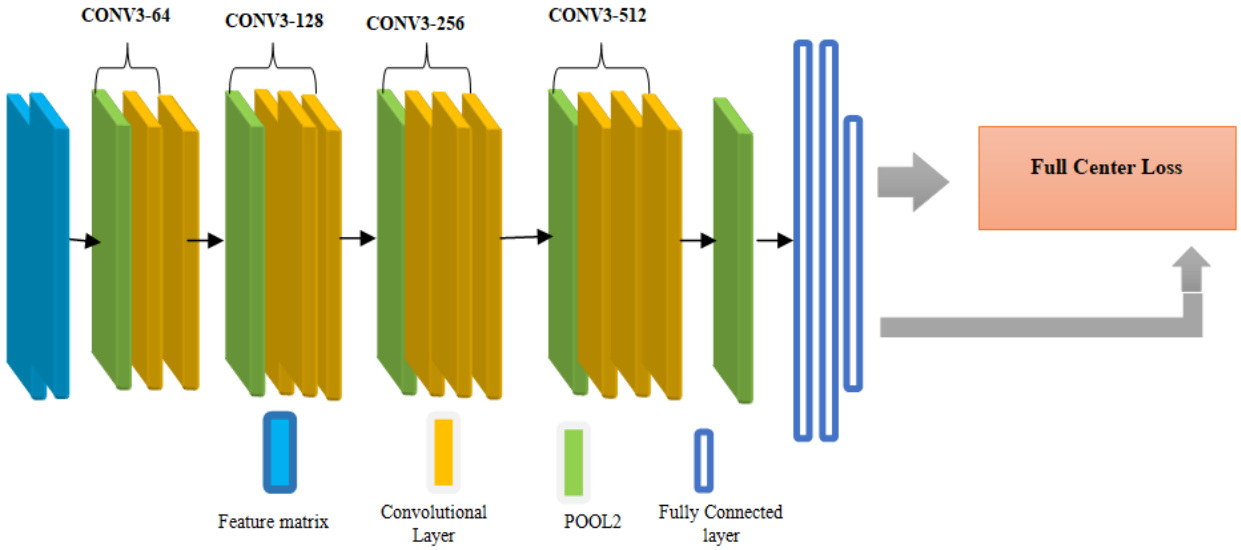


Figure 4: Proposed architecture of Deep representation learning with FCL

So we want to balance the data from the minority class to the majority class by applying SMOTE technique. Unbalanced data categorization remains a major challenge in the fraud detection process. Indeed, we find that one class variable has much fewer training occurrences than the other in an unbalanced sample. The first is known as the minority set, and the second is known as the majority set. When executing an imbalanced dataset to detect fraudulent transactions, most classification algorithms rank the majority class with a very high accuracy rate and the minority class with a much lower one. This means that the proposed classification model has a hard time detecting minority class samples. Samples from the majority classes (i.e., legal transaction class) exceed those from the minority classes (i.e., fraud transaction class) in the credit card fraud dataset, skewing the data in favour of the majority class. The gap between the two groups of cases in the data set must be narrowed to improve the effectiveness of categorization results.

To address the problem of class imbalance and produce a more accurate result, we use SMOTE [39-41] as a data balancing strategy. SMOTE is an oversampling technique that produces synthetic minority class samples. It has the potential to outperform basic oversampling and is commonly utilised. Figure 3 depicts the balanced data graph using SMOTE which balances the fraud and legal data.

4. CONVOLUTIONAL NEURAL NETWORK (CNN)

We employ a CNN [42] to detect credit card thefts in this paper because the CNN model is good for training huge amounts of data and includes a method to prevent over-fitting. Some domains, such as classification and audio signal processing, have shown success with convolutional neural networks. However, the CNN model is not appropriate for all types of data. To adjust the CNN [43] model, the feature transformations approach is provided. Credit card transaction characteristics can be divided into various categories. And, depending on the time window, each group has various characteristics. The deep neural network layers (e.g., CNN) for creating

separable and discriminative representations, and the fully centre loss layer for overseeing the model training, make up the fraud transaction detection model used in this paper.

The goal of this article is to improve the quality of those learned deep features and the performance of fraud transaction detection by optimising the loss function [44]. The training of deep convolution neural network layers that project the original feature space of transactions into a deep feature space is supervised by our loss function. The goal is to condense transactions from the same class as much as possible while separating transactions from other classes as much as possible. For this reason, we created an Full Center Loss (FCL) that combines two types of losses: Angel Center Loss (ACL) for dealing with the separability of transactions from different classes and DCL for dealing with the compactness of transactions from the same class.

Convolutional layers, pooling layers, and fully-connected (FC) layers are the three types of layers [45-46] that make up CNN. CNN architecture will be constructed when these layers are layered. There are two additional significant factors, the dropout layer and the activation function, which are explained below, in addition to these three layers.

Figure 4 depicts the architecture of Deep representation learning and the architecture is explained below.

4.1 Convolution Layer

This is the initial layer that extracts the different characteristics from the input data. The convolution mathematical operation is done between the input data and a filter of a certain size M x M in this layer. The dot product between the filter and the sections of the input data concerning to the size of the filter is taken by sliding the filter across the input dataset (M x M).

4.2 Pooling Layer

A Pooling Layer is usually applied after a Convolutional Layer. This layer's major goal is to lower the size of the convolved feature map to reduce computational expenses. This is accomplished by reducing the connections between layers and operating independently on each feature map. There are numerous sorts of pooling procedures, depending on the mechanism utilised.

4.3 Fully Connected Layer (FC)

The weights and biases, as well as the neurons, make up the FC layer, which is used to link the neurons between two layers. The last several layers of a CNN Architecture are generally positioned before the output layer.

4.4 Dropout

When all of the characteristics are connected to the FC layer, the training dataset is prone to overfitting. Overfitting happens when a model performs so well on training data that it has a negative impact on its performance when applied to new data.

4.5 Activation Function

Finally, the activation function is one of the most crucial elements in the CNN model. They're utilised to learn and approximate any form of network variable-to-variable association that's both continuous and complex. In simple terms, it determines which model information should fire in the forward direction and which should not at the network's end.

5 LOSS FUNCTION - Full Center Loss (FCL)

Angle center loss (ACL) is an upgraded Softmax Loss that was created specifically for our binary classification problem of credit card fraud detection. When compared to the softmax function, it can improve classification ability. Meanwhile, the advantage of the Softmax Loss simplicity is preserved in ACL [47]. DCL was first presented to quantify the aggregate of each class's deep characteristics.

$$L_{Full} = \sum_{i=1}^{k} (L_{ACL\_i} + \alpha L_{DCL\_i}) \quad (7)$$

where k is the number of the mini batch data used to train our deep representation learning model and is α hyper parameter to trade off these two losses.

5.1 Angle Center Loss (ACL)

Deep features should generally aim for the highest levels of intraclass compactness and interclass separability. The SL of a typical CNN model is straightforward and useful in many classification tasks, but it is useless for producing discriminative features. When utilising the original SL to resolve a binary class issue, the posterior probability of the learned deep representation di of an input sample x_i with label 0 or 1 may be expressed as this.

$$p0 = \frac{e^{W_0^T d_i + b_0}}{e^{W_0^T d_i + b_0} + e^{W_1^T d_i + b_1}} \quad (8)$$

$$p1 = \frac{e^{W_1^T d_i + b_1}}{e^{W_0^T d_i + b_0} + e^{W_1^T d_i + b_1}} \quad (9)$$

The weights and biases of the softmax layer in CNN corresponding to class 0 and 1 are (W0, b0) and (W1, b1), respectively. The last completely linked layer's output is d_i. The posterior probability of d_i belonging to classes 0 and 1 are p0 and p1, respectively.
The inherent angular distribution of features learnt by the original SL has been proven. If the label y_i is applied to an input deep representation di the original SL of deep representation di may be recast as follows:

$$L_{softmax\_i} = -\log\left(\frac{1}{1+e^{\left(W_{\tilde{y}_i}^T - W_{yi}^T\right)d_i + \left(b_{\tilde{y}_i} - b_{yi}\right)}}\right) \quad (10)$$

where $\tilde{y}_i$ denotes another class different from y_i in the binary classification.

The learnt deep representations of cases from various classes should retain separability as thoroughly as feasible to provide stable classification performance. The modified SL, on the other hand, can only directly minimise the angles between di and the related $W_{\tilde{y}}$i.
As a result, we create a new stronger constraint: W_{yi} and $W_{\tilde{y}_i}$ are in opposing directions: W= W_{yi} = $W_{\tilde{y}_i}$. Finally, ACL may be reformed as ACL with deep representation d_i.

$$L_{ACL\_i} = log\left(1 + e^{-2W^T d_i}\right) \quad (11)$$

5.2 Distance center loss (DCL)

The separability of learned deep representations from different classes is primarily determined by DCL. We use the centre loss in Euclidean space to quantify the compactness of intraclass deep representations, which may be quantified by the distance between an instance and its associated centre. The deep representation fi centre loss may be expressed as follows:

$$L_{D_{CL\_i}} = \frac{1}{2}\left\|d_i - e_{yi}\right\|\frac{2}{2} \quad (12)$$

where e_{y_i} denotes the corresponding class center of d_i with the label y_i.

We call it DCL since the original centre loss employs the Euclidean distance directly. CNNs supervised by DCL are trainable and can be optimised using the conventional stochastic gradient descent (SGD) approach. When training a model, the distance centre of every class is obtained by averaging the learnt deep representations in the class in Euclidean space. To minimise massive perturbations produced by a few mislabelled samples, a scalar is employed to restrict the update pace of the distance centres.

$$\frac{\partial L_{DCL\_i}}{\partial d_i} = d_i - e_{yi} \quad (13)$$

6 Experiment

6.1 Data Sets

The first credit card fraud detection data set is available on Kaggle [48] and has been utilised in numerous studies. The other is a set of confidential transaction data from a Chinese financial firm. Credit card transactions of European cardholders in September 2013 are included in this data collection. These transactions are generated in two days, and out of the total of 284 807 transactions, there are 492 fraudulent transactions. This data set is skewed: the positive class (fraud incidents) accounts for only 0.172 percent of total transactions. This data collection has 30 features for each transaction.

6.2 Performance Evaluation Metrics

A 70:30 train-test split ratio is utilised to evaluate the performance of the suggested technique for credit card fraud detection. Several measures, including the Confusion Matrix, Precision, Recall, Accuracy (ACC), AUC, and F1-score, are used to evaluate the performance [49] of the suggested technique.

The measures that were used are defined as follows.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{14}$$

$$Recall = \frac{TP}{TP+FN} \tag{15}$$

$$Precision = \frac{TP}{TP+FP} \tag{16}$$

$$F1\ score = \frac{2*Precision*Recall}{Precision+Recall} \tag{17}$$

When combined, precision and recall are critical measures to utilise with unbalanced data (i.e., F1-score). Precision is a measure of the number of relevant outcomes, whereas Recall is a measure of the applicability of the resulting scale and proximity to the desired solution. A high Recall score indicates a low rate of false negatives (FN), whereas a high Precision score indicates a low rate of false positives (FP). High Precision and Recall ratings suggest that the classifier restores results accurately and recovers the majority of positive outcomes. As a result, the Precision-Recall curve provides a complete picture of the classifier's accuracy and is robust even when the data set is unbalanced. In addition to the above metrics, we use the AUC value as a general performance indicator. The AUC is a graphic representation of the false positive rate (FPR) and true positive rate (TPR) at various levels. AUC is seen to be a superior overall performance metric to accuracy because it is not affected by a cut-off number. An AUC value close to one indicates a model that performs well overall. The confusion matrix is shown in table 1.

Table 1: Confusion Matrix

	True Fraud	True Legitimate
Predicted Fraud	TP	FP
Predicted Legitimate	FN	TN

6.3 Experimental Results and Discussions

On the Kaggle data set, experiments were run. To select the best features and to reduce the training time, a transaction aggregation strategy is used for feature extraction and the best features are selected using the bat algorithm.

We utilise the upsampling method SMOTE to produce more fraud transactions based on the real transactions to balance the data set with a ratio of 1:1, because there are only 492 fraud transactions in the data set, which can lead to a large variance in the test results for different models. The training set is made up of 70% of transactions, while the test set is made up of the remaining 30%. Because of a neural network's nonlinear characteristic combining ability, any neural network model outperforms RF. Table 2 shows the comparison of normal and ensemble classifiers with neural networks. Figure 5 shows the comparison of various classifiers with the proposed work

Second, our ACL beats other loss functions, implying that our ACL has a better maximum angle separation than others. Furthermore, when ACL and DCL are combined, the performance of our models (i.e., FCL) improves noticeably, indicating the relevance of intraclass compactness of acquired representations. This also demonstrates that FCL may increase fraud detection performance by strengthening deep representation learning models and obtaining better representations.Transactions from the first month are utilised as the training set (designated as T) and the remaining samples are used as the testing set in the experiment. Instead of testing every sample in the testing set at once, we split it into six groups (T1 through T6), each of which included samples from 10 consecutive days. We evaluate each of these sets of data separately in order to track performance changes.

Table 2: Comparison of normal and ensemble classifiers with neural networks

Technique	Precision (%)	Recall (%)	F1score (%)	Accuracy (%)
Decision tree	85	88	86	95
RFC	97	88	92	95
Logistic regression	90	81	85	87
CNN	98	97	97	99

Table 3: Performance of different models on Kaggle’s credit card fraud dataset

	T1		T2		T3		T4		T5		T6	
Method	F1	Auc_Pr	F1	Auc_Pr	F1	Auc_Pr	F1	Auc_Pr	F1	Auc_Pr	F1	Auc_Pr
SL	0.79	0.78	0.74	0.79	0.65	0.58	0.67	0.58	0.74	0.71	0.70	0.65
LMSL	0.79	0.80	0.76	0.83	0.67	0.53	0.62	0.61	0.78	0.78	0.71	0.62
ASL	0.80	0.78	0.66	0.84	0.67	0.57	0.66	0.64	0.78	0.72	0.73	0.70
ACL	0.80	0.81	0.82	0.83	0.66	0.67	0.69	0.77	0.81	0.79	0.74	0.67
FCL	0.81	0.82	0.86	0.88	0.74	0.72	0.82	0.82	0.85	0.82	0.79	0.82

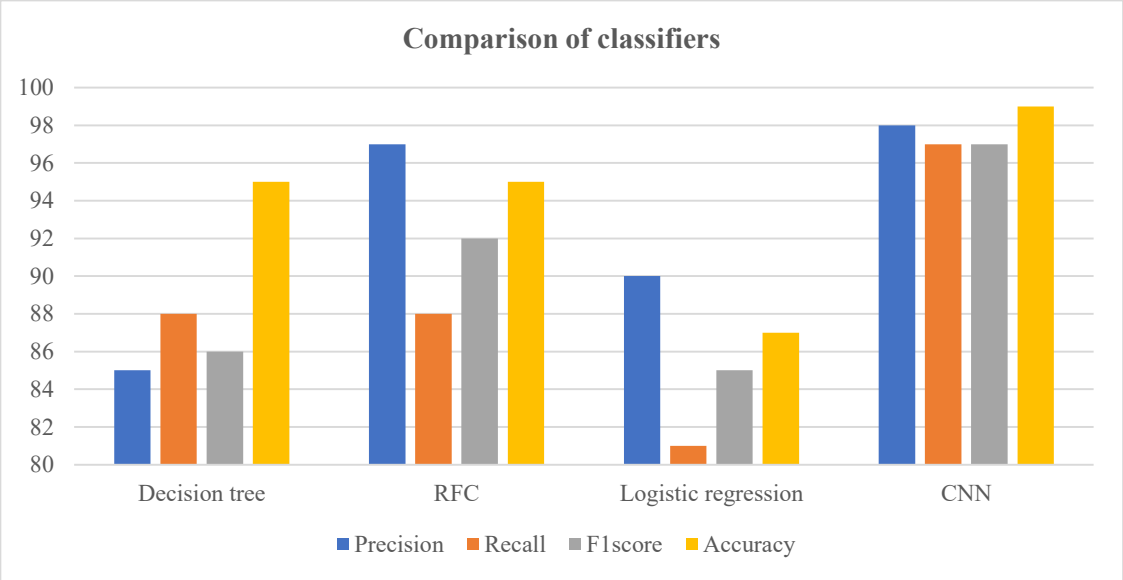


Figure 5: Comparison of various classifiers with proposed work

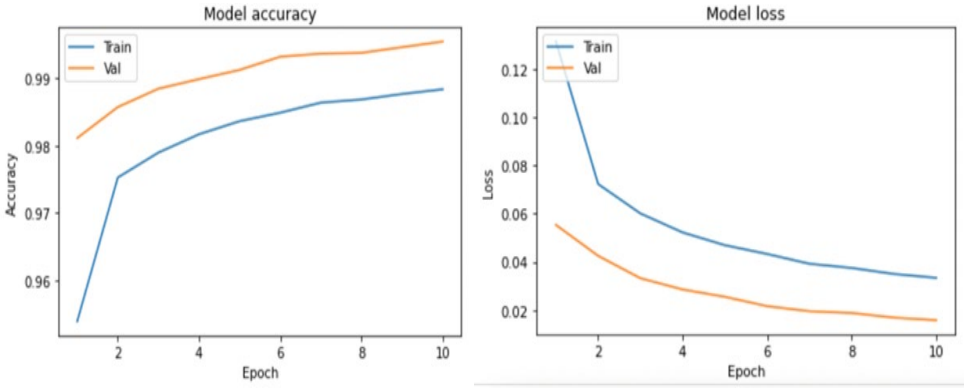


Figure 6: Graph for Model loss and Model Accuracy

Figure 6 depicts the graph for model loss and accuracy. The loss value indicates how well or poorly a model performs after each optimization iteration. The accuracy metric is used to interpretably quantify the algorithm's performance. The accuracy of a model is generally calculated as a percentage once the model parameters have been defined. Table 3 shows the performance of different models on Kaggle's credit card fraud dataset.

7 Conclusion and Prospectives

For credit card fraud detection, a deep representation learning model is developed, which has the advantage of achieving good and consistent results. The proposed technique will aid in automatically detecting fraud, uncovering hidden correlations in data, and requiring less time for the verification process than current fraud detection methods. This is accomplished by employing the Bat Optimization Algorithm to choose pertinent and

distinctive traits. The severely unbalanced dataset for credit card fraud is then balanced using the Synthetic Minority over-sampling approach (SMOTE). Finally, utilising the complete centre loss function to build the CNN model for anomaly detection in credit card data, fraud detection performance and stability are improved. The Kaggle dataset is used to evaluate the proposed model, which produces accurate results of about 99%. The advantages of our strategy are demonstrated by the experimental findings. There is still room for improvement, although our loss functions can provide more consistent performance for fraud detection. For instance, concept drift should be taken into account while evaluating the fraud detection model's performance stability. We intend to examine the idea of a drift problem from the perspective of a loss function in the future.

8 References

[1]. B. Baesens, V. Van Vlasselaer and W. Verbeke, Fraud Analytics Using Descriptive Predictive and Social Network Techniques: A Guide to Data Science for Fraud Detection, Hoboken, NJ, USA:Wiley, 2015.

[2]. Y. Singh, K. Singh and V. Singh Chauhan, "Fraud Detection Techniques for Credit Card Transactions," 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 821-824, doi: 10.1109/ICIEM54221.2022.9853183.

[3]. Y. Li et al., "Automated Anomaly Detection via Curiosity-Guided Search and Self-Imitation Learning," in IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 6, pp. 2365-2377, June 2022, doi: 10.1109/TNNLS.2021.3105636.

[4]. L. Zheng, G. Liu, C. Yan, and C. Jiang, “Transaction fraud detection based on total order relation and behavior diversity,” IEEE Trans. Comput. Soc. Syst., vol. 5, no. 3, pp. 796–806, Sep. 2018

[5]. Y. -Y. Hsin, T. -S. Dai, Y. -W. Ti, M. -C. Huang, T. -H. Chiang and L. -C. Liu, "Feature Engineering and Resampling Strategies for Fund Transfer Fraud With Limited Transaction Data and a Time-Inhomogeneous Modi Operandi," in IEEE Access, vol. 10, pp. 86101-86116, 2022, doi: 10.1109/ACCESS.2022.3199425.

[6]. V. Van Vlasselaer et al., “APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions,” Decision Support Syst., vol. 75, pp. 38–48, Jul. 2015.

[7]. Z. Yuan, H. Chen, T. Li, X. Zhang and B. Sang, "Multigranulation Relative Entropy-Based Mixed Attribute Outlier Detection in Neighborhood Systems," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 52, no. 8, pp. 5175-5187, Aug. 2022, doi: 10.1109/TSMC.2021.3119119.

[8]. Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in Proc. Eur. Conf. Computer Vis. (ECCV). Cham, Switzerland: Springer, 2016, pp. 499–515.

[9]. Yang, XS, “A New Metaheuristic Bat-Inspired Algorithm”, In: González, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N. (eds) Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). Studies in Computational Intelligence, vol 284, 2010 Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-12538-6_6

[10]. S. F. Abdoh, M. A. Rizka, and F. A. Maghraby, “Cervical cancer diagnosis using random forest classifier with SMOTE and feature reduction techniques,” IEEE Access, vol. 6, pp. 59475–59485, 2018.

[11]. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: Synthetic minority over-sampling technique,” J. Artif. Intell. Res., vol. 16, pp. 321–357, Jul. 2018.

[12]. Zhenchuan Li, Mian Huang, Guanjun Liu, Changjun Jiang, “A hybrid method with dynamic weighted entropy for handling the problem of class imbalance with overlap in credit card fraud detection,” Expert Systems with Applications, Volume 175, 2021, 114750, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2021.114750

[13]. Javad Forough, SaeedehMomtazi, “Ensemble of deep sequential models for credit card fraud detection,” Applied Soft Computing, Volume 99, 2021, 106883, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2020.106883.

[14]. S Akila, U Srinivasulu Reddy, “Cost-sensitive Risk Induced Bayesian Inference Bagging (RIBIB) for credit card fraud detection”, Journal of Computational Science,Vol 27,Pages 247-254,2018. https://doi.org/10.1016/j.jocs.2018.06.009.

[15]. K. Fu, D. Cheng, Y. Tu and L. Zhang, "Credit card fraud detection using convolutional neural networks", Proc. Int. Conf. Neural Inf. Process, pp. 483-490, 2016.

[16]. S. Wang, C. Liu, X. Gao, H. Qu and W. Xu, "Session-based fraud detection in online e-commerce transactions using recurrent neural networks", Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases, pp. 241-252, 2017.
[17]. S. Haseena, T. Revathi, Soft biometrics based face image retrieval using improved grey wolf optimization. IET Image Process 14, 451–461 (2020)
[18]. S. Haseena, T. Revathi, Deep learning-based facial expression recognition using improved Cat swarm optimization. J. Ambient Intell. Hum. Comput. 12, 3037–3053 (2020)
[19]. S. Haseena et al., Prediction of the age and gender based on human face images based on deep learning algorithm. Comp. Math. Methods Med. 16, 1413597 (2022). https://doi.org/10.1155/2022/1413597
[20]. Asha RB, Suresh Kumar KR, "Credit card fraud detection using artificial neural network", Global Transitions Proceedings, Vol 2,2021, Pages 35-41
[21]. Honghao Zhu, Guanjun Liu, Mengchu Zhou, Yu Xie, Abdullah Abusorrah, Qi Kang, "Optimizing Weighted Extreme Learning Machines for imbalanced classification and application to credit card fraud detection",Neurocomputing, Vol 407, 2020,Pages 50-62, https://doi.org/10.1016/j.neucom.2020.04.078
[22]. S. Subudhi and S. Panigrahi, "Application of OPTICS and ensemble learning for database intrusion detection", J. King Saud Univ. Comput. Inf. Sci., May 2019.
[23]. Saroja, Haseena & Dharshini, S. Deep learning approach for prediction and classification of potable water. ANAL. SCI. (2023). https://doi.org/10.1007/s44211-023-00328-2
[24]. H. Tingfei, C. Guangquan and H. Kuihua, "Using Variational Auto Encoding in Credit Card Fraud Detection", in IEEE Access, vol. 8, pp. 149841-149853, 2020, doi: 10.1109/ACCESS.2020.3015600.
[25]. Sikdar Md. S Askari, Md. Anwar Hussain, "IFDTC4.5: Intuitionistic fuzzy logic based decision tree for E-transactional fraud detection",Journal of Information Security and Applications, Vol52, 2020, 102469, ISSN 2214-2126
[26]. Carcillo, Fabrizio & Dal Pozzolo, Andrea & Le Borgne, Yann-Aël&Caelen, Olivier &Mazzer, Yannis &Bontempi, Gianluca, "SCARFF : a Scalable Framework for Streaming Credit Card Fraud Detection with Spark", Information Fusion. 41, 2017 10.1016/j.inffus.2017.09.005.
[27]. Jha, Sanjeev & Guillen, Montserrat & Westland, J, "Employing transaction aggregation strategy to detect credit card fraud", Expert Systems with Applications. 39. 12650–12657.2012, 10.1016/j.eswa.2012.05.018.
[28]. L. Zheng, G. Liu, C. Yan and C. Jiang, "Transaction Fraud Detection Based on Total Order Relation and Behavior Diversity", in IEEE Transactions on Computational Social Systems, vol. 5, no. 3, pp. 796-806, Sept. 2018, doi: 10.1109/TCSS.2018.2856910.
[29]. V. Van Vlasselaer et al., "APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions", Decision Support Syst., vol. 75, pp. 38–48, Jul. 2015.
[30]. V. Dheepa and R. Dhanapal, "Behavior based credit card fraud detection using support vector machines", ICTACT J. Soft Comput., vol. 4, no. 4, pp. 391-397, 2012.
[31]. A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi and G. Bontempi, "Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy", in IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 8, pp. 3784-3797, Aug. 2018, doi: 10.1109/TNNLS.2017.2736643.
[32]. S. Carta, G. Fenu, D. Reforgiato Recupero and R. Saia, "Fraud detection for E-commerce transactions by employing a prudential multiple consensus model", J. Inf. Secur. Appl., vol. 46, pp. 13-22, Jun. 2019.
[33]. T. Jiahong, W. Zhigang and L. Xiang, "Load Curve Clustering Based on Feature Engineering and Uniform Manifold Approximation," 2021 6th Asia Conference on Power and Electrical Engineering (ACPEE), Chongqing, China, 2021, pp. 883-887, doi: 10.1109/ACPEE51499.2021.9437006.
[34]. P. W. Tsai, J. S. Pan, B. Y. Liao, M. J. Tsai, V. Istanda, "Bat algorithm inspired algorithm for solving numerical optimization problems", Applied Mechanics and Materials, Vo. 148-149, pp.134-137 (2012)
[35]. X.-S. Yang, "A new metaheuristic bat-inspired algorithm", in Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). Berlin, Germany: Springer, 2010, pp. 65–74.
[36]. H. He and E. A. Garcia, "Learning from imbalanced data", IEEE Trans. Knowl. Data Eng., no. 9, pp. 1263–1284, Jun. 2008.
[37]. S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data", IEEE Trans. Neural Netw. Learn. Syst.,

[38]. Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 8, pp. 1798–1828, August 2013.
[39]. A. Ishaq, S. Sadiq, M. Umer, S. Ullah, S. Mirjalili, V. Rupapara, and M. Nappi, “Improving the prediction of heart failure Patients' survival using SMOTE and effective data mining techniques”,IEEE Access, vol. 9, pp. 39707_39716, 2021.
[40]. Asniar, N. U. Maulidevi, and K. Surendro, “SMOTE-LOF for noise identification in imbalanced data classification”, J. King Saud Univ. Comput. Inf. Sci., Feb. 2021.
[41]. Chawla, Nitesh & Bowyer, Kevin & Hall, Lawrence &Kegelmeyer, W, “SMOTE: Synthetic Minority Over-sampling Technique”, J. Artif. Intell. Res. (JAIR). 16. 321-357. 200210.1613/jair.953
[42]. A. M. Babu and A. Pratap, “Credit Card Fraud Detection Using Deep Learning”, 2020 IEEE Recent Advances in Intelligent Computational Systems (RAICS), 2020, pp. 32-36, doi: 10.1109/RAICS51191.2020.9332497.
[43]. Z. Zhang and S. Huang, “Credit Card Fraud Detection via Deep Learning Method Using Data Balance Tools”, International Conference on Computer Science and Management Technology (ICCSMT), 2020, pp. 133-137, doi: 10.1109/ICCSMT51754.2020.00033.
[44]. W. Zhang and Q. Liu, “Using the center loss function to improve deep learning performance for EEG signal classification”, Tenth International Conference on Advanced Computational Intelligence (ICACI), 2018, pp. 578-582, doi: 10.1109/ICACI.2018.8377524.
[45]. S. Albawi, T. A. Mohammed and S. Al-Zawi, “Understanding of a convolutional neural network”, International Conference on Engineering and Technology (ICET), 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.
[46]. S. Haseena, S. Bharathi, I. Padmapriya and R. Lekhaa, "Deep Learning Based Approach for Gender Classification," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2018, pp. 1396-1399, doi: 10.1109/ICECA.2018.8474919.
[47]. Z. Li, G. Liu and C. Jiang, “Deep Representation Learning With Full Center Loss for Credit Card Fraud Detection”,in IEEE Transactions on Computational Social Systems, vol. 7, no. 2, pp. 569-579, April 2020, doi: 10.1109/TCSS.2020.2970805.
[48]. Credit card fraud dataset from Kaggle https://www.kaggle.com/mlg-ulb/creditcardfraud
[49]. E. Ileberi, Y. Sun and Z. Wang, “Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost”, in IEEE Access, vol. 9, pp. 165286-165294, 2021, doi: 10.1109/ACCESS.2021.3134330.