Naif Arab University for Security Sciences
Journal of Information Security and Cybercrimes Research
مجلة بحوث أمن المعلومات والجرائم السيبرانية
https://journals.nauss.edu.sa/index.php/JISCR

JISCR

An Efficient Deep Learning Classification Model for Predicting Credit Card Fraud on Skewed Data

Naoufal Rtayli*
Faculty of Sciences, Abdelmalek Essaadi University, Tetouan, Morocco.

Received 30 Oct. 2021; Accepted 09 June. 2022; Available Online 22 June. 2022

Abstract

Due to fast-evolving technology, the world is moving to the use of credit cards rather than money in their daily lives, giving rise to many new opportunities for fraudsters to use credit cards maliciously. Based on the Nilson report, losses related to global cards were estimated to be over $35 billion by 2020. In order to maintain the security of users of these cards, the credit card company must develop a service to ensure that users are protected from any risks they may be exposed to. For this reason, we introduce a fraud detection model, denoted ST-BPNN, which is based on machine and deep learning approaches to identify fraudulent transactions. ST-BPNN was applied on real fraud detection data provided by the European bank. Comparing the obtained results from ST-BPNN with a recent state-of-the-art approach shows that our proposed model demonstrates high predictive performance for detecting fraudulent transactions.

I. Introduction

Since the introduction of credit cards and online payments, many scammers have found ways to exploit people and steal their credit card information for use in unauthorized purchases [1]. The result is a huge amount of fraudulent purchases every day [2]. Banks and e-commerce sites try to identify these fraudulent transactions and prevent them from happening again. Fig. 1 presents an example of a Credit Card Fraud Detection (CCFD) case.

Credit card fraud (CCF) involves the use of falsified or stolen credit card information and causes financial harm to the account holders or merchants involved [3]. Fraud is known to be dynamic and has no pattern, so it is not easily identified. Fraudsters use recent technological advances to their advantage. They somehow bypass security controls, resulting in the loss of billions of dollars. The total number of CCFs in the Single Euro Payments Area (SEPA) in 2016 was 1.8 billion euros out of a total of 4.38 trillion euros in transactions, which is 0.4% less than the previous year [4]. In 2016, based on Nelson's report, global credit card losses amounted to $21.84 billion and were estimated to reach $32 billion in 2020 [2]. Using machine learning and deep learning techniques to analyze and detect suspicious activity is one way to track fraudulent transactions to stop fraudsters before the transaction is processed and validated [5].

Keywords: Cybersecurity, Credit Card Fraud Detection, Imbalanced Data Problem, Artificial Deep Neural Networks, Machine learning Techniques, Classification Accuracy.

Production and hosting by NAUSS

* Corresponding Author: Naoufal Rtayli
Email: rtayli.naoufal@gmail.com
doi: 10.26735/TLYG7256

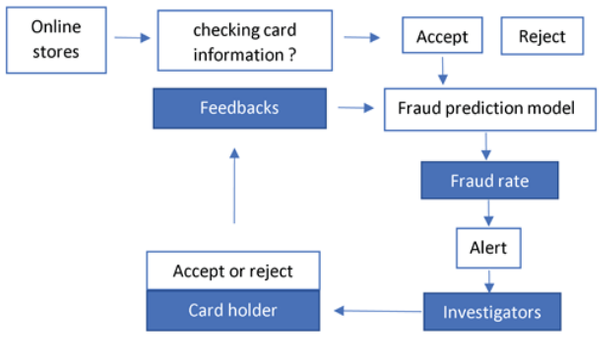

Fig. 1. Credit Card Fraud Detection (CCFD) case.

Deep Neural Networks (DNNs) are rapidly becoming one of the most popular machine learning (ML) tools, due to their ability to solve a wide range of problems, from language translation [6] [7], image recognition [8], atari gaming [9], and fraud detection [10]-[12]. Neural networks created through supervised learning are capable of discovering subtle relationships between variables, making them well suited for creating an alternative model for complex physical systems[1] [13]. Numerous ML algorithms can be employed to build surrogates, but neural networks have several distinct advantages; they can be scaled up to large volumes of high dimensional data, have low memory requirements, and can be easily updated when new data become available [14], [15].

Although it is possible to fit any function to a sufficiently large and shallow neural network [16], studies suggest that deep networks often work better than large networks with similar numbers of neurons [17]. The inclusion of more hidden layers allows higher levels of interaction between parameters, so that deep networks can discover non-linear relationships that may be undetectable with only two hidden layers [17] [1] [13]. Based on this observation, we propose a new model based on deep neural networks technology and machine learning techniques to address the problem of CCF. The proposed model is called ST-BPNN and consists of a pre-processing of machine learning techniques which are Synthetic Minority Oversampling Technique (SMOTE) and Tomek Link to solve the problem of imbalanced data, and back-propagation neural networks (BPNN) to detect fraud. The ST-BPNN is performed on a large set of imbalanced real-world data. We evaluate the effectiveness of ST-BPNN using different criteria such as Recall (sensitivity), AUC-ROC, AUPR, and F1 score, then compare the obtained results with a recent state-of-the-art approach.

The rest of this paper is organized as follows: Section II summarizes the related works, Section III provides details on the proposed model, and Section IV presents the experimental environment and ST-BPNN model implementation. The obtained results are discussed in Section V and findings and future work are summarized in Section VI.

II. Related Work

A comprehensive understanding of fraud detection technologies can be helpful for us to solve the problem of CCF. The work in [18] provides a comprehensive discussion on the challenges and problems of fraud detection research. Adewumi et.al. [19] discuss the most popular fraud types and current nature-inspired detection approaches that are used in the fraud detection system. Also, a significant number of research works have been done on CCFD. The techniques developed can be categorized into two sections, as discussed below:

Machine Learning-based approach: In [1], a survey of different data mining and machine learning techniques for CCFD was presented. The paper summarized a list of challenges one might encounter during CCFD in [20]. In [21], a comparison study of logistic regression and NB was performed. Tax *et al.* [22] explored an intuitive approach that produces random outliers evenly distributed throughout the hypercube containing the target data to assist in the identification of appropriate hyperparameters, and further enhanced it into a hypersphere in order to match the target data better. Weston *et al.* [23] applied peer group analysis on transaction records to identify aberrant values and abnormal transactions. Genetic algorithms combined with scatter search was used to minimize the number of wrongfully classified transactions [24]. Bahnsen *et al.* [25] suggested a cost-sensitive approach with minimal risk of bayes to identify cases of fraud.

Recently, we found many other approaches used in the field of Credit Card Fraud Identification

(CCFI) processes. N. Robinson *et al.* [26] used an approach known as Store Model Divergence for Pre-paid Cards, which uses the HMM of a merchant's terminals to capture fraudulent transactions in real-time. Salvatore Carta *et al.* [27] adopted new intelligence data technology using the PMC (Prudential Multi-Consensus Model). Their method is designed to bring learners together with different scenarios where one class is much smaller than the other classes or where various classification errors are considered in unique ways. Salazar *et al.* [21] studied the performance of their proposed semi-supervised machine learning algorithm to overcome the imbalanced classification problems. They augment the class of limited data to make the variance of the estimate lower by using a method of data subrogation. Then, they investigate the influence of this increase in many simulated and experimental scenarios of an application, for the automatic detection of CCF.

Saia [28] introduced a Discrete Wavelet Transformation (DWT) based approach for fraud detection, by developing an evaluative model with the ability to deal with imbalanced distribution and heterogeneous data. They detected fraudulent activity by exploiting only legitimate transactions through their model definition process, which is affected by less data variation. Furthermore, Saia and Carta [29] employed the Linear Dependence Based (LDB) model and benchmarked its performance versus random forests, which is one of the better known state-of-the-art models. They validated their work by performing the model on two real-world data sets having a strong imbalanced data distribution. In [30], they benefitted from the analysis of an evaluation criterion, in terms of domain frequency, of the spectral pattern of the data. Their method allows obtaining a more stable model to represent information and reduce problems of imbalance and heterogeneity of data.

Salazar *et al.* [31] discussed certain conceptual and empirical solutions after raising the main issues related to the problem of Automatic CCFD (ACCFD). They proposed a framework for ACCFD based on the aggregation of decisions as well as surrogate data. Then, they assessed its sensitivity using various fraud/legitimacy ratios and concluded the paper by suggesting a few areas for further research. Vergara *et al.* [32] has enhanced CCFD's performance with a number of algorithms using signal processing on graphics. They use three approaches: one is a version of standard Iterative Amplitude Adjusted Fourier Transform (IAAFT), and the remaining two are variants of Iterative Surrogate Signals on Graph (ISSG) algorithms. By applying these methods to various scenarios where different proportions of transactions are legitimate and illegitimate, detection skills are enhanced and assessed by Receiver Operating Characteristics (ROC) curves and Key Performance Indicators (KPI), both widely used in the financial aspects of business. Zareapoor *et al.* [33] integrated a sampling technique with a set of AdaBoost to enhance prediction performance on imbalanced data sets. More specifically, through an empirical experiment their technique shows more appropriate performance measures for exploring skewed datasets. Also, in [34] Zareapoor *et al.* developed a balancing strategy to overcome the well-known issues of classification and collection in the CCFD field. They created a contrast vector based on a client's historical behaviors and created a supervised learning model to classify clients. The model, tested on a set of real credit card data provided by FICO, shows significant performance compared to other leading classifiers. In other work, Zareapoor *et al.* [35] presented a hybrid model to handle datasets with a large number of classes, which substitute linear kernel for nonlinear ones without losing accuracy. It was performed on a real-world dataset with 20,000 to 65,000 classes, and it gave significant gains compared to several approaches.

Deep Learning-based approach: Deep learning arises from the idea of a multi-type representation of the human brain that incorporates basic characteristics at low-level or high-level abstractions. People hierarchically arrange their ideas and concepts. People learn simple concepts first then transform them into more abstract concepts. The human brain consists of many layers of neurons that are feature detectors and sense more abstract characteristics when the levels rise, like the deep neural network. It is easier to generalize for computers to interpret information more abstractly.

Artificial Neural Networks (ANNs) for CCFD have been discussed in several pieces of literature [36], [12], [37]. Among the types of ANNs are deep learning and shallow learning; the former has a complex structure with more than one hidden layer and more nodes in each of them than the shallow model. Roy *et al.* [12] and Jurgovsky *et al.* [36] introduced recurrent neural networks, which use a sequence of transactions as input to their model. Besides, Gupta *et al.* [37] compared different machine learning models based on a deep, feedforward neural network. Ogwueleka [38] employed an ANN with a rule-based component, whereas Patidar and Sharma [39] applied an ANN regulated by Genetic Algorithms. Syeda *et al.* [40] implemented a Fuzzy Neuron Network (FNN) running on Parallel Machines to speed up the production of rules for the CCFD's client. Srivastava *et al.* [41] used a Hidden Markov Model (HMM) initially performed on a CCT sequence of cardholders who behaved normally and indicated how the model can be useful for fraud detection. Kamaruddin and Ravi [42] proposed a one-class classification approach to overcome the imbalanced data problem. More specifically, they developed a hybrid system composed of Particle Swarm Optimization (PSO) and Auto Associative Neural Network (AANN) implemented within the Spark Computational Framework (SCF).

Some other studies have investigated the potential for mapping decision trees and randomized forests using neural networks [43], [44]. A particularly useful approach is to map trees in neural networks with two equivalent hidden layers, with the number of neurons in each layer related to the number of leaves in the decision tree [43], [44]. Mapping "warm starts" the process of neural network training by launching the network in a state that works in the same way as the decision tree; after further training, neural networks obtain a higher accuracy than the original tree-based model. Although both hidden layer models work well for medium-sized datasets, the networks can become large enough for high dimensional nonlinear regression problems with complex decision trees, making it difficult for the subsequent training for small datasets. In [33], the back-propagation algorithm is integrated with NB and C4.5 to detect fraud in an imbalanced data-space, generated by minority oversampling with replacement. Padmaja *et al.* [45] presented a fraud detection method that combines backpropagation, naïve Bayes, and C4.5 tree algorithms, and applied them to derived data from oversampling with replacement.

III. The Proposed Approach

Through this section, we present the proposed approach's steps for CCFD. In this approach, we used a fusion of machine and deep learning algorithms to build a CCFD. More specifically, the proposed model is built from the Back-Propagation Neural Networks (BPNNs) to detect CCF and a combination of SMOTE with Tomek links to tackle the imbalanced data problem in order to enhance the model prediction performance of legitimate and fraudulent transactions. In the area of the CCFD, the concept of classifiers combination is proving to be an important new path for improving individual classifiers' performance in terms of accurate and precise results [46]-[48].

A. Credit card fraud detection workflow

Our proposed approach for CCFD, depicted in Fig. 2, is developed by using the Synthetic Minority Oversampling (SMOTE) and Tomek Links (TL) Techniques to tackle the problem of imbalanced data and by using BPNNs model to identify fraudulent transactions. The proposed model is operated on a real-world dataset. It is denoted as ST-BPNN and is composed of the following steps illustrated in Fig. 2.

The ST-BPNN process is performed as follows:

- Preprocessing of imbalanced data using the SMOTE and Tomek links (TL) techniques.
- Fitting the ST-BPNN model using synthetic dataset generated by the SMOTE+TL techniques to improve their classification ability to separate legitimate transactions from fraudulent ones.
- Predicting fraudulent cases by performing ST-BPNN on the original dataset using the K-fold Cross-validation method.
- Evaluating the ST-BPNN prediction performance using AUPR, AUC-ROC, Sensitivity, and F1-scores metrics.

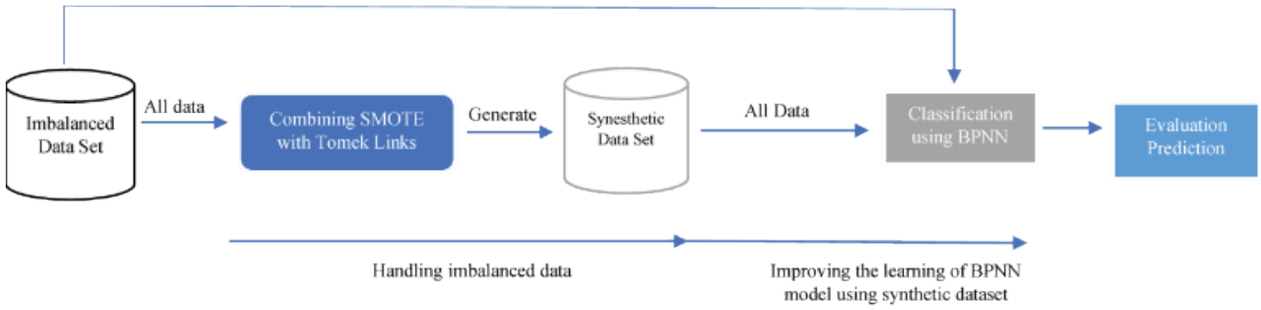

Fig. 2 Workflow of the ST-BPNN model for CCFD.

B. Data description

The used data set [49] to evaluate the performance of the proposed model in detecting fraudulent transactions comes from the European Bank, a dataset that provides transactions that occurred within two days, of which 492 were fraudulent from a total of 284,807 transactions. The data set is highly imbalanced, with positive classes (fraud) representing 0.172% of all transactions. In order to protect customer privacy, it contains only numeric input variables, which are the result of the PCA transformation [50].

Features (or variables) [51] V1, V2, ... V28 are the principal features converted with the PCA, while the ones that are not converted using the PCA are "Time" and "Amount", wherein Time refers to the time interval (in seconds) between both the current and the previous transaction; Amount is the value of the transaction. The target variable (Class) is binary; 1 = fraud, 0 = genuine.

C. Data visualization

Fig. 3 shows that the data set used is very imbalanced; the number of frauds (abnormal transactions) is very low compared to the number of genuine transactions (normal transactions) where the fraud rate is 0.17%. Therefore, this huge difference between the classes (legitimate and fraudulent) can lead to misclassification when detecting CCF.

D. The imbalanced data problem

Class imbalance, also known as the skewed distribution of classes, is a very common classification problem. Special data mining methods are applied along with standard clustering algorithms to deal

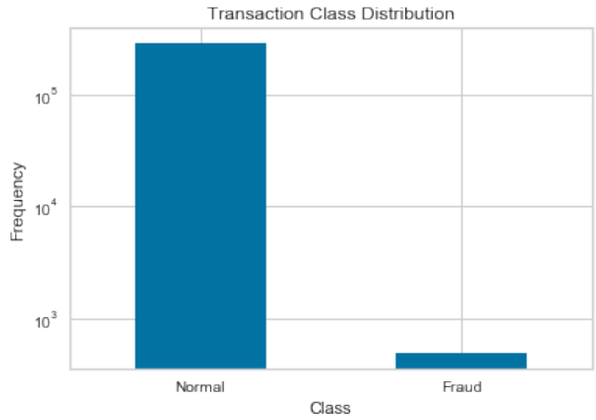

Fig. 3 Transactions distribution based on the target variable (genuine=0, fraud=1).

with this issue. Class imbalance results if one class has a higher number of instances than another. It is more vulnerable when we consider the Big Data context. Indeed, the dataset that is used to train the model contains a very small percentage of the minority class, also known as positive points, versus the majority class, which is known as negative points. The correct classification of the minority class over the majority class is in most cases more challenging and crucial, such as the detection of fraud.

In this case, fraud is the minority class, and it is more critical to identify fraudulent transactions because they are more harmful than normal ones. As a result of these class data ratios, it is very hard for ML classifiers to learn the minority class features and models. Models such as neural networks, decision trees and support vector machines, faced with an unbalanced dataset to detect fraudulent transactions, tend to maximize the overall prediction accuracy at the expense of the minority class [15]. This is due to a strong bias towards the majority class while ignoring the smaller class [19].

E. Synthetic minority oversampling technique

Several suggested approaches to the problem of class imbalance are provided at the data and algorithmic levels. The majority are designed for a two-class or binary problem where one class is strongly under-represented but associated with higher importance of identification. Data-level solutions attempt to rebalance the distribution of classes by resampling the data space, while at the algorithm level solutions essay to adjust the learning algorithm of the existing classifier to reinforce learning by relation to the minority class [17]. To tackle the problem of imbalanced data, we use SMOTE to generate synthetic examples by operating in the functionality space rather than in the data space. The minority class is oversampled by introducing synthetic samples along the line segments combining all or part of the k neighbors closest to the minority class. This technique overcomes the problem of over-sampling and widens the decision region of examples of the minority class, dealing with both a relative and absolute imbalance [52]. Fig. 4 illustrates how the SMOTE algorithm works.

Also, SMOTE as a method usable at the algorithmic level, has the capacity to increment the learning of the algorithm with regard to reducing both the FNR (False Negative Rate) and FPR (False Positive Rate). In the view of Kumari and Mishra [48], SMOTE is written in the following way:

Algorithm 1

1: **Input**: *Minority data* $D^{(t)} = \{x_i \in X\}$ *where* $i = 1,2,\dots,T$
2: Number of minority instances (T), $SMOTE\ percentage$
3: **For** $i = 1,2,\dots,T$ *do*
1: Find the k *nearest (minority class) neighbors of* x_i
2: $\widehat{N} = \left[\frac{N}{100}\right]$
3: **While** $\widehat{N} \neq 0$ *do*
1: Select one of the k *nearest neighbors, call this* $\bar{x}$
2: Select a random number $a \in [0,1]$
3: $\hat{x} = x_i + a(\bar{x} - x_i)$
4: Append $\widehat{N} = \widehat{N} - 1$
4: **End While**
4: **End For**
5: **Output**: *Return synthetic data* S

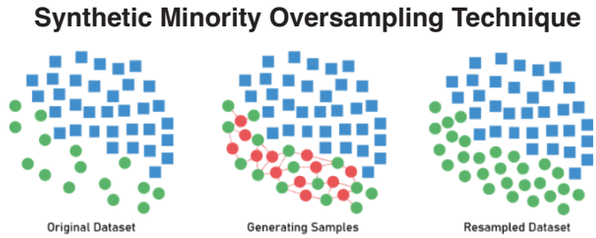

Fig. 4 SMOTE Process [53].

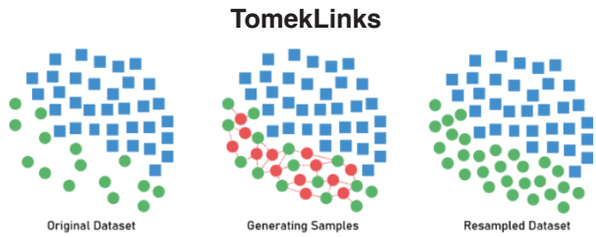

Fig. 5 Tomek Links Process [53].

F. Tomek Links Technique

A combination of Tomek Links and SMOTE is recommended in [54] [48] to exploit the advantages of each approach for tackling the imbalanced data and improving the classification performances of a fraud identification model.

Tomek links, a data cleaning technique, was proposed by Ivan Tomek [54]. Tomek Links (TL) modifies the condensed nearest neighbor process by keeping only limit samples in the condensed subset and thus reduces the computational load. Let S_i, S_j belong to different classes, and ⟦d(S⟧_i,S_j) is the distance between them [54] [55]. A pair ⟦(S⟧_i,S_j) is called a Tomek bond if there is no sample S_1, such as ⟦d(S⟧_i,S_1) < ⟦d(S⟧_i,S_j) or ⟦d(S⟧_j,S_1) <⟦ d(S⟧_i,S_j). The samples that can be considered as Tomek links are borderline or noisy observations and their removal could improve the decision limit of the problem [55]. Fig. 5 illustrates how Tomek Links algorithm works.

G. Deep neural network algorithm for CCFD

Deep Neural Network (DNN) plays an important role in the field of fraud detection with the advantages of self- adaptation, self-organization, better fault-tolerance, and robustness [56].

DNN is developed to simulate the function of the human brain and is built from simple processing units or neurons, which enable the network to learn sets of input-output mappings. It adjusts the weights of the connections in the neural network by

learning samples, aiming to solve nonlinear classification problems [39]. The processing unit or neuron is comprised of a set of synapses or connection links that take input signals, an adder to add input signals, and an activation function that limits the output level of a neuron [39]. Multilayer feed-forward neural networks are a subtype of the neural network distinguished by the presence of hidden layers of neurons. They are particularly well adapted to addressing complex problems, enabling non-linear relationships between input and output layers to be extracted and modelled [57]. Fig. 6 presents a structure example of the Backpropagation Neural Network Topology.

Typically, the backpropagation algorithm is composed of two parts: the forward transmission of information and the backpropagation [39] of error. In the forward transmission process, input information is transmitted through the hidden layers from the layer input to the output one. If the output layer does not get the desired output, calculate the error change value of the output layer, and then turn to reverse propagation and send the error signal back along the original connection path through the network so as to modify each neuron layer's weight until it reaches the required target. Hidden layer output, output layer output, and error function are represented in formulas (1), (2), and (3), respectively.

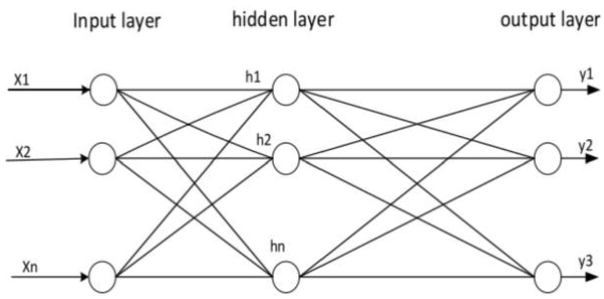

Fig. 6 A structure example of the Backpropagation Neural Network Topology.

$$z_j = f_1\left(\sum_{i=1}^{m} w_{1ij} x_i + b_{1j}\right) \quad (1)$$

$$y_k = f_2\left(\sum_{i=1}^{m} w_{2jk} z_j + b_{2k}\right) \quad (2)$$

$$E = \frac{1}{2} \sum_{k=1}^{n} (y_k - \hat{y}_k) \quad (3)$$

IV. Experimental Environment

This section provides the dataset characteristics, the development environment, the performed strategy, the metrics used to evaluate the classification performance, and the ST-BPNN model implementation.

We combined two balancing techniques (SMOTE with Tomek links) to preprocess the used dataset before performing CCFD through BPNN. Also, we used 10-fold cross-validation in our experiment, and the average prediction result is used for the ST-BPNN model evaluation. In our experiment, 30% of the dataset was randomly dedicated to testing, and the rest was used for training.

TABLE I
Datasets Characteristics

	Features number	Legitimate transactions	Fraudulent transactions	Size (Mo)
Dataset	30	284.315	492	143

A. The development environment

The development environment used to implement the proposed approach presented in this paper is based on the python language where the Scikit-learn libraries [58] are used to implement our proposed model ST-BPNN.

B. Strategy

To respect the transaction chronology, instead of a canonical k-fold cross-validation criterion we used the TimeSeries Split Scikit-learn function [59] to perform a time series cross-validation criterion. Such a function allows us to split our dataset in a series of training and test sets, respecting the transaction chronology. For the experiments, the TimeSeriesSplit method was used with n_splits = 10. The data imbalance problem, previously described in Section III, has been faced during the experiments using the combination of SMOTE+TL techniques.

C. Metrics

According to the considerations made in the imbalanced data problem section, the performance

of the involved algorithms has been evaluated by using various metrics: the Sensitivity, the AUPR, F1-score, and the AUC (i.e., Area Under the ROC Curve). The latter metrics are chosen because they provide information about the performance in terms of fraudulent transactions correctly classified (Sensitivity), a crucial indicator in the context taken into account, and in terms of the effectiveness of the adopted evaluation model (AUC). To evaluate the algorithm performance in terms of correct and incorrect classification of the legitimate transactions, we took into account two additional metrics, which provide specular information concerning the Sensitivity and Precision: the Area under Precision-recall (AUPR).

The formulation of all the aforementioned metrics is presented below:

1) Precision

Precision is a measure that calculates how many positive predictions are correctly identified as positive. It is formulated as follows:

$$\text{Precision}=100\times\frac{(TP)}{(TP+FP)} \quad (1)$$

2) Sensitivity

Sensitivity (Recall) calculates how many positive instances (true labels) are correctly predicted as positive. It is also known as sensitivity or true positive rate. It is formulated as:

$$\text{Sensitivity}=100\times\frac{(TP)}{(TP+FP)} \quad (2)$$

3) F-1score

F1-score is Precision and Recall's weighted average. It is defined as:

$$\text{F1- Score} = 2*(\text{Recall} * \text{Precision})/(\text{Recall}+\text{precision}) \quad (3)$$

4) The Curve of the Area Under the Receiver Operating Characteristic (AUC-ROC)

The AUC-ROC is obtained as a graph of the rate of true positives versus false-positive rates for different decision thresholds. It is mostly used to measure the performance of a classifier to show their capacity in classification in skewed and overlapping data sets. Fig. 7 presents an example of AUC-ROC.

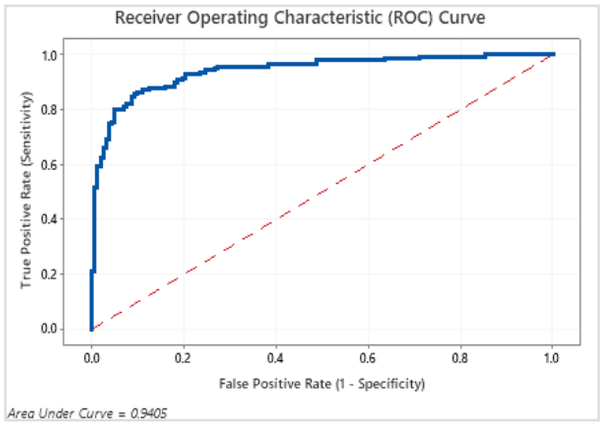

Fig. 7 Example of the AUC-ROC graph [69].

D. ST-BPNN implementation

In this section, the ST-BPNN is built and implemented in Scikit-learn [58], which is a commercial open-source machine learning library. The dataset is divided into a training set and a test set. ST-BPNN learning is performed on the training set and its performance is evaluated on the test set.

The design of the neural network topology is the critical factor affecting the accuracy of the classification system [14], [12]. Adding hidden nodes can increase the accuracy of the network [12]; however, an excessive number of hidden nodes will cause an over-fitting problem, which has a negative impact on generalization, leading to prediction bias; therefore, improving accuracy and generalization requires an adequate number of hidden nodes [60]. There has been no formal theory in determining the number of hidden nodes. The recommendation is based on previous and repeated experiments.

The most efficient network is the one with the same number of nodes in each hidden layer, according to Larochelle *et al.* [60], [61]. In the experiments, we test with a different number of nodes in the hidden layers, and we also get structures that work less well or the structure with an equal number of nodes in the hidden layers. Therefore, we adopt the same number of nodes, such as 4, 10, 16, 22, and 28, in the hidden layers, and we conduct ex-

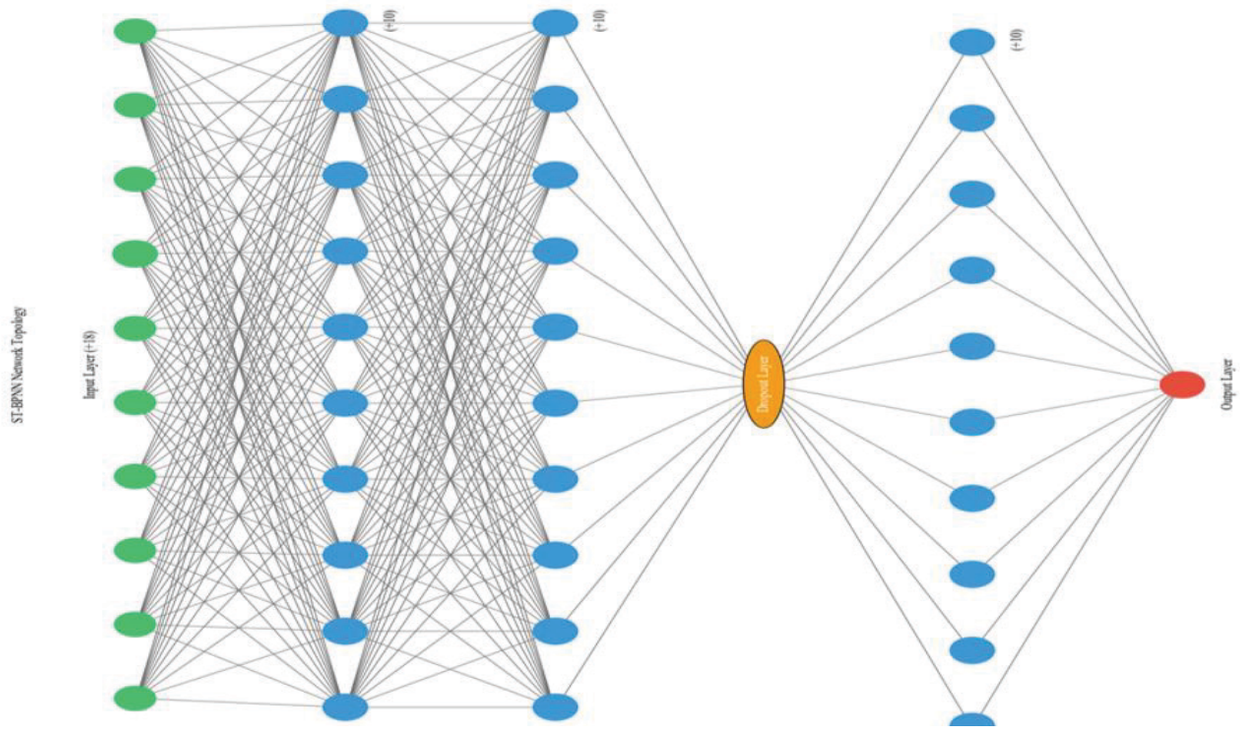

Fig. 8 Network topology of the ST-BPNN model.

periments starting from a small network with one hidden layer and then we extend the network layer by layer up to 6 hidden layers. We performed a test and found that the network with 3 hidden layers with 28 nodes in each hidden layer gave a better result. The network was trained with a learning rate of 0.001 per 450 iterations and a regularization parameter L2 of 0.001. The network topology of ST-BPNN is as shown in Fig. 8.

V. Results and Discussion

In this section, we review the results obtained after the experiments of our proposed approach on a real data set. To perform ST-BPNN, we divide the dataset used into two subsets of data: the first subset of data represents the training set (75% of the original dataset) for ST-BPNN training and a test set (25% of the original dataset) to evaluate its performance. We report the results of the experiments performed by comparing our solution to recent state-of-the-art approaches. Discussions on the results are also highlighted.

A. The results of our model with and without SMOTE+TL

As we have seen in the results presented in Table II, we found that the results of DNN developed using SMOTE+TL techniques (presented in section III) on the training data are better than the results without using SMOTE+TL.

Fig. 9 and 10 present, respectively, the data distribution after using SMOTE+TL techniques and without them. As shown in Fig. 10, we can observe that the numbers of fraudulent transactions that present the minority class variable are multiplied using SMOTE+TL techniques (Explained in section III) as a solution to the imbalanced data problem, while in Fig. 10 the fraudulent transactions are not. After that, we train the developed BPNN model on the synthetic dataset (Fig. 9) where the fraudulent and genuine transactions are balanced in order to increment their learning rate concerning the distinct ability of the proposed model of the fraudulent transaction from the legitimate one.

Comparing two models based on DNN results, the ST-BPNN model (DNN based on SMOTE+TL)

TABLE II
PERFORMANCE RESULTS OF THE BPNN MODEL AFTER AND BEFORE USING SMOTE+TL TECHNIQUES.

	Sensitivity	AUC	AUPR	F1-score	Legitimate	Fraudulent	Total transactions
DNN vbased on SMOTE+TL	1	1	0.99	0.92	284,315	492	284.807
DNN without SMOTE+TL	0.79	0.978	0.83	0.81			

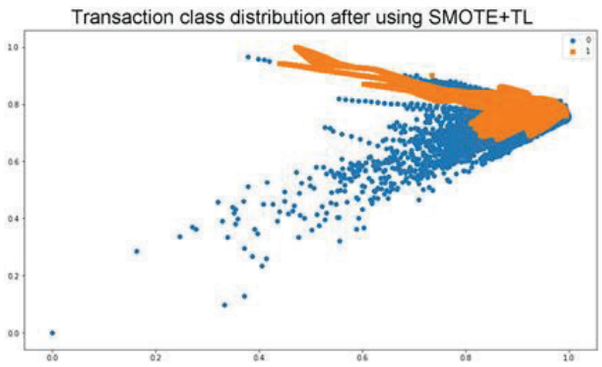

Fig. 9 Transaction class distribution after using SMOTE+TL techniques (1=fraud, 0=genuine).

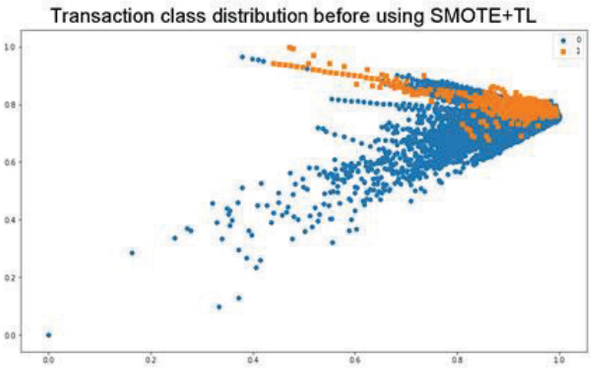

Fig. 10 Transaction class distribution before using SMOTE+TL techniques (1=fraud, 0=genuine).

scores higher in terms of all performance criteria on the test set, where ST-BPNN achieves 99% for AUPR and 100% for both Sensitivity and AUC. Whereas, BPNN (DNN without SMOTE+TL) scores 83% for AUPR, 97.8% for AUC, and 79% for Sensitivity.

As a result, it is concluded that pre-processing (e.g. the process of under-sampling or over-sampling) using SMOTE+TL techniques on the imbalanced training set improves the overall performance of the proposed model to correctly detect fraud operations. Therefore, SMOTE+TL techniques are adopted in this work.

B. Comparison with state-of-the-art approaches

The objective of this subsection is to compare the performance of ST-BPNN with recent studies [29], [30], [62]-[67] on CCFD using the same real-world dataset. Fig. 11 summarizes this comparison.

The results highlighted in Fig. 11, 12, 13, and 14 indicate that the proposed ST-BPNN approach has the potential to improve the performance of a CCFD system in terms of the number of correctly classified fraudulent transactions. This awareness is associated with the sensitivity value (i.e., 100%) which indicates its ability to correctly classify fraudulent transactions more than the best competing algorithm (GS-OCSVM [67], which has a sensitivity value of 97.1%). Also, by following Fig. 12, it is apparent that the ST-BPNN has identified all fraudulent transactions (that are 492 frauds) where the number of Error type 2 (fraudulent transactions classified as legitimate) is 0. Moreover, the results obtained from the ST-BPNN in terms of F1-score is better than the recent related work [64]. ST-BPNN achieves 92% while [64] achieves 83%.

By analyzing Fig. 13, AUPR measurement results highlight the effectiveness of the ST-BPNN model which performs well with a highly imbalanced dataset and has a very good rate of precision and recall (sensitivity) measures; it has 99%, demonstrating its ability to classify new transactions as legitimate or fraudulent. Also, we obtained the same result in terms of AUC (Fig. 11 and 14). Indeed, ST-BPNN reaches 100% as an AUC rate. It retains the high-performance value compared to other models.

In summary, we have proven that in real scenarios characterized by a high data imbalance, the proposed ST-BPNN model can significantly improve a CCFD system, thus reducing losses due to the misclassification of fraudulent events.

It may be noted that the rationale for the ST-BPNN approach, and the reason it works well in the CCFD field, is because legitimate transactions are much higher in number and generally share a similar pattern that is easy to recognize. As a result, several algorithms are able to more accurately assess whether a transaction is legitimate. On the other hand, when a sample is fraudulent, most algorithms give a lower degree of probability on their classification, whether the transaction is legitimate or fraudulent. We have solved this problem in our proposed ST-BPNN algorithm by combining the strengths of SMOTE, Tomek Links techniques with the Back Propagation Neural Networks model, and it is the key to achieving high levels of performance.

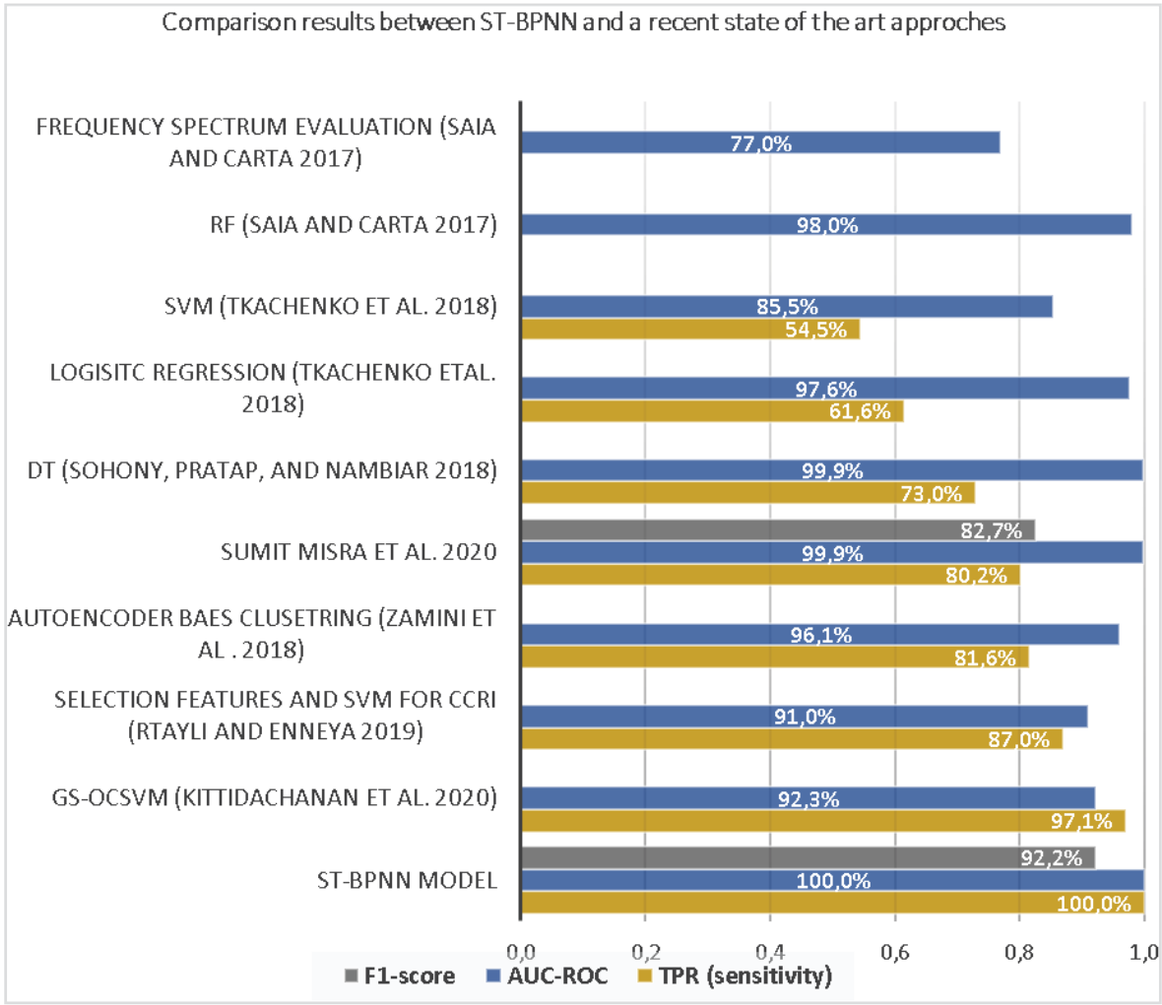

Fig. 11 Comparison results of the ST-BPNN in terms of Sensitivity, AUC-ROC, and F1-score with the recent state-of-the-art approaches.

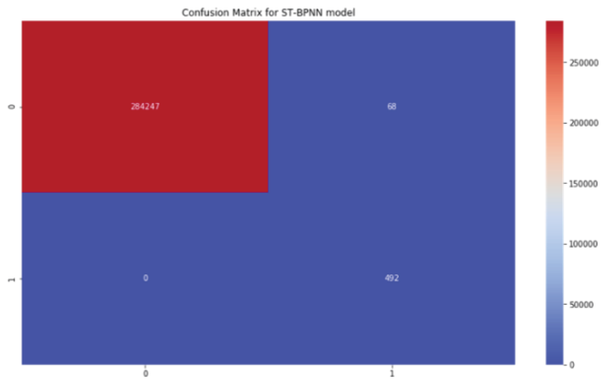

Fig. 12 Confusion matrix for ST-BPNN for CCFD.

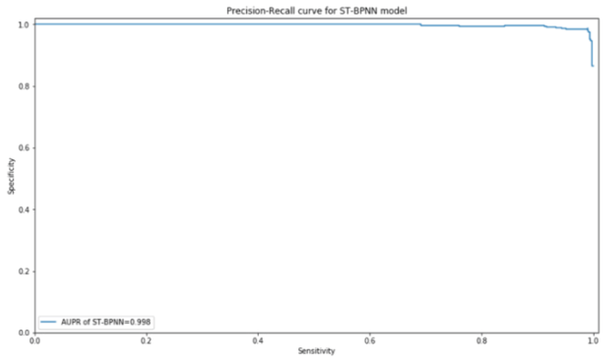

Fig. 13 The precision-recall curve of the proposed model.

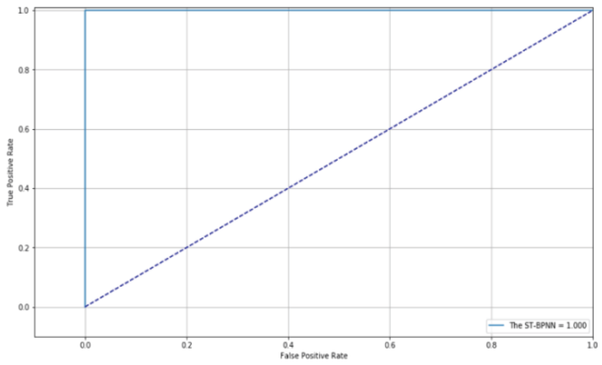

Fig. 14 AUC-ROC of the proposed model.

VI. Conclusion

For over 20 years now, fraud detection research has been in existence and has used a variety of methods ranging from manual verification to end-client authentication. Models of machine learning have also been very successful in this area. Recently, deep learning models have been implemented in many applications, made possible by growing computing power and cost. In this paper, we have built the ST-BPNN model that we propose from two classification methods. The first is a combination of SMOTE with Tomek links techniques to solve the problem of data imbalance, as well as to increase the learning rate of the CCFD model. The second is a deep learning-based model using the backpropagation neural network approach to classify and identify fraudulent transactions from legitimate ones. The model was tested on more than 280.000 transactions obtained from the European bank. The experiments demonstrated that the fusion of machine and deep learning approaches improved the classification performance significantly. Moreover, the findings show that the use of an imbalanced training set by resampling can enhance network performance on the test set. As future work, we expect to study an extended model on the scope of fraud detection in order to build an Adaptive Credit Card Fraud Detection System.

Funding

This article did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of Interest

Authors declare that they have no conflict of interest.

References

[1] R. R. Popat and J. Chaudhary, “A Survey on Credit Card Fraud Detection Using Machine Learning,” in *2018 2nd Int. Conf.* Trend. Electron. *Inf. (ICOEI)*, India, 2018, pp. 1120-1125, doi: 10.1109/ICOEI.2018.8553963.

[2] The Nilson Report, Oct. 2016. [Online]. Available: https://nilsonreport.com/publication_newsletter_archive_issue.php?issue=1096

[1] N. Rtayli, and N. Enneya, “Enhanced credit card fraud detection based on SVM-recursive feature elimination and hyper-parameters optimization,” *J. Inf. Secur. Appl.*, vol. 55, p. 102596, 2020, doi: 10.1016/j.jisa.2020.102596.

[2] “Fifth report on card fraud,” EUROPEAN Central Bank, Sept. 2018. [Online]. Available: https://www.ecb.europa.eu/pub/pdf/cardfraud/ecb.cardfraudreport201809.en.pdf

[3] N. Rtayli and N. Enneya, “Credit Card Risk Detection

based on Feature-Filter and Fraud Identification," in *2019 3rd Int. Conf. Intell. Comput. Data Sciences (ICDS)*, Morocco, 2019, pp. 1-8, doi: 10.1109/ICDS47004.2019.8942373.

[4] Y. Wu *et al.*, "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation," *arXiv:1609.08144*, 2016. [Online]. Available: https://arxiv.org/abs/1609.08144

[5] R. Akmeliawati, M. P. Ooi, and Y. C. Kuang, "Real-Time Malaysian Sign Language Translation using Colour Segmentation and Neural Network," in *2007 IEEE Instrum. Meas. Technol. Conf. IMTC 2007*, Poland, 2007, pp. 1-6, doi: 10.1109/IMTC.2007.379311.

[6] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang and P. S. Yu, "A Comprehensive Survey on Graph Neural Networks," in *IEEE Trans Neural Netw Learn Syst*, vol. 32, no. 1, pp. 4-24, Jan. 2021, doi: 10.1109/TNNLS.2020.2978386.

[7] G.V. de la Cruz Jr, Y. Du, and M. E. Taylor, "Pre-training with non-expert human demonstration for deep reinforcement learning," *Knowl. Eng. Rev.*, vol. 34, 2019, Art. no. e10, doi: 10.1017/S0269888919000055.

[8] J. M. Johnson, and T. M. Khoshgoftaar, "Medicare fraud detection using neural networks," *J. Big Data*, vol. 6, July 2019, Art. no. 63, doi: 10.1186/s40537-019-0225-0.

[9] Y. Lu, "Deep neural networks and fraud detection," Dep. Math., Uppsala Univ., Sweden, U.U.D.M Proj. Rep. 2017:38, 2017.

[10] A. Roy, J. Sun, R. Mahoney, L. Alonzi, S. Adams, and P. Beling, "Deep learning detecting fraud in credit card transactions," in *2018 Syst. Inf. Eng. Des. Symp. (SIEDS)*, USA, 2018, pp. 129-134, doi: 10.1109/SIEDS.2018.8374722.

[11] D. Choi and K. Lee, "An Artificial Intelligence Approach to Financial Fraud Detection under IoT Environment: A Survey and Implementation," *Secur. Commun. Netw.*, vol. 2018, Sept. 2018, Art. no. 5483472, doi: 10.1155/2018/5483472.

[12] A. Fawzi, S. Moosavi-Dezfooli, P. Frossard, and S. Soatto, "Empirical Study of the Topology and Geometry of Deep Networks," in *2018 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, USA, 2018, pp. 3762-3770, doi: 10.1109/CVPR.2018.00396.

[13] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in 15th *Eur. Conf. Comput. Vis.*, in Computer Vision – ECCV 2018, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., in Lucture Notes in Computer Science, vol. 11218, 2018, doi: 10.1007/978-3-030-01264-9_9.

[14] Y. Yu, T. Hur, J. Jung, and I. G. Jang, "Deep learning for determining a near-optimal topological design without any iteration," *Struct. Multidiscip. Optim.*, vol. 59, pp. 787-799, 2019, doi: 10.1007/s00158-018-2101-5.

[15] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, Mar. 2019, Art. no. 27, doi: 10.1186/s40537-019-0192-5.

[16] A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *J. Netw. Comput. Appl.*, vol. 68, pp. 90-113, June 2016, doi: 10.1016/j.jnca.2016.04.007.

[17] A. O. Adewumi and A. A. Akinyelu, "A survey of machine-learning and nature-inspired based credit card fraud detection techniques," *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, pp. 937-953, 2017, doi: 10.1007/s13198-016-0551-y.

[18] A. Walke, "Comparison of Supervised and Unsupervised Fraud Detection," in *Int. Conf. Comput.*, in Advances in Data Science, Cyber Security and IT Applications, A. Alfaries, H. Mengash, A. Yasar, and E. Shakshuki, Eds., in Communications in Computer and Information Science, vol. 1097, 2019, pp. 8-14, doi: 10.1007/978-3-030-36365-9_2.

[19] A. Salazar, G. Safont, and L. Vergara, "Semi-Supervised Learning For Imbalanced Classification Of Credit Card Transaction," in *2018 Int. Jt. Conf. Neural Netw. (IJCNN)*, Brazil, 2018, pp. 1-7, doi: 10.1109/IJCNN.2018.8489755.

[20] D. Tax and R. Duin, "Uniform object generation for optimizing one-class classifiers," *J. Mach. Learn. Res.*, vol. 2, pp. 155-173, 2001.

[21] D. J. Weston, D. J. Hand, N. M. Adams, C. Whitrow, and P. Juszczak, "Plastic card fraud detection using peer group analysis," *Adv. Data Anal. Classif.*, vol. 2, pp. 45-62, 2008, doi: 10.1007/s11634-008-0021-8.

[22] E. Duman and M. H. Ozcelik, "Detecting credit card fraud by genetic algorithm and scatter search," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 13057-13063, 2011, doi: 10.1016/j.eswa.2011.04.110.

[23] A. C. Bahnsen, A. Stojanovic, D. Aouada, and B. Ottersten, "Cost Sensitive Credit Card Fraud Detection Using Bayes Minimum Risk," in *2013 12th Int. Conf. Mach. Learn. Appl.*, 2013, pp. 333-338, doi: 10.1109/ICMLA.2013.68.

[24] W. N. Robinson and A. Aria, "Sequential fraud detection for prepaid cards using hidden Markov model divergence," *Expert Syst. Appl.*, vol. 91, pp. 235–251, 2018, doi: 10.1016/j.eswa.2017.08.043.

[25] 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, 2019, doi: 10.1016/j.jisa.2019.02.007.

[26] R. Saia, "A Discrete Wavelet Transform Approach to Fraud Detection," in *Int. Conf. Netw. Syst. Secur.*, in Network and System Security, Z. Yan, R. Molva, W. Mazurczyk, and R. Kantola, Eds., in Lecture Notes in Computer Science, vol. 10394, 2017, pp. 464-474, doi: 10.1007/978-3-319-64701-2_34.

[27] R. Saia and S. Carta, "A linear-dependence-based approach to design proactive credit scoring models," in *IC3K 2016 - Proc. 8th Int. Jt. Conf. Knowl. Discov. Knowl. Eng. Knowl. Manag.*, Portugal, 2016, pp. 111-120, doi: 10.5220/0006066701110120.

[28] R. Saia and S. Carta, "Evaluating credit card transactions in the frequency domain for a proactive fraud detection approach," in *ICETE 2017 - Proc. 14th Int. Jt. Conf. E-bus. Telecommun.*, Spain, 2017, pp. 335-342, doi: 10.5220/0006425803350342.

[29] A. Salazar, G. Safont, A. Rodriguez, and L. Vergara, "New Perspectives of Pattern Recognition for Automatic Credit Card Fraud Detection," in *Encyclopedia of Information Science and Technology*, M. Khosrow-Pour, Ed., 5th ed. Pennsylvania, USA: IGI Global, 2017, vol. 7, ch. 428, pp. 4937-4950.

[30] L. Vergara, A. Salazar, J. Belda, G. Safont, S. Moral, and S. Iglesias, "Signal processing on graphs for improving automatic credit card fraud detection," in *2017 Int. Carnahan Conf. Secur. Technol. (ICCST)*, Spain, 2017, pp. 1-6, doi: 10.1109/CCST.2017.8167820.

[31] M. Zareapoor and P. Shamsolmoali, "Boosting prediction performance on imbalanced dataset," *Int. J. Inf. Commun. Technol.*, vol. 13, no. 2, pp. 186-195, 2018, doi: 10.1504/IJICT.2018.090556.

[32] M. Zareapoor and J. Yang, "A Novel Strategy for Mining Highly Imbalanced Data in Credit Card Transactions," *Intell. Autom. Soft Comput.*, 2017, doi: 10.1080/10798587.2017.1321228.

[33] M. Zareapoor, P. Shamsolmoali, D. Kumar Jain, H. Wang, and J. Yang, "Kernelized support vector machine with deep learning: An efficient approach for extreme multiclass dataset," *Pattern Recognit. Lett.*, vol. 115, pp. 4-13, 2018, doi: 10.1016/j.patrec.2017.09.018.

[34] J. Jurgovsky *et al.*, "Sequence classification for credit-card fraud detection," *Expert Syst. Appl.*, vol. 100, pp. 234-245, June 15, 2018, doi: 10.1016/j.eswa.2018.01.037.

[35] C. Mishra, D. L. Gupta, and R. Singh, "Credit Card Fraud Identification Using Artificial Neural Networks," *Int. J. Comput. Syst.*, vol. 04, no. 07, pp. 151–159, 2017.

[36] J. Akhilomen, "Data Mining Application for Cyber Credit-Card Fraud Detection System," in *Ind. Conf. Data Min.*, in Advances in Data Mining: Applications and Theoretical Aspects, P. Perner, Ed., in Lucture Notes in Computer Science, vol. 7987, 2013, pp. 218-228, doi: 10.1007/978-3-642-39736-3_17.

[37] R. Patidar and L. Sharma, "Credit card fraud detection using neural network," unpublished.

[38] M. Syeda, Yan-Qing Zhang, and Yi Pan, "Parallel granular neural networks for fast credit card fraud detection," in *2002 IEEE World Congr. Comput. Intell. 2002 IEEE Int. Conf. Fuzzy Syst. FUZZ-IEEE'02. Proc. (Cat. No.02CH37291)*, USA, 2002, pp. 572-577 vol.1, doi: 10.1109/FUZZ.2002.1005055.

[39] A. Srivastava, A. Kundu, S. Sural, and A. Majumdar, "Credit Card Fraud Detection Using Hidden Markov Model," in *IEEE Trans. Dependable Secure Comput.*, vol. 5, no. 1, pp. 37-48, Jan.-March 2008, doi: 10.1109/TDSC.2007.70228.

[40] S. K. Kamaruddin and V. Ravi, "Credit card fraud detection using big data analytics: Use of PSOAANN based one-class classification," in *ACM Int. Conf. Proc. Series*, India, 2016, pp. 1-8, Art. no. 33, doi: 10.1145/2980258.2980319.

[41] G. Biau, E. Scornet, and J. Welbl, "Neural Random Forests," *Sankhya A*, vol. 81, pp. 347-386, 2019, doi: 10.1007/s13171-018-0133-y.

[42] S. Wang, C. Aggarwal, and H. Liu, "Using a random forest to inspire a neural network and improving on it," in *Proc. 17th SIAM Int. Conf. Data Min.*, USA, 2017, pp. 1-9, doi: 10.1137/1.9781611974973.1.

[43] C. Phua, D. Alahakoon, and V. Lee, "Minority report in fraud detection: classification of skewed data," *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 50-59, 2004, doi: 10.1145/1007730.1007738.

[44] A. A. Shah, M. Ehsan, K. Ishaq, Z. Ali, and M. Farooq, "An Efficient Hybrid Classifier Model for Anomaly Intrusion Detection System," *Int. J. Comput. Sci. Netw. Secur.*, vol. 18, no. 11, pp. 127-135, 2018.

[45] F. Carcillo, Y. A. Le Borgne, O. Caelen, Y. Kessaci, F. Oblé, and G. Bontempi, "Combining unsupervised and supervised learning in credit card fraud detection," *Inf. Sci.*, vol. 557, pp. 317-331, 2019, doi: 10.1016/j.ins.2019.05.042.

[46] P. Kumari and S. P. Mishra, "Analysis of Credit Card Fraud Detection Using Fusion Classifiers," in *Proc. Int. Conf. CIDM 2017*, in Computational Intelligence in Data Mining,

H. S. Behera, J. Nayak, B. Naik, and A. Abraham, Eds., in Advances in Intelligent Systems and Computing, vol. 711, 2018, pp. 111-122, doi: 10.1007/978-981-10-8055-5_11.

[47] Raghunath, Feb. 16, 2017, "Credit card Fraud data," distributed by data.world, https://data.world/raghu543/credit-card-fraud-data

[48] G. -h. Li *et al.*, "The flywheel fault detection based on Kernel principal component analysis," in *2019 IEEE 3rd Inf. Technol. Netw. Electr. Autom. Control Conf. (ITNEC)*, China, 2019, pp. 425-432, doi: 10.1109/ITNEC.2019.8729163.

[49] Y. Miao, Z. Ruan, L. Pan, J. Zhang, Y. Xiang, and Y. Wang, "Comprehensive Analysis of Network Traffic Data," in *2016 IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, Fiji, 2016, pp. 423-430, doi: 10.1109/CIT.2016.22.

[50] F. Carcillo, Y.-A. Le Borgne, O. Caelen, and G. Bontempi, "Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization," *Int. J. Data Sci. Anal.*, vol. 5, pp. 285-300, 2018, doi: 10.1007/s41060-018-0116-z.

[51] Y. Charfaoui, "Resampling to Properly Handle Imbalanced Datasets in Machine Learning," medium.com. https://heartbeat.comet.ml/resampling-to-properly-handle-imbalanced-datasets-in-machine-learning-64d82c16ceaa (accessed Oct. 02, 2020).

[52] F. Jobse, "Detecting suspicious behavior in the Bitcoin network," M.S. thesis, Dep. Commun. Inf. Sci., Tilburg Univ., Netherlands, 2017.

[53] Q. Wang, J. Xin, J. Wu, and N. Zheng, "SVM classification of microaneurysms with imbalanced dataset based on borderline-SMOTE and data cleaning techniques," in *9th Int. Conf. Mach. Vis. (ICMV 2016)*, France, 2017, doi: 10.1117/12.2268519.

[54] N. K. Gyamfi and J. Abdulai, "Bank Fraud Detection Using Support Vector Machine," in *2018 IEEE 9th Annu. Inf. Technol. Electron. Mob. Commun. Conf. (IEMCON)*, Canada, 2018, pp. 37-41, doi: 10.1109/IEMCON.2018.8614994.

[55] Y. Safi and A. Bouroumi, "Prediction of forest fires using Artificial neural networks," *Appl. Math. Sci.*, vol. 7, no. 5-8, pp. 271–286, 2013, doi: 10.12988/ams.2013.13025.

[56] I. Stančin and A. Jović, "An overview and comparison of free Python libraries for data mining and big data analysis," in *2019 42nd Int. Conv. Inf. Commun. Technol. Electron. Microelectron. (MIPRO)*, Croatia, 2019, pp. 977-982, doi: 10.23919/MIPRO.2019.8757088.

[57] C. Bergmeir, R. J. Hyndman, and B. Koo, "Validity of Cross-Validation for Evaluating Time Series Prediction," *Comput. Stat. Data Anal.*, vol. 120, pp. 70-83, 2018, doi: 10.1016/j.csda.2017.11.003.

[58] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," *J. Mach. Learn. Res.*, vol. 10, pp. 1-40, 2009, Art. no. 1.

[59] H. Larochelle, D. Erhan, and P. Vincent, "Deep learning using robust interdependent codes," in *Proc. 12th Int. Conf. Artif. Intell. Stat.*, USA, 2009, vol. 5, pp. 312-319.

[60] R. Tkachenko, A. Doroshenko, I. Izonin, Y. Tsymbal and B. Havrysh, "Imbalance Data Classification via Neural-Like Structures of Geometric Transformations Model: Local and Global Approaches," in *Int. Conf. Comput. Sci. Eng. Educ. Appl.*, in Advance in Computer Science for Engineering and Education, Z. Hu, S. Petoukhov, I. Dychka, M. He, Eds., in Advances in Intelligent Systems and Computing, vol. 754, 2018, pp. 112-122, doi: 10.1007/978-3-319-91008-6_12.

[61] I. Sohony, R. Pratap, and U. Nambiar, "Ensemble learning for credit card fraud detection," in *CoDS-COMAD '18 Proc. ACM India Jt. Int. Conf. Data Sci. Manag. Data*, India, 2018, pp. 289-294, doi: 10.1145/3152494.3156815.

[62] S. Misra, S. Thakur, M. Ghosh, and S. K. Saha, "An Autoencoder Based Model for Detecting Fraudulent Credit Card Transaction," *Procedia Comput. Sci.*, vol. 167, pp. 254–262, 2020, doi: 10.1016/j.procs.2020.03.219.

[63] M. Zamini and G. Montazer, "Credit Card Fraud Detection using autoencoder based clustering," in *2018 9th Int. Symp. Telecommun. (IST)*, Iran, 2018, pp. 486-491, doi: 10.1109/ISTEL.2018.8661129.

[64] N. Rtayli and N. Enneya, "Selection Features and Support Vector Machine for Credit Card Risk Identification," *Procedia Manuf.*, vol. 46, pp. 941–948, 2020, doi: 10.1016/j.promfg.2020.05.012.

[65] K. Kittidachanan, W. Minsan, D. Pornnopparath, and P. Taninpong, "Anomaly Detection based on GS-OCSVM Classification," in 2020 *12th Int. Conf. Knowl. Smart Technol. (KST)*, Thailand, 2020, pp. 64-69, doi: 10.1109/KST48564.2020.9059326.

[66] Minitab, "Receiver operating characteristic (ROC) curve for Fit Binary Logistic Model," https://support.minitab.com/en-us/minitab/19/help-and-how-to/statistical-modeling/regression/how-to/fit-binary-logistic-model/interpret-the-results/all-statistics-and-graphs/receiver-operating-characteristic-roc-curve/ (accessed May 09, 2020).