IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 November 2023, accepted 12 January 2024, date of publication 22 January 2024, date of current version 30 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3357091

RESEARCH ARTICLE

Hybrid Undersampling and Oversampling for Handling Imbalanced Credit Card Data

MARAM ALAMRI AND MOURAD YKHLEF
Information System Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

Corresponding author: Maram Alamri (maalamri@ksu.edu.sa)

This work was supported by the "Research Center of the Female Scientific and Medical Colleges," Deanship of Scientific Research, King Saud University.

ABSTRACT Recent developments in the use of credit cards for a range of daily life activities have increased credit card fraud and caused huge financial losses for individuals and financial institutions. Most credit card frauds are conducted online through illegal payment authorizations by data breaches, phishing, or scams. Many solutions have been suggested for this issue, but they all face the major challenge of building an effective detection model using highly imbalanced class data. Most sampling techniques used for class imbalance have limitations, such as overlapping and overfitting, which cause inaccurate learning and are slowed down by noisy features. Herein, a hybrid Tomek links BIRCH Clustering Borderline SMOTE (BCBSMOTE) sampling method is proposed to balance a highly skewed credit card transaction dataset. First, Tomek links were used to undersample majority instances and remove noise, and then BIRCH clustering was applied to cluster the data and oversample minority instances using B-SMOTE. The credit card fraud-detection model was run using a random forest (RF) classifier. The proposed method achieved a higher F1-score (85.20%) than the baseline sampling techniques tested for comparison. Because of the enormous number of credit card transactions, there was still a small false-positive rate. The proposed method improves the detection performance owing to the well-organized balancing of the dataset.

INDEX TERMS Borderline SMOTE, class imbalance, credit card, fraud detection, sampling techniques, Tomek links.

I. INTRODUCTION

The rapid development of e-commerce has resulted in an increase in the number of people shopping online. These customers pay credit cards or use a mobile wallet to make purchases. Consequently, credit cards have become the primary payment method in the online world. Given the massive volume of daily transactions, criminals have innumerable opportunities to find new ways to attack and steal credit card information. Thus, credit card fraud is a serious problem for businesses and can cause significant financial and personal losses. As a result, businesses have increasingly focused on developing new ideas and methods for detecting and preventing fraud, gaining the trust of their customers, and protecting their privacy.

To build an accurate credit card fraud-detection model, transactions should be analyzed in terms of attributes, features, and values. Fraud detection models are computed based on samples of fraudulent and legitimate transactions to classify new transactions as fraudulent or legitimate, respectively. Credit card transactions fall under a vastly imbalanced publicly available dataset. This dataset is highly imbalanced because it contains many more legitimate transactions than fraudulent ones. As a result, classification models can achieve very high accuracy without detecting fraudulent transactions. Classification with imbalanced data is one of the most challenging problems in data mining [1].

Class inequality has received considerable attention in recent years. The learning and classification processes are impacted when one class is significantly more represented than the other. This is particularly the case for a minority class consisting of rarely seen instances, irregular patterns, and abnormal behavior, which is challenging to identify [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Cong Pu.

The best way to handle this type of problem is to balance data using sampling techniques. There are three main approaches to sampling techniques: data, algorithms, and hybrids [2]. The data-level approach can be divided into three categories: over-sampling, undersampling, and hybrid sampling. The most common of these is over-sampling.

Researchers using a data-level approach have attempted to balance datasets prior to the use of conventional classification methods to avoid the influence of the majority class [3]. Researchers adopting the algorithm-level approach have worked on the internal algorithm structure and have attempted to eliminate algorithm sensitivity from the majority class so that the outcomes of classification algorithms do not vary from the majority class [3]. In addition, to tackle the issue of unbalanced data in credit card transactions, recent studies have improved the detection system by combining data-level and algorithm-level approaches in a hybrid approach.

This study focuses on improving data sampling techniques using a data-level approach with hybrid undersampling and oversampling. Hybrid Tomek links and balanced Iterative reducing and clustering using hierarchy (BIRCH) clustering borderline synthetic minority oversampling technology (BCBSMOTE). The goal of the proposed Tomek links and BIRCH BCBSMOTE methods is to enhance data resampling and overcome the limitations of oversampling. The remainder of this paper is organized as follows. Section II reviews related studies on sampling techniques for credit card transaction data. Section III describes the dataset used. While section IV defines the evaluation metrics for the performed experiments. Section V describes the proposed hybrid Tomek links and BCBSMOTE in detail. Section VI reports the experiments performed along with their results and discusses the reported results. Finally, Section VII draws the conclusion of this paper, with a few directions for future work.

II. RELATED WORKS

In the credit card transaction dataset, the number of genuine transactions is higher than the number of fraudulent transactions, which causes a high imbalance in the data. This negatively affects the performance of the fraud-detection model and produces inaccurate results. A recent study [1] highlighted that the issue of highly imbalanced classes is a challenge for credit card fraud detection models. In the area of data mining, prediction involves detecting events, but uncommon events are difficult to identify owing to their inconsistency and variety, and the misclassification of uncommon occurrences can result in significant costs. One solution is to apply sampling methods in the preprocessing stage.

One study [1] investigated the performance of a classification model combining oversampling and undersampling methods for detecting fraud cases from a fraud-detection dataset. It used five oversampling techniques – random oversampling, SMOTE, adaptive synthetic (ADASYN), B-SMOTE, and support vector machine SMOTE (SVMSMOTE) – combined with random undersampling, and evaluated the model using a random forest (RF) classifier. The results showed that a combination of random undersampling (RUS) and one or more oversampling techniques has a high potential to increase accuracy. The study concluded that the combination of oversampling and undersampling techniques positively affected the model's performance compared to individual sampling techniques.

Another study [4] compared five oversampling techniques, SMOTE, ADASYN, B1-SMOTE, B2-SMOTE, and SVMSMOTE, to generate an improved model that could solve the imbalance problem for credit card transaction data. The experiment was conducted using six different machine-learning algorithms: RF, k-nearest neighbors (KNN), logistic regression (LR), naïve Bayes (NB), SVM, and decision trees (DTs). The authors noticed that oversampling techniques improved the performance of the models and claimed that there was no preference for one oversampling technique over another, as everything depended on the type of machine learning (ML) algorithm being used.

Other researchers [5] explored different undersampling techniques using SMOTE and SMOTE-Tomek for unbalanced data. The classification models used in this study (KNN, LR, RF, and SVM) were trained on balanced data to detect fraudulent credit card transactions. The performance of the classifiers on balanced data showed that RF with SMOTE and SMOTE-Tomek were the best. Two other papers, [6] and [7], applied SMOTE-Tomek to a credit card transaction dataset to solve the problem of data imbalance. They found that using SMOTE-Tomek improves the learning rate and outperforms the detection model performance with imbalanced datasets.

Additional study [8] used a hybrid SMOTE and edited nearest neighbor (SMOTE-EEN) method to balance the class distribution in a credit card dataset. The results showed that SMOTE-ENN achieved a high performance with an ensemble deep learning model. The hybrid sampling method improved the performance of the detection model. Moreover, [9] used a hybrid SMOTE-ENN to balance a credit card dataset. SMOTE-ENN is a hybrid resampling method that oversamples minority class samples using SMOTE and removes overlapping instances with ENN. This study discovered that using resampled data enhances the performance of the detection model and concluded that combining SMOTE-ENN with a boosted long short-term memory (LSTM) classifier is a successful approach to detecting fraud in credit card transactions.

Another study [10] proposed SMOTE with adaptive qualified synthesizer selection (ASN-SMOTE), an effective oversampling method based on KNN and SMOTE. The proposed ASN-SMOTE filters noise in the minority class by determining whether the nearest neighbor of each minority instance is related to the minority or majority class. Then, ASN-SMOTE uses the nearest majority instance of each minority instance to correctly perceive the decision boundary within which the appropriate minority instances are selected adaptively

for each minority instance using the recommended adaptive neighbor selection method to synthesize new minority instances. [10] concluded that ASN-SMOTE achieved the best results when compared to nine state-of-the-art oversampling methods.

[11] used a hybrid sampling method of RUS and B-SMOTE to address class imbalance. It was found that this hybrid effectively improved the detection model, with an F1-score of 70%. According to the study, this hybrid overcomes the information loss caused by RUS, as well as the overfitting and overgeneration caused by SMOTE in large datasets [11].

Further study [12] proposed a new undersampling method for handling unbalanced data. This clustering-based noise-sample-removed undersampling (NUS) method removes noise samples from both the majority and minority classes before combining them with undersampling techniques. An experiment was conducted on 13 public and three real-world datasets, and it was found that NUS outperformed several well-known methods, including RUS, SMOTE, ADASYN, SMOTE + Tomek Link, and ENN [12].

Table 1 presents a summary of the selected studies on sampling techniques. These techniques have been used to balance highly skewed credit card transaction data; however, they may result in overlapping and loss of relevant information.

TABLE 1. Summary of sampling techniques in credit card transaction dataset.

Ref	Year	Dataset	Sampling Techniques	Approaches
[8]	2023	Kaggle	SMOTE-ENN	Hybrid approach
[12]	2023	13 public & 3 real world datasets	NUS	Data-level approach
[4]	2022	Kaggle	SMOTE, ADASYN, B1-SMOTE, B2-SMOTE, SVMSMOTE	Data-level approach, Hybrid approach
[5]	2022	Kaggle	Undersampling, SMOTE, SMOTE-Tomek	Data-level approach, Hybrid approach
[6]	2022	Kaggle	SMOTE-Tomek	Hybrid approach
[7]	2022	Kaggle	SMOTE-Tomek	Hybrid approach
[9]	2022	Kaggle	SMOTE-ENN	Hybrid approach
[10]	2022	UCI datasets	ASN-SMOTE	Hybrid approach
[11]	2022	Kaggle	Hybrid Random undersampling and Borderline-SMOTE	Hybrid approach
[13]	2021	Kaggle	Random undersampling	Data-level approach
[1]	2020	Kaggle	Combine Random oversampling, SMOTE, ADASYN, Borderline-SMOTE, and SVMSMOTE with Random undersampling	Data-level approach, Hybrid approach

III. DATASET

Publicly accessible datasets of financial services, particularly in the newly growing field of mobile money transactions, are lacking. Many researchers have worked in the field of fraud-detection value financial datasets. As financial transactions are inherently private, there are no publicly accessible datasets that contribute to the issue at hand. The dataset was created using the PaySim simulator to generate synthetic credit card transactions [13].

Data sets produced by PaySim can help academics, financial institutions, and governmental agencies test their fraud detection techniques or assess the effectiveness of other techniques under comparable testing settings using a shared, openly accessible, synthetic dataset [13]. PaySim generates a synthetic dataset from aggregated data from a private dataset that mimics the normal functioning of transactions and later injects malicious activity to evaluate the performance of fraud-detection algorithms. It replicates mobile money transactions using a sample of genuine transactions collected from a month's worth of financial logs from an African country's mobile money services. The original records were provided by a multinational corporation operating a mobile finance service available in over 14 countries worldwide [13]. The dataset contains 11 attributes and over six million records.

The main reason for the synthetic dataset approach is that the Kaggle dataset used by most researchers is transformed using principal component analysis (PCA), and there are only time and amount attributes. Thus, its attributes are limited and cannot analyze the customer's behavior perfectly; therefore, other attributes are needed, such as those of a synthetic dataset.

Table 2 provides a description of each attribute in the PaySim dataset. For the is-Fraud attribute in this dataset, the fraudulent agents aim to profit by taking control of customers' accounts and trying to empty their funds by transferring them to another account and then cashing out of the system; isFlaggedFraud is an attribute that flags illegal attempts, defined here as any attempt to transfer more than 200,000 USD in a single transaction.

TABLE 2. Dataset attributes.

Attributes	Description
Step	Address a unit of time in the real world. In case 1 step is 1 hour. Total steps: 744 (30 days simulation)
Type	CASH-IN, CASH-OUT, DEBIT, PAYMENT and TRANSFER
Amount	Amount of the transaction
nameOrig	The customer who started the transaction
oldBalanceOrig	Initial balance before the transaction
newBalanceOrig	The new balance after the transaction
nameDest	The customer who is the recipient of the transaction

A. DATASET PREPARATION

The data were prepared by cleaning, a procedure that removes skewed data, outliers, and missing values. The data were preprocessed before feeding them to the model for training to eliminate noise. Random sampling was performed along with selecting informative features, addressing missing values, and structuring.

Missing values: One of the processes of data cleaning is to check whether there are missing or null values in the dataset by using the *isnull ()* method to check these values in the dataset. This method returns the DataFrame object in which all values are replaced with a Boolean value of either 'True' or 'False' (for null values and otherwise). When applied to the PaySim dataset, no null or missing values were found.

Duplicated values: After checking the null values, the next step is to check the duplicated values, which can be performed using the *duplicated ()* method. This method discovers duplicate rows by row throughout the dataset and returns the Boolean values for each row. No duplicated values or false-value returns were found when it was applied to the PaySim dataset.

B. DATASET ANALYSIS

Data analysis was used to investigate the dataset and determine significant information. Exploratory data analysis (EDA) is an important step in gaining complete insight into a dataset [14]. It is performed to evaluate and understand the entire distribution of the data, as well as to determine the correlation and dependency among several input features [14]. Thus, EDA identifies fraud and normal transactions in different transaction types. The relationship between transaction types and fraudulent transactions should also be classified, and the amount of original balance in normal and fraudulent transactions should be defined. The PaySim dataset contains 11 attributes and 636,2620 transaction records. There are 635,4407 normal transactions and 8,213 fraudulent transactions, as shown in Figure 1. It can be observed that there is a high skew in the dataset (which must be balanced using sampling techniques to improve the detection model).

Figure 2 shows that the newbalanceOrig and oldbalanceOrig columns have a very high correlation (almost 1). Thus, for data preprocessing, one of the columns should be dropped. In addition, the isFlaggedFraud column should be dropped because it does not contribute significantly to determining whether a transaction is fraudulent because the flagged algorithm is weak. The number of transactions is the key factor in identifying whether it may be a fraud. The higher the initial balance in the original account before the transaction, the more susceptible it is to fraudulent transactions. The time at which a transaction occurs is also related to the likelihood of the transaction being fraudulent.

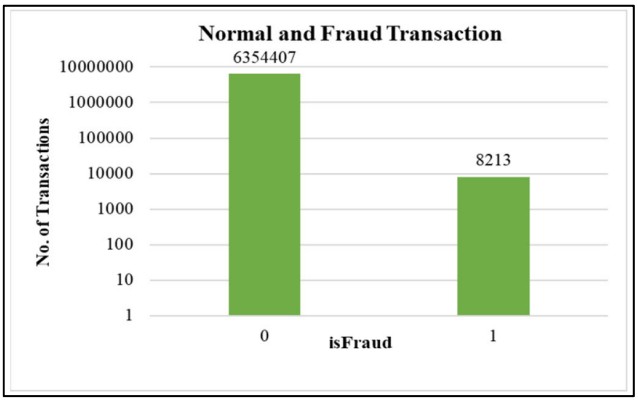

FIGURE 1. Numbers of normal and fraud transactions in PaySim dataset.

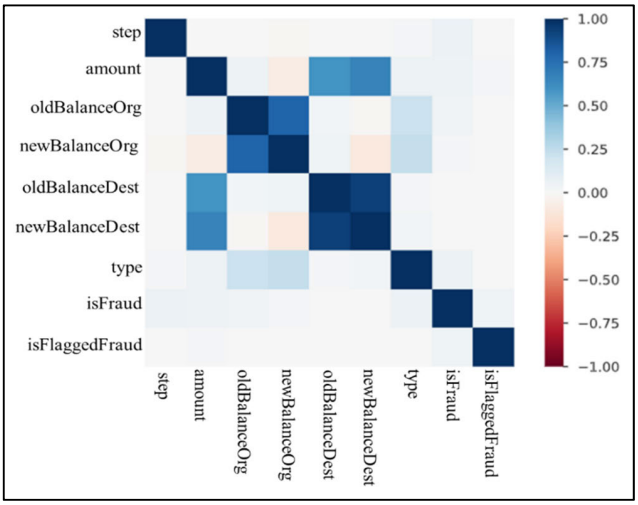

FIGURE 2. Correlation matrix for PaySim dataset.

There are five types of transactions in the PaySim dataset. As Figure 3 shows, the cash-out and payment types are the

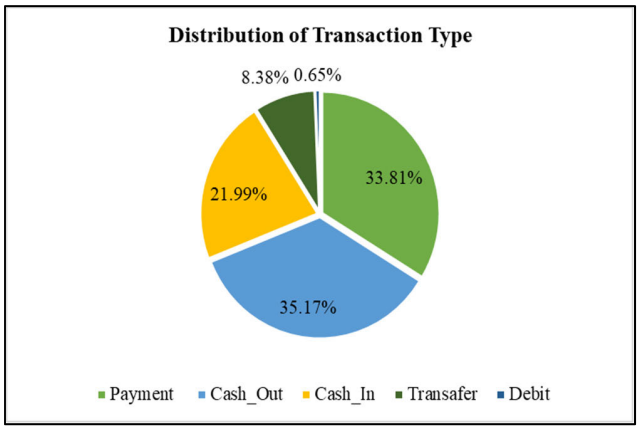

FIGURE 3. Proportions of transaction types in PaySim dataset.

most common types of transactions, while debit and transfer are the least common.

From Figure 4, it can be observed that the only fraudulent transactions are transfers and cash outs. The numbers of fraudulent transactions for these two types are very similar.

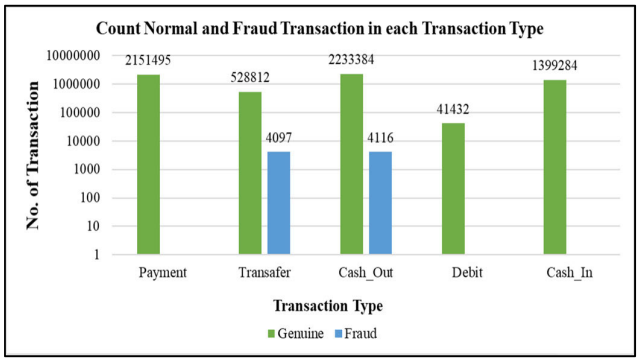

FIGURE 4. Number of normal transactions and fraudulent transactions for each type of transaction.

IV. EVALUATION METRICS

To validate and test the credit card fraud detection model, the test dataset was processed to validate that it produced correct results based on the evaluation metric. The evaluation of ML algorithms is generally performed using different metrics, such as accuracy, precision, recall, F1-scores, area under the receiver operating characteristic curve (AUC-ROC), and area under the average precision and recall curve (AUPRCC).

The confusion matrix is used to assess the performance of a classification model [15]. It displays the numbers of true positives, false positives, and false negatives. True positives are cases in which the model correctly predicts a positive outcome, whereas true negatives are those in which the model correctly predicts a negative outcome [15]. The number of false positives is the number of instances in which the model predicts a positive outcome but the actual outcome is negative. The number of false negatives is the number of instances in which the model predicts a negative outcome but the actual outcome is positive [15].

The accuracy, precision, and recall metrics are described with respect to the confusion matrix in Table 3. Accuracy is the most obvious measure of a model's predictive ability. The numerator in this measure contains all correctly labelled positive and negative class instances (TP: fraud; TN: or non-fraud) [16]:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad (1)$$

Precision, also known as the positive predictive value, is the proportion of true positives to predicted positives generated by a model. A precision value of 1 indicates that all predicted positive instances are indeed positive (FP: incorrectly classified fraud transactions) [15].

$$Precision = (TP)/(TP + FP) \quad (2)$$

Recall, also known as the true-positive rate, is the proportion of predicted positives to all positive instances in the sample. A recall value of 1 indicates that all positive samples were correctly identified (FN: incorrectly classified non-fraud transactions)) [15].

$$Recall = (TP)/(TP + FN) \quad (3)$$

TABLE 3. Confusion matrix.

		Actual Class	
		Positive (Fraud)	**Negative (Non-Fraud)**
Predicted Class	**Positive**	True positive (TP)	False positive (FP)
	Negative	False negative (FN)	True negative (TN)

For a classification task and imbalanced dataset, the F1-score is the harmonic mean of the precision and recall values. The F1-score was calculated as follows:

$$F1 = 2 \times (Recall \times Precision)/(Recall + Precision) \quad (4)$$

State-of-the-art sampling techniques [4], [6], [9], [10], such as SMOTE, B-SMOTE, ADASYN, SMOTE-Tomek, and SMOTEEEN, were used as a baseline for comparison. The evaluation was performed by testing these algorithms using the synthetic dataset PaySim and comparing them with respect to increasing the true positives and reducing the false positives and error rates, along with the ability to handle a large balanced dataset and gain higher accuracy and f1-scores.

V. HYBRID SAMPLING TECHNIQUES FOR BALANCING DATASET

This section introduces the sampling method used for the highly skewed credit card dataset. It presents a hybrid sampling technique for balancing credit card datasets using

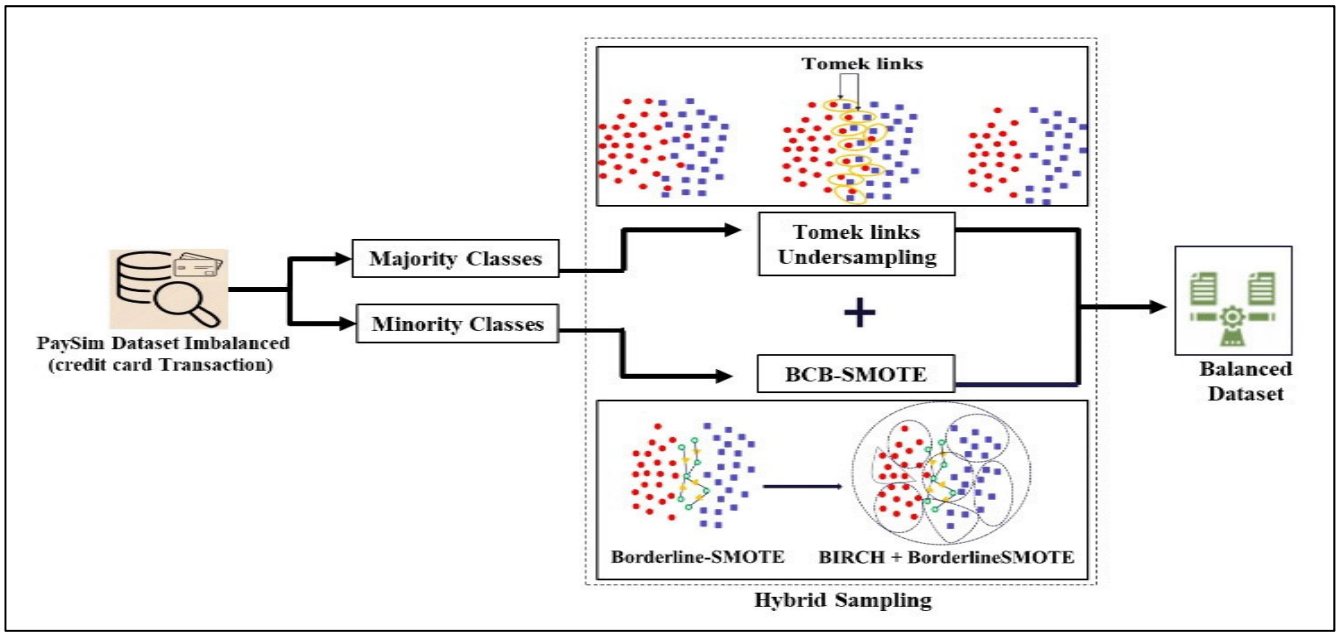

FIGURE 5. The proposed method for hybrid sampling to balance the PaySim dataset.

Tomek links for undersampling, combined with BIRCH clustering and Borderline-SMOTE for oversampling. In addition, the evaluation and results of the proposed method are discussed. In addition, the results of the proposed method were compared with those of the latest existing sampling techniques.

A. METHODOLOGY

Class imbalance problems are common in fraud-detection problems, as there are always less fraud data than non-fraud data. This problem has a negative impact on the algorithm because classifiers are frequently biased towards the majority class and produce poor performance results as a result. In real credit card transactions, there is a highly imbalanced distribution of examples, with the minority class usually being much smaller than the majority class. In most learning algorithms, the data distribution is assumed to be balanced and oriented towards the learning and recognition of the majority class. Consequently, minority samples were incorrectly classified [17].

In recent years, the learning problem of imbalanced datasets has been extensively studied, and sampling methods have been developed to solve this problem by balancing the data distribution, mainly by oversampling the minority class or undersampling the majority class [17]. The most popular oversampling method is the SMOTE. SMOTE has been improved in various applications, such as the neighborhood cleaning rule (SMOTE-NCL) [17], deep attention (DA-SMOTE) [18], B-SMOTE [19], SMOTE-ENN [20], and SMOTE-Tomek links [21].

The PaySim dataset used here is highly skewed, which is a major characteristic of financial transaction datasets. Thus, a data sampling approach was adopted using hybrid undersampling (Tomek links) and oversampling (BCB-SMOTE) as illustrated in Figure 5; we have shown in the literature review that the hybrid sampling approach is proved to outperform the result compared with the undersampling or oversampling separately.

To balance the PaySim dataset, the dataset was split into two sets: the training set and test set for the evaluation of the proposed method. The training set comprised 80% of the entire dataset and 20% of the test set. The proposed hybrid undersampling and oversampling used the training set, whereas the test set was used with an RF model. As shown in Figure 6, after splitting the dataset, the training set contained 508,3526 normal transactions and 6,570 fraud transactions that were used to train the model. Thus, the remainder of the dataset comprises 1,269,709 normal transactions and 1,643 fraud transactions to be used to test the model.

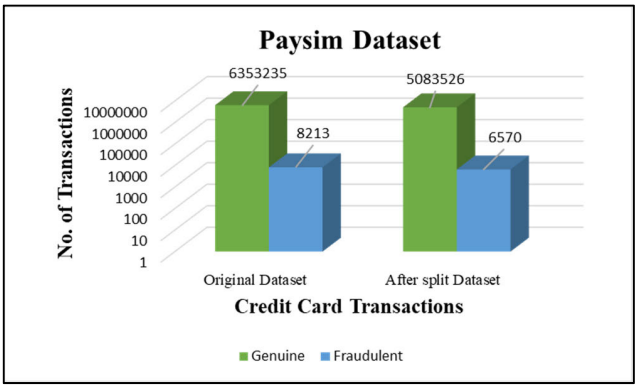

FIGURE 6. The PaySim dataset divided into train and test sets.

B. UNDERSAMPLING USING TOMEK LINKS

Tomek links are essentially data-reduction techniques. They are an improved version of nearest neighbor rule (NNR). Tomek links can be used as undersampling or data cleaning methods. The idea here was to use Tomek links to undersample the majority class by removing Tomek links, although samples from both the majority and minority classes were eliminated rather than only those from the majority class that form Tomek links [22].

The primary goal of this technique is to first identify two samples, x, which belongs to the majority class and y, which belongs to the minority class, where x and y form a Tomek link [23]. Two conclusions can be drawn regarding the samples used in the Tomek links. Either one of them has noise or an unwanted sample, which is regarded as a less crucial sample case, or both are boundary values [23]. If one is a noise-generating or unwanted sample, either the minority class sample or majority class sample can be eliminated from the dataset. An alternative is to eliminate both samples that have been utilized as boundary values [23]. The primary benefit of Tomek links is that they emphasize noise cancellation. One of the samples in the Tomek links is likely to be located in the cluster of the other sample when there are two data samples, one belonging to the majority class and the other to the minority. Another benefit of Tomek links is that they do not change the rest of the dataset, thereby decreasing the possibility of losing crucial data [23].

The goal of this study was to undersample the majority class with particular emphasis on removing the sample of the majority class from the Tomek links. The majority class was more likely to produce noise or unwanted samples because of the large sample size in this study, as the PaySim dataset has six million records. This was implemented using Algorithm 1 [23].

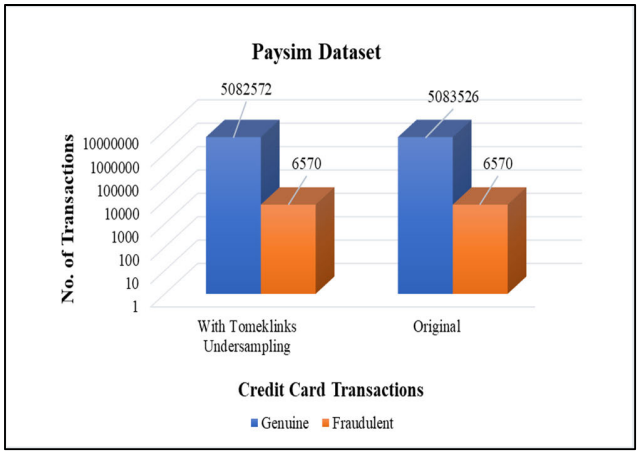

FIGURE 7. Applying Tomek links undersampling to the PaySim dataset.

As illustrated in Figure 7, the majority of the sample classes in the dataset decreased after applying the Tomek link undersampling by removing the noise generator samples. Thus, the dataset was ready for oversampling using BCBSMOTE.

Algorithm 1 Tomek Links Undersampling Algorithm

1. Consider a sample dataset with a majority and minority class.
2. For each sample, let 'x' be in the majority class and repeat steps 3 to 7.
3. Find the nearest neighbour of 'x' in the entire dataset.
4. Let 'y' be the nearest neighbour of 'x'.
5. If 'y' belongs to the majority class, go to step 3 for the next sample.
6. Calculate the nearest neighbour of 'y'; let 'z' be the nearest neighbour of 'y'.
7. If 'z' and 'x' are the same sample points, then 'x' and 'y' are nearest neighbours of each other.
8. Thus, 'x' and 'y' form Tomek links.
9. Remove 'x' from the sample dataset.
10. Repeat from step 3 until there are no further modifications, or no sample is removed.
11. The updated sample dataset will work as the dataset for classification.

C. OVERSAMPLING USING BCBSMOTE

The second part of the proposed hybrid sampling method uses clustering techniques to cluster the data based on distance metrics and to identify the data points that belong together. To ensure that the noise was oversampled, the minimum size of any cluster was restricted the minimum size of any cluster, that is, clusters with fewer data points than cluster size were simply be dropped from the overall data. Subsequently, oversampling was applied to each cluster to generate the samples. Each cluster generates a specific subset of the total samples that need to be generated. All data points generated by the clusters were combined and generated by clusters and combined with the original dataset. The proposed method improves on the shortcomings of the B-SMOTE algorithm (global perspective) by locally and adaptively clustering the minority class samples to form clusters and then generating samples from each local cluster to improve the imbalance within and between classes.

1) BIRCH

BIRCH is a fast clustering method for incremental clustering that uses a tree structure [24]. This algorithm is appropriate for large samples and introduces data points measured from multiple dimensions incrementally and dynamically to create a cluster with the best possible quality within memory and time constraints. It operates sufficiently quickly to complete clustering with only one dataset scan [24]. The two key features of BIRCH are clustering features (CF) and clustering feature trees. These hierarchical trees have three different types of nodes: non-leaf nodes, leaf nodes, and minimum clusters [25]. The CF number parameters were as follows: B is the largest child node containing a non-leaf node, L is the maximum value of the smallest cluster contained in the leaf, and T is the maximum diameter of the smallest cluster [25].

The following phases comprise the BIRCH algorithm (Fig. 8). The CF number was first established in memory for clustering after reading each data point [25]. The tree must be rebuilt from the leaf node if all memory is used [26]. The first phase involved removing outliers, merging adjacent clusters, and filtering the CF tree. Second, the dataset was reduced by creating a smaller CF tree to condense the data [25]. Third, to produce a better CF tree, global clustering such as K-means or agglomerative clustering is used to group all CF clusters. Fourth, to ensure that errors are fixed, cluster refinement is completed by rescanning the original raw data [26]. The CF tree issue produced when the original data are scanned only once is fixed by cluster refining [25].

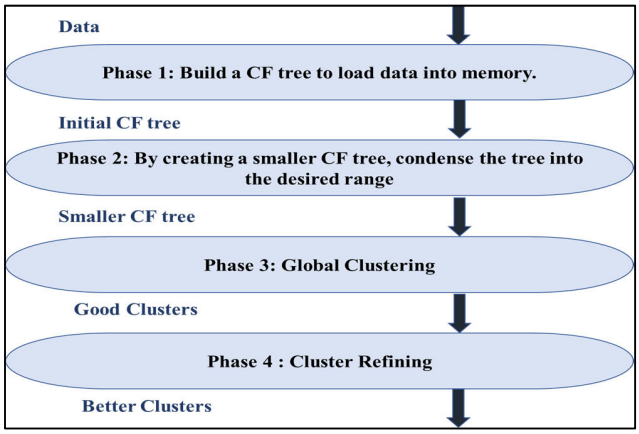

FIGURE 8. The BIRCH phases.

2) BORDERLINE-SMOTE

B-SMOTE emphasizes minority samples that are near majority samples. The ADASYN and this algorithm are comparable [27]. The algorithm creates its closest neighbors from a few minority samples. Only those minority samples that have the priority of their nearest neighbors in the majority class are kept. A subsequent SMOTE-like step is applied using minority samples that have been preserved [27], as shown in Algorithm 2.

3) BCBSMOTE

BIRCH clustering is used with B-SMOTE to oversample minority sample classes essentially by clustering the data and then applying B-SMOTE to each cluster to generate more samples (each cluster will generate a specific ratio of the whole data to be generated). Initially, BIRCH clusters all minority classes over the training data to identify the data points that belong together. The BIRCH algorithm has several crucial parameters: threshold, which is the maximum number of data samples to be condensed in the sub-cluster of the leaf node in the CF tree; branching factor, which is the factor used to specify the number of CF sub-clusters that can be made in a node; and n cluster, which is the number of clusters. In this experiment, the n cluster was set to (cluster size)2, the size of the data cluster.

Algorithm 2 The Borderline-SMOTE Algorithm

1. Compute the closest m samples from the available dataset for each sample in a few classes x_i. m' denotes the number of additional categories in the most recent samples.
2. Organize the samples x_i:

   If $m' = m$, the samples around x_i are all from distinct categories and are referred to be noise data. As such data will have a negative impact on the generation effect, it is recommended that these samples not be included in the generation.

   If $m/2 \leq m' < m$, more than half of the m surrounding x_i samples are of distinct categories. Define Danger as the border sample.

   If $0 \leq m' < m/2$, more than half of the surrounding m samples of x_i are of the same categories, designated as Safe.
3. After marking, apply the SMOTE method to enlarge the Danger samples. Select x_i from the Danger dataset samples and compute k-nearest neighbour samples of the same kind x_{zi}. New samples x_n are generated at random using the formula

$$x_n = x_i + \beta(x_{zi} - x_i)$$

Where β is a random number between 0 and 1.

First, the cluster is performed using BIRCH over the training data to determine which data points belong together; then, there will be a dictionary giving the number of data points in each cluster. Subsequently, the number of samples generated by each cluster was calculated. If a cluster has more than five data points, the number of clusters responsible for generating samples is determined. Finally, B-SMOTE was applied to each cluster to generate samples according to the contribution of the cluster. The steps of the algorithm were applied here, as shown in Algorithm 3.

D. RF CLASSIFIER

To evaluate the proposed hybrid sampling method, the PaySim credit card dataset was classified using the RF algorithm after balancing. A group of DT classifiers comprises RF. Compared to DTs, it has the advantage of correcting overfitting. To train each tree, a random subset of the training set was sampled [28]. Next, a DT is built, in which each node is divided into a chosen feature of a random subset of the functionality. Because each tree in the RF is trained independently of the others, it is extremely quick to train datasets with many features and data instances. The RF algorithm is resistant to overfitting and offers a good estimate of the generalization error [28]. These steps were applied in the present case.

The main benefit of using RF in this study was that it required minimal training time compared to the other

Algorithm 3 BIRCH Clustering Borderline SMOTE (BCB-SMOTE) Algorithm

Input:

X_train: Feature matrix of the training data.
y_train: Target labels for the training data.
cols: List of column names.
cluster size: Minimum cluster size for BIRCH clustering.

Output:

generated_dataset: DataFrame containing the original and synthetic data.

1. Employ BIRCH clustering on *X_train* with *y_train*= 1 to identify clusters.
2. Identify minority class clusters with a size equal to or greater than *cluster size*.
3. Based on the percentage of samples in each cluster, calculate how many samples it should be generating.
4. For each selected cluster, generate samples using B-SMOTE as per cluster contribution calculated in step 3.
5. Update the *generated_dataset* by adding synthetic data.
6. Return the final *generated_dataset.*

algorithms. The F1-score is essential for balancing dataset evaluation, the accuracy of predicting credit card fraud is extremely important, and RF predicts the output with great precision, even for large datasets [28].

E. EVALUATION

Although accuracy is a crucial measurement and standard in conventional classification evaluation measures, it is not applicable in the classification of imbalanced data because, even if a prediction is inaccurate, it will still be highly accurate because of the infrequency of items in the minority sample. Consequently, researchers have proposed effective methods for assessing imbalanced dataset indicators [11]. In this study, five metrics were used to evaluate the performance of the proposed method: accuracy, F1-score, recall, precision, and AUPRC. These metrics were based on a confusion matrix. Four of these – F1-score, recall, precision, and AUPRC (i.e., not accuracy) – were utilized to evaluate the performance of the proposed hybrid sampling method.

The AUPRC provides the area under precision and recall for several thresholds [29]. This is a plot of precision versus recall, which corresponds to the false discovery rate curve. It is simple to compare various classification models using the AUPRC, which summarizes the precision-recall curve [30]. The AUPRC value of the perfect classifier was 1. The system's high recall and precision produce results with accurate labels [30]. The AUPRC metric examines the positive predictive value and true positive rate, making it more sensitive to improvements for the positive class (fraud class) [31].

This study compared the hybrid Tomek links-BCBSMOTE algorithm with three oversampling algorithms, SMOTE, B-SMOTE, and ADASYN, as well as two hybrid sampling techniques, SMOTEENN and SMOTE-Tomek using the PaySim dataset. To demonstrate that the balanced dataset created by the hybrid Tomek links-BCBSMOTE algorithm was valid and stable, an RF classification model was employed for testing.

VI. RESULTS AND DISCUSSION

This section reviews the results of the experiment of the proposed method on the PaySim dataset. The dataset was divided into two subsets – the training and the test sets, which comprised 80% and 20% of the original dataset, respectively in order to evaluate the performance of the hybrid sampling technique using an RF classifier by contrasting our results with those of other, widely-used state-of-the-art sampling methods, the outcomes of the experiments are reported.

Table 4 provides detailed information on the performance measurements for all the applied methods. The proposed method had the highest F1-score (85.20%), precision (81.27%), and AUPRC (72.77%). The accuracies of all the sampling methods were similar (99.90-99.95%). The proposed and B-SMOTE methods had the highest accuracy (99.95%) however, their recall metrics were lower than those of the other sampling methods. The precision of the proposed method (81.27%) was higher than that of B-SMOTE and other sampling techniques. Thus, the proposed hybrid Tomek links the BCBSMOTE sampling method outperforms other sampling methods.

TABLE 4. Performance evaluation.

RF with	Accuracy	F1-score	Recall	Precision	AUPRCC
SMOTE	99.92%	77.71%	97.07%	64.78%	62.89%
B-SMOTE	99.95%	84.67%	89.47%	80.37%	71.92%
ADASYN	99.92%	76.98%	97.80%	63.46%	62.07%
SMOTE-Tomek	99.92%	77.81%	97.44%	64.76%	63.11%
SMOTEEN	99.90%	71.65%	97.26%	56.72%	55.17%
Hybrid Tomek links BCBSMOTE (Proposed)	99.95%	85.20%	89.53%	81.27%	72.77%

Table 5 presents the confusion matrix for the PaySim dataset obtained using the RF classifier for balancing using the proposed method. It can be observed that TP is high, which results in a recall value (89.53%), and FP is low, which results in a precision value (81.27%). Precision and Recall are an important evaluation metrics used in fraud detection. Their significance is based on their ability to minimizing FP rate and detect positive cases respectively. However, there is trade-off between precision and recall. Increasing recall may lead to decrease precision. Thus, F1-score and AUPRC is considered to provide comprehensive evaluation of the model performance.

TABLE 5. RF confusion matrix using proposed sampling method (Tomek Links BCBSMOTE).

		Actual Class	
		Positive (Fraud)	**Negative (Non-Fraud)**
Predicted Class	**Positive**	True positive (TP) 1270542	False positive (FP) 172
	Negative	False negative (FN) 339	True negative (TN) 1471

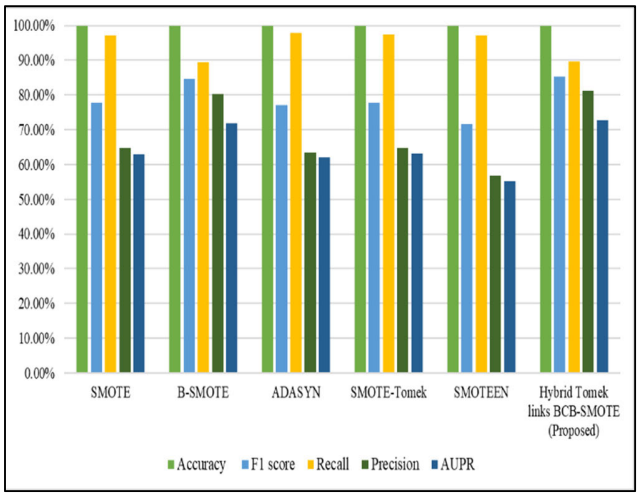

FIGURE 9. Comparison between state-of-art sampling methods and proposed method.

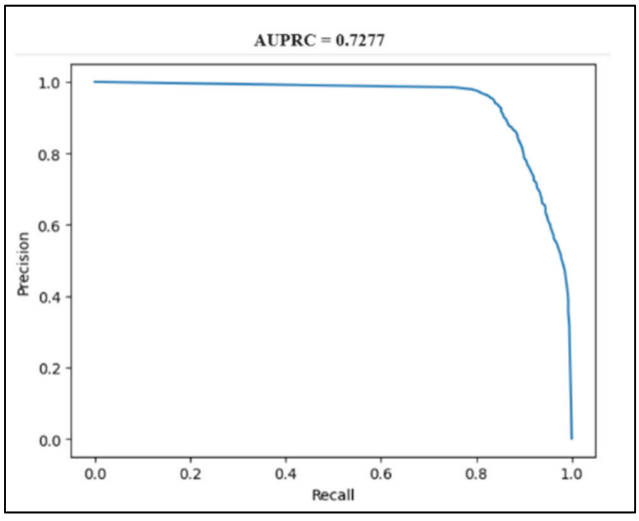

FIGURE 10. AUPRC for hybrid Tomek links BCBSMOTE.

Figure 9 shows that the proposed hybrid method achieved better results than the other methods tested by reducing errors. The B-SMOTE performance was also better than that of other sampling techniques. With regard to the F1-score, precision, and AUPRC, hybrid Tomek links BCBSMOTE achieved the highest values, and SMOTEEEN achieved the lowest. In terms of recall, B-SMOTE achieves the lowest value, while ADASYN achieves the highest.

The AUPRC metric illustrates the trade-off between precision and recall in a binary classification model, especially when dealing with imbalanced datasets. It provides an in-depth evaluation of the model's ability to distinguish between positive and negative instances. Here, the AUPRC for the proposed hybrid Tomek link BCBSMOTE method is shown with recall plotted on the x-axis and precision on the y-axis (Figure 10).

VII. CONCLUSION

The daily use of bank credit cards has grown dramatically along with technological innovations. As a result, the use of credit cards fraudulently by others is a new offense that is expanding quickly. Therefore, detecting and preventing these attacks has become an active field of research. Credit card fraud detection encounters challenges owing to an imbalanced dataset, which causes inaccurate results through the detection system. This study presents a hybrid sampling technique to balance the PaySim credit card transaction dataset. The proposed method uses hybrid Tomek links to undersample majority samples and BCBSMOTE to oversample minority samples. This method takes advantage of the Tomek links method to remove noise samples and of BIRCH clustering in B-SMOTE to cluster a large dataset and eliminate overfitting. It outperformed existing state-of-the-art methods in terms of the F1-score, precision, and AUPRC metrics.

In the future, optimization-based feature engineering for detecting customer spending behavior will be applied to increase the F1-score and decrease the false-positive rate in credit card fraud detection model.

ACKNOWLEDGMENT

This research was supported by a grant from the "Research Center of the Female Scientific and Medical Colleges," Deanship of Scientific Research, King Saud University.

REFERENCES

[1] H. Shamsudin, U. K. Yusof, A. Jayalakshmi, and M. N. A. Khalid, "Combining oversampling and undersampling techniques for imbalanced classification: A comparative study using credit card fraudulent transaction dataset," in *Proc. IEEE 16th Int. Conf. Control Autom. (ICCA)*, Oct. 2020, pp. 803–808, doi: 10.1109/ICCA51439.2020.9264517.

[2] W. W. Soh and R. Yusuf, "Predicting credit card fraud on a imbalanced data," *Int. J. Data Sci. Adv. Anal.*, vol. 1, no. 1, pp. 12–17, Apr. 2019. [Online]. Available: http://ijdsaa.com/index.php/welcome/article/view/3

[3] P. Kaur and A. Gosain, "Comparing the behavior of oversampling and undersampling approach of class imbalance learning by combining class imbalance problem with noise," in *Advances in Intelligent Systems and Computing*. Singapore: Springer, 2017, pp. 23–30, doi: 10.1007/978-981-10-6602-3_3.

[4] R. Qaddoura and M. M. Biltawi, "Improving fraud detection in an imbalanced class distribution using different oversampling techniques," in *Proc. Int. Eng. Conf. Electr., Energy, Artif. Intell. (EICEEAI)*, Nov. 2022, pp. 1–5, doi: 10.1109/EICEEAI56378.2022.10050500.

[5] K. Praveen Mahesh, S. Ashar Afrouz, and A. Shaju Areeckal, "Detection of fraudulent credit card transactions: A comparative analysis of data sampling and classification techniques," in *Proc. J. Phys., Conf.*, Jan. 2022, vol. 2161, no. 1, Art. no. 012072, doi: 10.1088/1742-6596/2161/1/012072.

[6] N. Rtayli, "An efficient deep learning classification model for predicting credit card fraud on skewed data," *J. Inf. Secur. Cybercrimes Res.*, vol. 5, no. 1, pp. 57–71, Jun. 2022, doi: 10.26735/tlyg7256.
[7] S. O. Akinwamide, "Prediction of fraudulent or genuine transactions on credit card fraud detection dataset using machine learning techniques," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 6, pp. 5061–5071, Jun. 2022, doi: 10.22214/ijraset.2022.44962.
[8] Q. Li and Y. Xie, "A behavior-cluster based imbalanced classification method for credit card fraud detection," in *Proc. 2nd Int. Conf. Data Sci. Inf. Technol.* New York, NY, USA: ACM, Jul. 2019, pp. 134–139, doi: 10.1145/3352411.3352433.
[9] E. Esenogho, I. D. Mienye, T. G. Swart, K. Aruleba, and G. Obaido, "A neural network ensemble with feature engineering for improved credit card fraud detection," *IEEE Access*, vol. 10, pp. 16400–16407, 2022, doi: 10.1109/ACCESS.2022.3148298.
[10] X. Yi, Y. Xu, Q. Hu, S. Krishnamoorthy, W. Li, and Z. Tang, "ASN-SMOTE: A synthetic minority oversampling method with adaptive qualified synthesizer selection," *Complex Intell. Syst.*, vol. 8, no. 3, pp. 2247–2272, Jun. 2022, doi: 10.1007/s40747-021-00638-w.
[11] E. F. Ullastres and M. Latifi, "Credit card fraud detection using ensemble learning algorithms MSc research project MSc data analytics," M.S. thesis, Nat. College Ireland, Dublin, Ireland, May 2022.
[12] H. Zhu, M. Zhou, G. Liu, Y. Xie, S. Liu, and C. Guo, "NUS: Noisy-sample-removed undersampling scheme for imbalanced classification and application to credit card fraud detection," *IEEE Trans. Computat. Social Syst.*, pp. 1–12, Mar. 2023, doi: 10.1109/TCSS.2023.3243925.
[13] E. A. Lopez-Rojas, A. Elmir, and S. Axelsson, "PaySim: A financial mobile money simulator for fraud detection," in *Proc. 28th Eur. Modeling Simulation Symp. (EMSS)*, Sep. 2016, pp. 249–255.
[14] A. A. Arfeen and B. M. A. Khan, "Empirical analysis of machine learning algorithms on detection of fraudulent electronic fund transfer transactions," *IETE J. Res.*, pp. 1–13, Mar. 2022, doi: 10.1080/03772063.2022.2048700.
[15] I. A. Mondal, Md. E. Haque, A.-M. Hassan, and S. Shatabda, "Handling imbalanced data for credit card fraud detection," in *Proc. 24th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2021, pp. 1–6, doi: 10.1109/ICCIT54785.2021.9689866.
[16] A. Alharbi, M. Alshammari, O. D. Okon, A. Alabrah, H. T. Rauf, H. Alyami, and T. Meraj, "A novel text2IMG mechanism of credit card fraud detection: A deep learning approach," *Electronics*, vol. 11, no. 5, p. 756, Mar. 2022, doi: 10.3390/electronics11050756.
[17] Y. Sun and F. Liu, "SMOTE-NCL: A re-sampling method with filter for network intrusion detection," in *Proc. 2nd IEEE Int. Conf. Comput. Commun. (ICCC)*, Oct. 2016, pp. 1157–1161, doi: 10.1109/COMPCOMM.2016.7924886.
[18] H. Mansourifar and W. Shi, "Deep synthetic minority over-sampling technique," Mar. 2020, *arXiv:2003.09788*.
[19] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," in *Proc. Int. Conf. Intell. Comput.*, 2005, pp. 878–887, doi: 10.1007/11538059_91.
[20] S. Choirunnisa and J. Lianto, "Hybrid method of undersampling and oversampling for handling imbalanced data," in *Proc. Int. Seminar Res. Inf. Technol. Intell. Syst. (ISRITI)*, Nov. 2018, pp. 276–280, doi: 10.1109/ISRITI.2018.8864335.
[21] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, pp. 20–29, Jun. 2004, doi: 10.1145/1007730.1007735.
[22] A. Abd El-Naby, E. E.-D. Hemdan, and A. El-Sayed, "An efficient fraud detection framework with credit card imbalanced data in financial services," *Multimedia Tools Appl.*, vol. 82, no. 3, pp. 4139–4160, Jan. 2023, doi: 10.1007/s11042-022-13434-6.
[23] A. Bansal and A. Jain, "Analysis of focussed under-sampling techniques with machine learning classifiers," in *Proc. IEEE/ACIS 19th Int. Conf. Softw. Eng. Res., Manage. Appl. (SERA)*, Jun. 2021, pp. 91–96, doi: 10.1109/SERA51205.2021.9509270.
[24] C.-R. Wang and X.-H. Shao, "An improving majority weighted minority oversampling technique for imbalanced classification problem," *IEEE Access*, vol. 9, pp. 5069–5082, 2021, doi: 10.1109/ACCESS.2020.3047923.
[25] X. Xiong, Y. Huang, Y. Zhang, F. Zhang, Y. Jia, and J. Xi, "Adaptive hybrid sampling algorithm based on BIRCH clustering," in *Proc. IEEE 5th Inf. Technol.,Netw.,Electron. Autom. Control Conf. (ITNEC)*, vol. 5, Oct. 2021, pp. 110–114, doi: 10.1109/ITNEC52019.2021.9587242.
[26] M. M. Barham, "An improved BIRCH algorithm for breast cancer clustering," M.S. thesis, Dept. Comput. Sci., Fac. Inf. Technol., Middle East Univ., Amman, Jordan, Jun. 2020.
[27] F. de la Bourdonnaye and F. Daniel, "Evaluating resampling methods on a real-life highly imbalanced online credit card payments dataset," Jun. 2022, *arXiv:2206.13152*.
[28] F. Tadvi, S. Shinde, D. Patil, and S. Dmello, "Real time credit card fraud detection," *Int. Res. J. Eng. Technol.*, vol. 8, no. 5, p. 2021, 2021.
[29] V. S. S. Karthik, A. Mishra, and U. S. Reddy, "Credit card fraud detection by modelling behaviour pattern using hybrid ensemble model," *Arabian J. Sci. Eng.*, vol. 47, no. 2, pp. 1987–1997, Feb. 2022, doi: 10.1007/s13369-021-06147-9.
[30] V. Arora, R. S. Leekha, K. Lee, and A. Kataria, "Facilitating user authorization from imbalanced data logs of credit cards using artificial intelligence," *Mobile Inf. Syst.*, vol. 2020, pp. 1–13, Oct. 2020, doi: 10.1155/2020/8885269.
[31] 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, Dec. 2020, Art. no. 102596, doi: 10.1016/j.jisa.2020.102596.

MARAM ALAMRI received the B.S. and M.S. degrees in computer sciences and telecommunications and information systems from the University of Essex, Essex, U.K., in 2014 and 2015, respectively, where she is currently pursuing the Ph.D. degree with the Department of Information Systems, King Saud University (KSU). She is also a Lecturer with the Department of Information Systems, KSU. Her current research interests include data sciences, artificial intelligence, and cyber security.

MOURAD YKHLEF received the B.Eng. degree in computer science from Constantine University, Algeria, the M.Sc. degree in artificial intelligence from Sorbonne Paris Nord University (previously Paris 13), France, and the Ph.D. degree in computer science from Bordeaux 1 University, France. He is currently a Professor with the Department of Information Systems, King Saud University (KSU), Riyadh, Saudi Arabia. His main research interests include data science and artificial intelligence.