mathematics

MDPI

Article

The Imbalanced Classification of Fraudulent Bank Transactions Using Machine Learning

Alexey Ruchay [1,2,*], Elena Feldman [2], Dmitriy Cherbadzhi [2] and Alexander Sokolov [1]

[1] Department of Information Security, South Ural State University (National Research University), Chelyabinsk 454080, Russia
[2] Department of Mathematics, Chelyabinsk State University, Chelyabinsk 454001, Russia
[*] Correspondence: ran@csu.ru

Abstract: This article studies the development of a reliable AI model to detect fraudulent bank transactions, including money laundering, and illegal activities with goods and services. The proposed machine learning model uses the CreditCardFraud dataset and utilizes multiple algorithms with different parameters. The results are evaluated using Accuracy, Precision, Recall, F1 score, and IBA. We have increased the reliability of the imbalanced classification of fraudulent credit card transactions in comparison to the best known results by using the Tomek links resampling algorithm of the imbalanced CreditCardFraud dataset. The reliability of the results, using the proposed model based on the TPOT and RandomForest algorithms, has been confirmed by using 10-fold cross-validation. It is shown that on the dataset the accuracy of the proposed model detecting fraudulent bank transactions reaches 99.99%.

Keywords: bank transactions; imbalanced classification; detection of fraudulent transactions; machine learning

MSC: 91-10

check for updates

Citation: Ruchay, A.; Feldman, E.; Cherbadzhi, D.; Sokolov, A. The Imbalanced Classification of Fraudulent Bank Transactions Using Machine Learning. *Mathematics* **2023**, *11*, 2862. https://doi.org/10.3390/math11132862

Academic Editors: Yongzeng Lai and Junseok Kim

Received: 6 April 2023
Revised: 15 June 2023
Accepted: 21 June 2023
Published: 26 June 2023

1. Introduction

The modern financial sector requires high security standards due to rapid technological progress [1,2]. Banks utilize many security measures for transactions, including artificial intelligence (AI), neural networks, and machine learning [3–6]. The use of these techniques allows advanced technological problems to be solved. The most common types of financial frauds are money laundering, identity fraud, and credit and debit card frauds [1].

The key role in securing financial systems is anti-money-laundering (AML), while identifying illegal transactions [7–9]. The launch of Bitcoin (BTC) has created a paradox: the anonymity allows criminals to remain unknown; however, forensic or AML analysis can be performed with access to the BTC transaction database. The goal of AML analytics is to identify fraudulent transactions in massive, ever-growing datasets. Semi-automatic or manual transaction analysis produces a large number of errors while the success of machine learning methods shows great potential in AML analysis [10–12].

Current AI methods are not sufficient to process the online transaction data stream. To analyze different security methods, a set of transactions filtered by specially created rules needs to be used. All transactions passing through the filter are considered legitimate and do not get analyzed further. The filter parameters include the direct characteristics of the transaction or its parties with the exclusion of any indirect characteristics. The results of this process can be invalid; thus, the most important obstacles in AML are [1] the inability to make real-time online selections of parameters that are unique to specific financial activities, the lack of a centralized transaction analysis tool, a large volume of false positives for fraudulent transactions, and the high cost of manual analysis.

The current task is to develop a system to identify suspicious transactions. Such a system would help speed up and simplify data processing, decrease the risks of transactions, store information and data about criminals, provide credit score verification, and trace the users involved in money laundering. Due to the high cost of these systems, only large financial institutions can afford them. In 24 h, large companies process up to 20 million transactions [13]. Unfortunately, manual analysis identifying suspicious activities often produces false results and requires extra time and resources in order for it to be performed correctly. Therefore, the development and implementation of an antifraud system is essential [14–17].

In past studies, the identification of fraudulent transactions using machine learning algorithms has been primarily accomplished via classification [18–28]; however, traditional algorithms are unreliable and inadequate for precise classifications. This study describes the development of a reliable machine learning model to detect fraudulent bank transactions. We use the CreditCardFraud dataset [29], which contains more than 200,000 credit card transactions.

In this article, the model is used to identify fraudulent transactions using the following algorithms: linear regression, logistic regression, quadratic regression, naive Bayes classifier, k-nearest neighbors algorithm, multi-layer perceptron (MLP), discriminant function analysis, quadratic discriminant analysis (QDA), decision trees, random forest, support vector machine (SVM), and adaboost. The article describes the optimization methods of hyperparameters for machine learning algorithms to increase classification reliability.

Imbalanced classification is an important machine learning problem. Class imbalance is primarily used in fraud detection, medical diagnostics, pollution detection, and remote sensing. Class imbalance emerges due to the few examples gathered from the minority class,or by a natural imbalance in the underlying probability distribution of the data. Most classification algorithms are based on the balanced distribution of class labels.

Our goal is to maximize the percentage of fraudulent transactions detected (the recall for the fraud class) while potentially sacrificing the percentage of predicted frauds that turn out to be actual frauds (the precision for the fraud class) and to maximize the percentage of correctly classified transactions (the accuracy).

The CreditCardFraud dataset is imbalanced (492 are fraudulent and 284,315 are legitimate) [29]. Including the Tomek links resampling algorithm increased the classification reliability of fraudulent BTC transactions in the imbalanced class condition [30].

The main contributions of this article are insights into machine learning methods which can provide the reliable classification of fraudulent bank transactions and an improvement in the classification of fraudulent bank transactions by using data preprocessing, hyperparameter optimization, and a resampling algorithm.

The remainder of this article is organized as follows: Section 2 discusses the requirements of the proposed model, Section 3 provides a detailed description of the model, Section 4 describes the evaluation process, Section 5 discusses the proposed model, and Section 6 concludes.

2. Related Work

It is important for banks to identify suspicious transactions before fraud happens. As an example the Nilson report states that cumulative damages from credit card fraud were USD 22.8 billion in 2016 and USD 28.7 in 2019 [31].

All international credit card payment systems use their own fraud detection methods. Often, all of the transactions are checked using antifraud rules, lists, and filters. In Russia these systems are also used to identify cyber criminals; however, the first priority is to find the illegal exchange of goods and services.

Bank transaction security has been improved due to the addition of a built-in EMV chip to credit and debit cards and the client authentication process, by means of two-step authentication. Despite this, the number of fraudulent bank transactions has not reduced: credit card fraud totals EUR 1.5 billion a year [14]. Experts agree that to reduce credit card fraud using authentication methods—the development and implementation of the fraud identification systems based on data analysis—is required.

Experts classify credit card fraud into different forms, for example, fraud with false identity or using behavioral biometrics. In the first case, a criminal applies for a credit card with false identification, while in the case of behavioral biometrics, a criminal obtains and uses the cardholder's existing credit card credentials. Fraudulent bank transactions are divided into six categories [14]: lost or stolen card fraud, fake card fraud, online fraud, bankruptcy fraud, retailer fraud, and stolen in transit card fraud. In addition, bank transaction fraud can be classified by three categories: bankcard fraud, retail fraud, and Internet fraud.

Recently, it has been suggested to distinguish two categories of fraud: fraud with a card (face-to-face) and fraud without a card (e-commerce fraud). Bank transaction fraud can be categorized into five different types [14]:

- Lost and stolen cards (<1% of all fraudulent transactions). Most of the time elderly people are the victims in this category; fraudsters get the PIN by "shoulder-reading" and subsequently steal the card. In this case, the fraudster is the thief and the credit card does not go through an organized crime resale network.
- Cards that do not reach their destination (<1% of all fraudulent transactions). In this case the credit card may have been stolen during production or upon delivery by mail. To avoid this type of fraud, banks ask customers to obtain and activate the card at their office.
- Identity theft. A card obtained by using fake or stolen documents.
- Counterfeit cards (>10% of all fraudulent transactions). The card is copied using a genuine card or during database hacking, and then reproduced on a counterfeit plastic card by international organized crime groups. The criminal obtains and reproduces the magnetic stripe data of the cards. This type of fraud had been prevalent in the past but has been partially solved by EMV technology as magnetic stripe terminals are no longer used in the EU; however, they remain in Asia and America. This kind of non-contact payment is not appealing for criminals, since only small value payments are processed.
- Fraud without a card makes up over 90% of all fraudulent transactions. Most credit card fraud occurs during electronic transactions. The card number, expiration date, and CVC are extracted during database hacks, then these credentials are sold. The value of the credentials depends on the re-sale capabilities (the first digits of the card numbers represent the bank and, accordingly, the blocking policies they have). Many retailers (90%) use 3-D Secure. This technology protects the cardholder through two-step authentication; however, some large retailers, such as Ebay or Amazon, do not protect their users' transactions with 3-D Secure. Another problem hindering the fight against cardless fraud is that companies do not report attacks that result in a data breach since it can cause bad publicity, which might lead to financial losses.

Typical fraud detection systems have several control levels. Each level can be automated or managed manually. Part of the automated approach consists of machine learning algorithms. These algorithms are used to create predictive models based on annotated bank transactions. Over the past ten years regular credit card fraud research has made it possible to develop models based on supervised or unsupervised machine learning, and partial machine learning [17].

The authors analyze methods of money laundering detection based on Big Data [32]. Big Data is represented in complex systems that perform tasks to prevent legalization and laundering of money that has been obtained illegally (the SAS Anti-Money Laundering System, SAS AML).

Machine learning methods for binary classification to predict fraudulent transactions using multilayer perceptrons, random forest, logistic regression, and convoluted network graphs are used to detect anomalous BTC transactions [10]. Convoluted network diagrams have emerged as a potential tool for AML analysis, and they are particularly attractive as a new way to capture and analyze transactions. The results reflect the advantage of the random forest method of classification, though the F1 score of 0.796 is not sufficiently reliable for classifying abnormal BTC transactions.

Recent studies built deep neural networks for the detection of irregularities either by discriminatively mapping normal samples and abnormal samples to different regions of the feature space, or by fitting multiple different distributions [33–37].

The effect of high cardinality attributes on credit card fraud detection has been investigated, and the Value Clustering for Categorical Attributes algorithm was used to reduce the cardinality of their domains while preserving fraud-detection capabilities [38].

Using machine learning to predict fraudulent financial transactions without balancing data, as in real life there are usually many more honest than fraudulent ones, an accurate and reliable method is proposed. Additional fraudulent transaction samples were generated using a conditional generative adversarial network for tabular data (CTGAN) to make the classifier more robust and accurate [39].

By combining a nature-inspired hyperparameter setting with several supervised classifier models implemented in a modified version of the XGBoost algorithm, a unique hybrid technique for financial payment fraud detection is presented [40]. Using machine learning and domain data to identify fraudulent payments, a modified XGBoost model can be created and tested by tuning. Experiments showed that the model successfully classified with 99.64% accuracy.

An AE-PRF fraud detection method has been proposed which uses AE to reduce the dimensionality of the data and extract the features of the data [23]. In addition, the method uses RF with a probabilistic classification to classify the data as fraudulent together with the corresponding probability. AE-PRF outputs a final classification as fraudulent if the corresponding probability exceeds a predefined probability threshold.

An AED-LGB algorithm has been proposed to solve bank credit card fraud [24]. The AED-LGB algorithm first extracts the feature data using an autoencoder. The features are then fed into the LightGBM algorithm for classification and prediction.

3. The Proposed Model

Figure 1 presents a flowchart of the proposed model. First, the CreditCardFraud data are preprocessed using outlier reduction, missing value reduction, repetition reduction, data normalization, imbalance resampling, and data splitting. The preprocessed data are divided into a training set and a test set. The training set data are used for model training by machine learning and, during training, the hyperparameters are constantly optimized to obtain the trained model. Afterward, the testing data are used to evaluate the trained model by classifying the data into fraudulent data or legitimate data.

3.1. The CreditCardFraud Data

We use the CreditCardFraud dataset [29], which is imbalanced (492 transactions are fraudulent and 284,315 are legitimate). The experiment consisted of a resampling algorithm, which increased the classification reliability of fraudulent bank transactions in an imbalanced class condition [29].

To train the model to detect fraudulent transactions, it is necessary to prepare a large dataset. The analysis should be carried out with bank transaction data; however, there are no suitable datasets available in the public domain. The only option is the CreditCardFraud dataset containing transactions made by European credit card holders in the fall of 2013. This dataset contains bank transaction over the course of two days. During this period 492 fraudulent credit card transactions were recorded out of 284,807 transactions. The positive class (fraud) is 0.172% of total transactions, so the dataset is imbalanced.

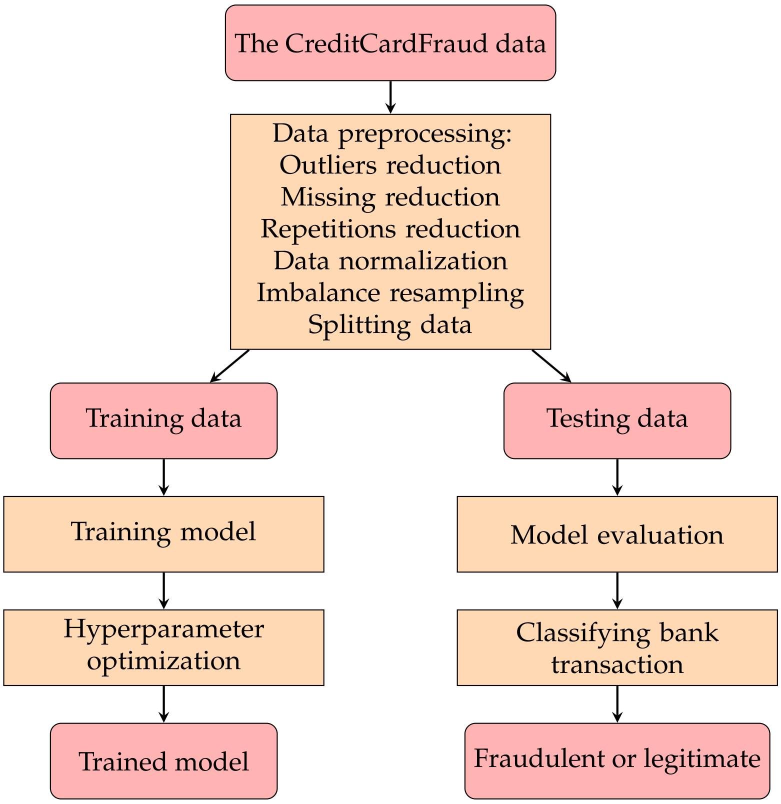

Figure 1. A flowchart of the proposed model.

The CreditCardFraud dataset contains only numeric features. These features are the result of converting the original values using the principal component method. The original functions and additional information could not be obtained to maintain confidentiality. Table 1 shows some of the features of the dataset used in this experiment. Features $V1, V2, \ldots, V28$ are the main components obtained using the method of principal components. Attributes "Time" and "Amount" are the only ones publicly available. "Time" of dataset contains seconds between the first and all of the following transactions. "Amount" lists the summary of transactions. The correct answer is tagged as "Class", and this attribute equals 1 in a case of fraudulent transaction and 0 in a legitimate transaction case.

Table 1. The CreditCardFraud dataset [29].

Time	V1	V2	V3	V27	V28	Amount	Class
0.0	1.191857	0.266151	0.166480	−0.008983	0.014724	2.69	0
1.0	−1.358354	−1.340163	0.379780	−0.055353	−0.059752	378.66	0
1.0	−0.966272	−0.185226	1.792993	0.062723	0.061458	123.50	0
2.0	−1.158233	0.877737	1.548718	0.219422	0.215153	69.99	0

3.2. Data Preprocessing

The most important step for machine learning is data preprocessing, because the models are based on machine learning algorithms which increase the dataset reliability. Since the parameters have a different range of values and units of measurement, they need to be normalized before the training of the model to make sure that all the parameters inside the model contain all of the appropriate values.

The following normalization, standardization, and scaling aggregators from the SciKit-Learn library were used [41]: Normalizer, StandardScaler, MinMaxScaler, MaxAbsScaler, RobustScaler, PowerTransformer QuantileTransformer, and PolynomialFeatures. The StandardScaler normalization algorithm scales each feature to a specified range. The normalization is represented as:

$$y = \frac{2(x - x_{min})}{x_{max} - x_{min}} - 1, \quad (1)$$

where x is the sample value, x_{min} and x_{max} are the minimum and maximum values of all samples, respectively, and y is the normalized value of feature.

The data were initially partitioned at random into two parts: the training dataset (70%) and the test dataset (30%). Additionally, 20% of the training dataset was used for validation.

3.3. Machine Learning Model of Fraudulent Credit Card Transactions

Detection of fraudulent credit card transactions can be based on supervised machine learning [42–44]. In supervised machine learning, the objective function f is determined using labeled training data (x, y), where x is a vector of input features and y is an output label. To correctly determine the label of a new example, the learning algorithm generalizes the correlation between the vector of features and the label in the training data.

Assume that $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ is the training dataset, where $x_i = (x_{i,1}, \ldots, x_{i,d})$ $\in X$ for $1 \leq i \leq n$, which is a vector of features for x_i and $y_i \in Y$ is its relevant label. Using the training dataset, the goal of supervised learning is to learn an objective function for a set of class labels. Assume that (x_i, y_i) are independent and equally distributed. Based on the test data, the predictive ability of the objective function can be estimated. The machine learning model is a classification model when the labels are discrete states or symbols.

One of the main machine learning goals is the search of the objective function $f : X \to Y$, where X are input data and Y is an output variable. The supervised learning or building of the model is the process of finding the objective function f. It can be detected only if there are enough tag data and all of the records are labeled in the current dataset.

The dataset, M, of the fraudulent credit card transactions model dataset consists of n = 284,807 transactions. An answer label on the dataset indicates whether this transaction is fraudulent or legitimate. The objective function f: $M \in L$ is introduced to determine whether a definite example m is "fraudulent" 1 or "legitimate" 0. Function f: $M \in 0, 1$ can be determined by one of the machine learning algorithms for a set of n labeled examples $\{(m_1; l_1), (m_2, l_2), (m_n, l_n)\}$, where $m_i \in M$ and $l_i \in 0, 1$ for $1 \leq i \leq n$.

Each transaction is associated with a feature vector, which consists of 30 values. For the objective function f it is required to determine whether transaction m is fraudulent (represented as 1) or legitimate (represented as 0). The function f can be detected using one of the machine learning algorithms for the dataset of n marked transactions.

The following machine learning algorithms are used to detect fraudulent bank transactions: linear regression, logistic regression, quadratic regression, k-nearest neighbors algorithm, support vector machine (SVM), multi-layer perceptron (MLP), discriminant function analysis, quadratic discriminant analysis (QDA), decision trees, random forest, naive Bayes classifier, and adaboost.

Our experiments with models involve testing various combinations of hyperparameters to find the optimal response. We used the GridSearchCV algorithm from the SKlearn library [41] to automate finding the best combination of hyperparameters. The GridSearchCV algorithm exhaustively generates candidates from a grid of parameter values specified. When performing hyperparameter optimization, we first need to define a region or parameter grid in which we include a set of possible values of hyperparameters that can be used to build the model. The grid search method is then used to place these hyperparameters into the matrix and the model is trained on each combination of hyperparameter values. The model with the best performance is then selected. Some algorithms can lead to overfitting, especially tree-based methods; therefore, we used regularization and early stopping to avoid overfitting.

3.4. Imbalanced Classification

If a certain class of the dataset is too small, the class is called a minority and the opposite class that is strongly represented is called the majority. There are several approaches to solve the problem of the dataset being imbalanced. One is to utilize different resampling strategies. Class balance restoration can be achieved by using one of the two methods. The first method consists in removing some transactions of the majority class (undersampling) and the second method is to use artificial data to increase some numbers of the minority (oversampling).

The oversampling approach is not as straightforward as the duplicated minority class, so samples are redistributed into multiple training batches. This approach is not practical because we are not trying to learn the spatial distribution of the database of credit card frauds with chosen features.

To solve the problem of an imbalanced dataset, the Tomek links algorithm was used [30]. All of the majority records included in Tomek links must be removed. Tomek links can be defined as follows: consider $E_i = (x_i, y_i)$ and $E_j = (x_j, y_j)$ are two class instances, where $y_i \neq y_j$, and $d(E_i, E_j)$ is the range between E_i and E_j, so pair (E_i, E_j) is called the Tomek link, if there is no case E_l, where $d(E_i, E_l) < d(E_i, E_j)$ or $d(E_j, E_l) < d(E_i, E_j)$. The Tomek links algorithm is good for removal of the records that can be considered fraudulent [45].

The Tomek links algorithm is used to clean the data followed by random undersampling as an option for data cleaning.

3.5. Model Evaluation

Some evaluation criteria have been used to estimate the performance of the models represented in this study for imbalanced classification of fraudulent bank transactions.

In this study, we used *Accuracy*, *Precision*, *Recall*, $F1$ Score, and Index of Balanced Accuracy (*IBA*) as measures to evaluate the model quality. The accuracy is a ratio of how often a classifier gives a correct prediction and is equal to

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (2)$$

where TP is the number of true positive results when the example of a positive class is correctly predicted by the model as belonging to the positive class; TN is the number of true negative results when the example of a negative class is correctly predicted as belonging to the negative class; FP is the number of false positive results, where the example of a negative class is a false prediction of the model; FN is the number of false negative results, where the example of a positive class is incorrectly predicted as belonging to the negative class.

Precision is the ratio which measures the frequency of classifier correctness, where it correctly predicts a positive class mark for the positive class.

$$Precision = \frac{TP}{TP + FP}. \quad (3)$$

Recall is the ratio which estimates the ability of a classifier to correctly predict the positive class, i.e., how often the classifier predicts a mark of the positive class dataset, when the data truly belongs to this class.

$$Recall = \frac{TP}{TP + FN}. \quad (4)$$

$F1$ Score is the harmonic average of Precision and Recall:

$$F1 = 2\,\frac{Precision\ Recall}{Precision + Recall}. \quad (5)$$

$F1$ Score's best value is 1 and worst is 0.

The geometric mean (*Gmean*) is a parameter $\sqrt{TP + TN}$ used to maximize the TP and TN classifications while maintaining the balance between them. *Gmean* minimizes the negative impact of class distribution errors, although it is unable to define the contribution of each class to the overall index, giving the same result for different combinations of TP and TN.

To estimate the binary classifier in cases of imbalanced data, the Index of Balanced Accuracy (*IBA*) was proposed [46]

$$IBA = (1 + Dominance) \cdot Gmean^2, \tag{6}$$

where *Dominance* can be defined as $TP - TN$, which is used to estimate the relationship between TP and TN. Replacing *Dominance* and *Gmean* in the final equation provides useful information to better understand how IBA supports a compromise between *Dominance* and *Gmean*.

4. Experimental Evaluation

The following steps were used for model training: data preprocessing (outlier reduction, missing value reduction, repetition reduction, and data normalization), resampling, and splitting into training and test datasets.

Before proceeding to the learning step, it is essential to check the original dataset for missing values. We initially checked all of the spaces in the tables, and verified that all values were set and that all of the records were unique. In total, 283,726 records were unique.

Initially, the dataset with 492 minority records was highly imbalanced and the percentage of these unique records was around 96%. As a result, it was decided to exclude non-unique records from the dataset and to work with the remaining 283,726 rows. The training accuracy was slightly better using this dataset. The Tomek links algorithm was used on the resulting dataset to solve the imbalanced dataset problem. Using this algorithm, 248 records were removed from the dominant class and this algorithm increased the learning accuracy percentage.

Using different combinations of machine learning algorithms from ready-made libraries is required to classify the transactions. The original dataset uses different functional transformation methods to classify the data. These methods are valuable for machine learning, since different metrics can be measured in different ranges or the values of the same metric are too strong. The functional transformation is divided into the normalization and standardization of data. Data normalization changes the data ranges without changing the data into a uniform format. Data standardization involves changes within a uniform format. Data standardization is applicable when using algorithms based on distance measurements, for example, k-nearest neighbors or support vector machine (SVM).

For the test and training samples we selected legitimate and fraudulent transactions in a ratio of 3:7. The fraudulent bank transaction model uses manually selected algorithms. To evaluate the reliability of learning models we use the following metrics: *Precision*, *Accuracy*, $F1$ Score, *Recall* and *IBA*.

The following variations of training the model for detecting fraudulent bank transactions were analyzed, including 440 different combinations in total:

1. With and without the removal of duplicate source data (two options);
2. With and without data optimization using the Tomek links algorithm (two options);
3. With and without the data normalization algorithms: MinMaxScaler, MaxAbsScaler, StandardScaler, PowerTransformer, QuantileTransfomer, Normalizer, FunctionTransformer, PolynomialFeatures, and RobustScaler (10 options);
4. With different machine learning algorithms: RandomForest, AdaBoost, K-Neighbors, DecisionTree, LogisticRegression, SVC, CatBoost, XGBoost, LGBM, TPOT, and AutoSklearn (11 options).

Based on the RandomForest machine learning method, it was concluded that using the Tomek links algorithm increases the built model accuracy by 0.0001; therefore, this data optimization was used for the each successive method. Deleting the duplicated initial data did not affect the model accuracy.

Figure 2 contains the results of the RandomForest machine learning method with various data normalizers. The best scaling functions are Normalizer, PowerTransformer, StandardScaler, and QuantileTransformer. Whitening, as a data preprocessing step, is used to remove correlation or dependencies between features in a dataset. These normalizers demonstrated the same accuracy; however, due to *IBA*, the best scaling method is the QuantileTransformer. This confirmed the theoretical assumptions that this scaler was developed for classification problems.

Normalization Algorithm	*Accuracy*	*IBA*
MinMaxScaler	0.9998	0.9012
MaxAbsScaler	0.9998	0.9111
StandardScaler	0.9999	0.9225
PowerTransformer	0.9999	0.9256
QuantileTransformer	0.9999	0.9306
Normalizer	0.9999	0.9199
RobustScaler	0.9998	0.9056
FunctionTransformer	0.9998	0.9168
PolynomialFeatures	0.9998	0.9044
Whitening	0.9998	0.9296

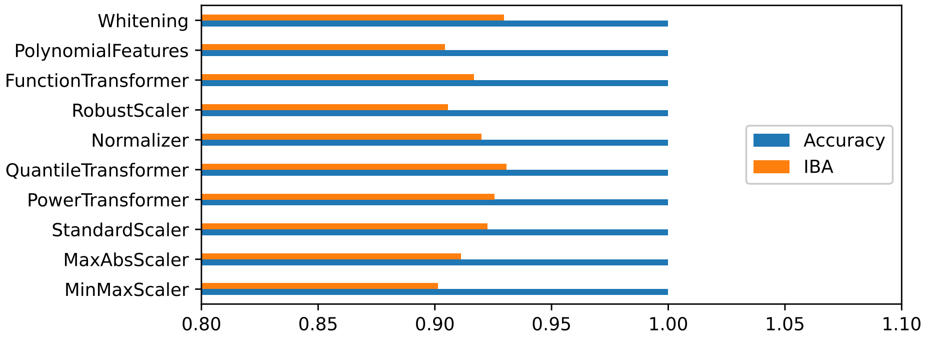

Figure 2. The data normalization reliability on a test dataset using metrics accuracy and *IBA*.

Additionally, the TPOT algorithm was used for the ready-made library of automated machine learning which optimizes the machine learning capabilities using genetic programming [47], and the AutoSklearn was used for the automated machine learning with the enumeration of machine learning algorithms and their hyperparameters [48].

Figure 3 shows the best results for the each machine learning method. Metrics *Recall*, *F*1 Score, and *Precision* are calculated separately for each transaction. Figure 3 shows the best results specifically for fraudulent bank transactions, since it is important to perform an error correction of the fraudulent transaction class to fully understand the effectiveness of detecting such transactions. Based on the results from Figure 3 we concluded that the best accuracy rates were achieved by CatBoost, AutoSklearn, XGBoost, and RandomForest.

Method	*Precision*	*Recall*	*F1* Score	*Accuracy*	*IBA*
RandomForestClassifier	0.9866	0.9366	0.9610	0.9999	0.9306
AdaBoostClassifier	0.8113	0.7273	0.7670	0.9993	0.7073
K-NeighborsClassifier	0.6881	0.6773	0.6993	0.9985	0.6885
DecisionTreeClassifier	0.9129	0.9302	0.9215	0.9997	0.9236
LogisticRegression	0.7047	0.6913	0.6980	0.9990	0.6697
SVM	0.6425	0.6772	0.6328	0.9983	0.6567
CatBoostClassifier	0.9799	0.9281	0.9533	0.9998	0.9214
XGBoostClassifier	0.9820	0.9239	0.9521	0.9998	0.9168
LGBMClassifier	0.5313	0.8161	0.5634	0.9979	0.7997
TpotClassifier	0.9888	0.9366	0.9620	0.9999	0.9306
AutoSklearnClassifier	0.9930	0.9006	0.9446	0.9998	0.8917

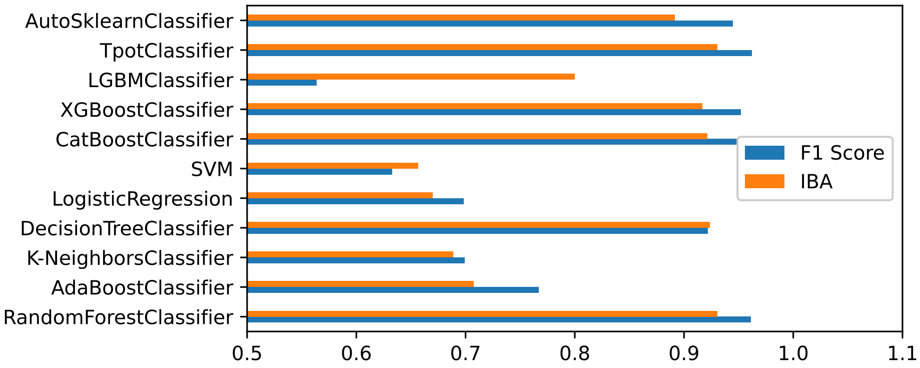

Figure 3. The results of classification method reliability on the test dataset; *Precision*, *Recall*, and *F*1 Score are specified for transaction class "fraudulent".

Since TPOT and AutoSklearn are automated and choose the best hyperparameters and training methods, Figure 3 lists the methods and their variations with the best results. For TPOT, it was XGBClassifier with the following parameters: learning_rate = 0.5, max_depth = 7, min_child_weight = 1, n_estimators = 100, n_jobs = 1, subsample = 0.8; for AutoSklearn it was the RandomForest method with the following parameters: learning_rate = 0.5, max_depth = 10, min_child_weight = 1, n_estimators = 100, n_thread = 1, subsample = 0.7.

Since the parameters are approximately the same, the models were trained using different methods with the same hyperparameters, instead of using brute force in order to get these hyperparameters.

Figure 4 lists the best results achieved in this study and a comparison with the other studies [18–28], which also built models for detecting fraudulent bank transactions on the same dataset. For the fraudulent bank transactions the metrics are *Precision*, *Recall*, *F*1 Score, and *IBA*. If their values add up, then they get listed as a dash. The results obtained in this study were more accurate than the ones previously obtained.

The proposed model based on the TPOT and RandomForest algorithms provided the best results with a Recall of 0.9366 on the testing dataset. Recall demonstrates the ability of the algorithm to detect the class at all while the models [18–28] provide poor results. However, the accuracy of the models is high here, because the CreditCardFraud dataset is highly imbalanced. Thus, we can conclude that the model based on the TPOT and RandomForest algorithms performed better than the other models [18–28]. The created models are open and accessible to the research community [49].

Algorithm	***Precision***	***Recall***	***F*1 Score**	***Accuracy***	***IBA***
RandomForestClassifier	0.9866	**0.9366**	**0.9610**	**0.9999**	**0.9306**
CatBoostClassifier	0.9799	0.9281	0.9533	0.9998	0.9214
XGBoostClassifier	0.9820	0.9239	0.9521	0.9998	0.9168
TpotClassifier	**0.9888**	**0.9366**	**0.9620**	**0.9999**	**0.9306**
AutoSklearnClassifier	0.9930	0.9006	0.9446	0.9998	0.8917
AE + PRF (0.03) (Lin, 2021)	0.99757	0.89109	-	0.99738	-
AE + PRF (0.25) (Lin, 2021)	0.9998	0.8142	-	0.9995	-
AED-LGB (0.2) (Du, 2023)	0.9997	0.8039	-	0.9993	-
AED-LGB (0.05) (Du, 2023)	0.9932	0.8929	-	0.9987	-
KNN (Du, 2023)	0.9711	0.8835	-	0.9691	-
LightGBM (Du, 2023)	0.9995	0.8529	-	0.9696	-
Random Forest (Du, 2023)	0.9989	0.8025	-	0.9583	-
k-NN (Awoyemi, 2017)	0.9711	0.8835	-	0.9691	-
AE (Pumsirirat, 2018)	0.97077	0.83673	-	0.97054	-
AE-based clustering (Zamini, 2018)	0.98932	0.81632	-	0.98902	-
SVM with AdaBoost (Randhawa, 2018)	0.99957	0.82317	-	0.99927	-
NN+NB with MV (Randhawa, 2018)	0.99978	0.78862	-	0.99941	-
LogisticRegression (Kaggle,joparga3)	0.9900	0.9000	0.9400	0.9400	-
K-Neighbors (Kaggle,joparga3)	0.9900	0.8600	0.9200	0.9300	-
SVM (Kaggle,joparga3)	0.9900	0.8800	0.9300	0.9300	-
LogisticRegression (Kaggle,shivamb)	0.9226	0.9184	-	0.9315	-
Autoencoders (Habr,478286)	0.8400	0.7400	0.7900	0.8341	-
IsolationForest (Habr,477450)	-	-	-	0.8049	-
Local Outlier factor (Dornadula,2019)	0.0038	-	-	0.8990	-
Isolation forest (Dornadula,2019)	0.0147	-	-	0.9011	-
SVM (Dornadula,2019)	0.7681	-	-	0.9987	-
Logistic regression (Dornadula,2019)	0.8750	-	-	0.9990	-
Decision tree (Dornadula,2019)	0.8854	-	-	0.9994	-
Random forest (Dornadula, 2019)	0.9310	-	-	0.9994	-
Local Outlier factor (Dornadula,2019)	0.2941	-	-	0.4582	-
Isolation forest (Dornadula,2019)	0.9447	-	-	0.5883	-
Logistic regression (Dornadula,2019)	0.9831	-	-	0.9718	-
Decision tree (Dornadula,2019)	0.9814	-	-	0.9708	-
Random forest (Dornadula,2019)	0.9996	-	-	0.9998	-

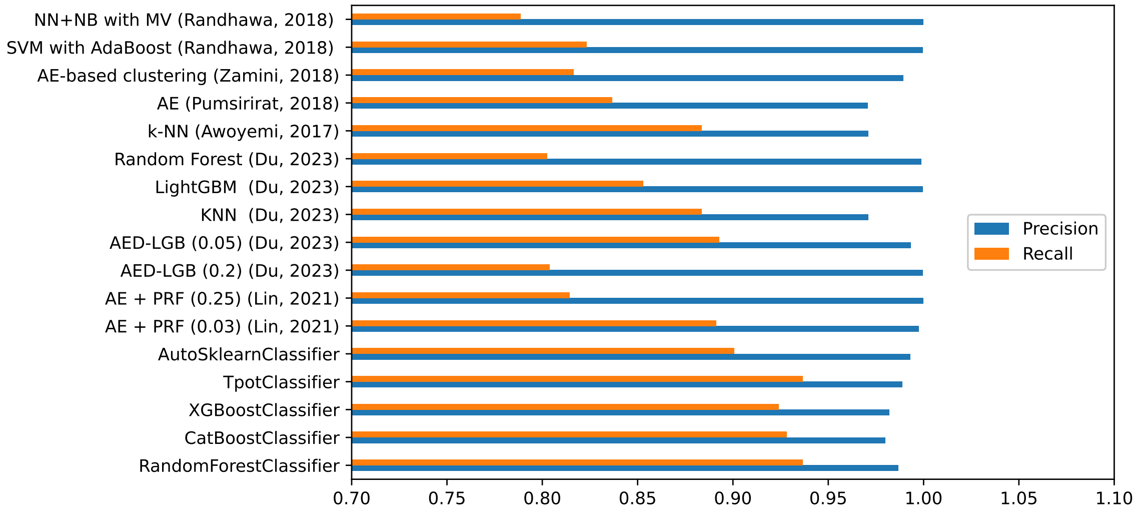

Figure 4. The results of classification methods' reliability [18–28] on a test dataset using metrics *Accuracy*, *Precision*, *Recall*, *F*1 Score, and *IBA*. Metrics' precision, recall, and *F*1 Score are specified for transaction class "fraudulent". We mark the best results in bold.

IBA and *Precision* are verified with 10-fold cross-validation. For the test and training samples we have selected legitimate and fraudulent transactions in a ratio of 3:7. The result of this analysis is shown in Figure 5 for the following top models: CatBoostClassifier, TPOT, RandomForestClassifier, XGBClassifier, and LGBMClassifier. The metric values remain almost the same for all ten iterations. Consequently, the TPOT and RandomForest models are stable when fraudulent bank transactions are detected.

Algorithm	**Accuracy**		**IBA**	
	Mean	**SD**	**Mean**	**SD**
RandomForestClassifier	0.9999	0.7864	0.9306	6.1886
LGBMClassifier	0.9996	0.5099	0.9199	6.6320
XGBClassifier	0.9994	0.1377	0.9168	6.6133
CatBoostClassifier	0.9993	0.7603	0.9214	6.6755
TPOT	0.9999	0.7603	0.9306	6.6755

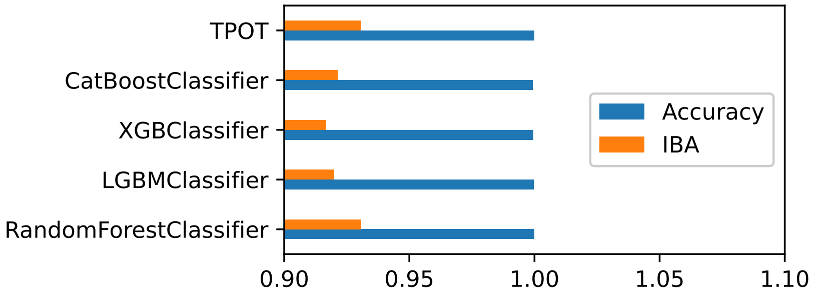

Figure 5. The 10-fold cross-validation of the Accuracy score and the IBA metric for RandomForestClassifier, LGBMClassifier, XGBClassifier, CatBoostClassifier, and TPOT. Mean: average of various metrics; SD ($\times 10^{-4}$): standard deviation of various metrics.

5. Discussion

When credit card fraud is detected, the costs of incorrectly identifying the class label are limited. The costs associated with overdetection of fraudulent transactions must be issued as quickly as possible after these transactions occur. In addition, there are costs associated with inaccurately identifying a legitimate transaction as fraudulent, but these tend to be lower and easier to manage. If a transaction is classified as fraudulent, a finance company may freeze the account and contact the owner. If the transaction proves to be legitimate, the finance company can unfreeze the account and the owner can generally proceed with using the account as if nothing happened. A finance company hiring someone to research and develop strategies to detect fraudsters can have a deterrent effect and even prevent certain cases of fraud.

Imbalanced data, without any adjustments, have a dramatic effect on the performance of classification algorithms. Several methods have been proposed for adjusting the standard classification process when there are imbalanced data [45]. Many experiments were conducted with different methods of adjustment; however, conclusions were inconclusive. Some researchers present the idea of the superiority of sampling techniques while others recommend cost-sensitive and ensemble methods as optimal solutions [45].

The benefits of sampling techniques include simplicity and portability; however, they have some limitations such as information loss, class overlapping, overfitting and a prediction bias towards the minority class. In addition, it is difficult to differentiate between the minority class and noise. The Tomek links algorithm can solve this problem by removing noise; however, the problem of imbalanced class distribution is still present. In this article, we recommend the use of the Tomek links algorithm as a method of data cleaning followed by random undersampling as an option for data cleaning. This improves the performance of the classification algorithm because removing noise from the majority class, followed by random sampling, reduces the chance of information loss. Our conclusion supports the finding of Elhassan et al., where the authors expressed this idea using the same approach [45].

6. Conclusions

We proposed a model for detecting fraudulent bank transactions based on machine learning. In the training we used a dataset consisting of 284,807 credit card transactions, of which 492 were classified as fraudulent and 284,315 as legitimate. The model was assessed based on *Accuracy*, *F*1 score, *Recall*, *Precision*, and *IBA*. The model for detecting of fraudulent bank transactions uses different machine learning algorithms with hyperparameters.

The majority class usually dominates in cases of imbalanced classification, meaning examples of the minority class can be misclassified. The data presented in this study were severely imbalanced. Our main objective was to demonstrate that an improvement in the classification performance of the classification algorithms towards prediction of a positive rare class using a resampling algorithm can be achieved. The undersampling algorithm demonstrated superior performance.

The majority class usually dominates in cases of imbalanced classification, so examples of the minority class are misclassified. The data presented in this study were severely imbalanced. Our main objective was to demonstrate that an improvement in the classification performance of the classification algorithms towards prediction of a positive rare class using a resampling algorithm can be achieved. The undersampling algorithm demonstrated superior performance.

Using the Tomek links resampling algorithm, the classification reliability of fraudulent bank transactions was increased in comparison to the best result of the imbalanced CreditCardFraud dataset. This result bettered those in [18–28]. As a result the accuracy of the model is 0.9999 based on the TPOT and the RandomForest algorithms, which was confirmed using 10-fold cross-validation.

Author Contributions: Conceptualization, A.R. and E.F.; methodology, A.R.; software, D.C.; validation, E.F., A.R. and D.C.; formal analysis, A.S.; investigation, A.R.; resources, A.R.; data curation, D.C.; writing—original draft preparation, A.R.; writing—review and editing, A.R.; visualization, A.R.; supervision, A.R.; project administration, A.R.; funding acquisition, A.R. All authors have read and agreed to the published version of the manuscript.

Funding: The research was funded by the Russian Science Foundation (project No. 22-71-10095).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and source code presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khrestina, M.P.; Dorofeev, D.I.; Kachurina, P.A.; Usubaliev, T.R.; Dobrotvorskiy, A.S. Development of Algorithms for Searching, Analyzing and Detecting Fraudulent Activities in the Financial Sphere. *Eur. Res. Stud. J.* **2017**, *20*, 484–498.
2. Alsuwailem, A.; Saudagar, A. Anti-money laundering systems: A systematic literature review. *J. Money Laund. Control.* **2020**, *23*, 833–848. [CrossRef]
3. Stojanović, B.; Božić, J. Robust Financial Fraud Alerting System Based in the Cloud Environment. *Sensors* **2022**, *22*, 9461. [CrossRef]
4. Srokosz, M.; Bobyk, A.; Ksiezopolski, B.; Wydra, M. Machine-Learning-Based Scoring System for Antifraud CISIRTs in Banking Environment. *Electronics* **2023**, *12*, 251. [CrossRef]
5. Razaque, A.; Frej, M.B.H.; Bektemyssova, G.; Amsaad, F.; Almiani, M.; Alotaibi, A.; Jhanjhi, N.Z.; Amanzholova, S.; Alshammari, M. Credit Card-Not-Present Fraud Detection and Prevention Using Big Data Analytics Algorithms. *Appl. Sci.* **2023**, *13*, 57. [CrossRef]
6. Bakumenko, A.; Elragal, A. Detecting Anomalies in Financial Data Using Machine Learning Algorithms. *Systems* **2022**, *10*, 130. [CrossRef]
7. Jullum, M.; Løl, A.; Huseby, R.B.; Ånonsen, G.; Lorentzen, J. Detecting money laundering transactions with machine learning. *J. Money Laund. Control.* **2020**, *23*, 173–186. [CrossRef]

8. Weber, M.; Chen, J.; Suzumura, T.; Pareja, A.; Ma, T.; Kanezashi, H.; Kaler, T.; Leiserso, C.E.; Schardl, T.B. Scalable graph learning for anti-money laundering: A first look. *arXiv* **2018**, arXiv:1812.00076.
9. Singh, K.; Best, P. Anti-money laundering: Using data visualization to identify suspicious activity. *Int. J. Account. Inf. Syst.* **2019**, *34*, 100418. [CrossRef]
10. Weber, M.; Domeniconi, G.; Chen, J.; Weidele, D.; Bellei, C.; Robinson, T.; Leiserson, C. Anti-Money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics. *arXiv* **2019**, arXiv:1908.02591.
11. Feldman, E.V.; Ruchay, A.N.; Matveeva, V.K.; Samsonova, V.D. Bitcoin abnormal transaction detection model based on machine learning. *Chelyabinsk Phys. Math. J.* **2021**, *6*, 119–132. [CrossRef]
12. Feldman, E.V.; Ruchay, A.N.; Matveeva, V.K.; Samsonova, V.D. Bitcoin Abnormal Transaction Detection Based on Machine Learning. Recent Trends in Analysis of Images, Social Networks and Texts (AIST 2020). *Commun. Comput. Inf. Sci.* **2021**, *1357*, 205–215.
13. Deng, W.; Huang, T.; Wang, H. A Review of the Key Technology in a Blockchain Building Decentralized Trust Platform. *Mathematics* **2023**, *11*, 101. [CrossRef]
14. Lucas, Y. *Credit Card Fraud Detection Using Machine Learning with Integration of Contextual Knowledge*; Artificial Intelligence; Universite de Lyon: Lyon, France; Universitat Passau: Passau, Germany, 2019.
15. Maniraj, S.P.; Aditya, S.; Shadab, A.; Swarna, S. Credit Card Fraud Detection using Machine Learning and Data Science. *Int. J. Eng. Res. Technol.* **2019**, *8*. [CrossRef]
16. Lebichot, B.; Le Borgne, Y.A.; He-Guelton, L.; Oble, F.; Bontempi, G. Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection. In *Recent Advances in Big Data and Deep Learning: Proceedings of the International Neural Networks Society (INNSBDDL 2019)*; Springer: Berlin/Heidelberg, Germany, 2020.
17. Carcillo, F.; Borgne, Y.L.; Caelen, O.; Kessaci, Y.; Oble, F.; Bontempi, G. Combining unsupervised and supervised learning in credit card fraud detection. *Inf. Sci.* **2019**, *557*, 317–331. [CrossRef]
18. Dornadula V.N.; Geetha S. Credit Card Fraud Detection using Machine Learning Algorithms. *Procedia Comput. Sci.* **2019**, *165*, 631–641. [CrossRef]
19. In Depth Skewed Data Classif. Available online: https://www.kaggle.com/joparga3/in-depth-skewed-data-classif-93-recall-acc-now (accessed on 1 January 2023).
20. Semi Supervised Classification Using AutoEncoders. Available online: https://www.kaggle.com/shivamb/semi-supervised-classification-using-autoencoders (accessed on 1 January 2023).
21. Fraud Detection with Random Forest, Neural Autoencoder, and Isolation Forest Algorithms. Available online: https://habr.com/company/nix/blog/478286/ (accessed on 1 January 2023).
22. 9 Approaches for Detecting Anomalies. Available online: https://habr.com/post/477450/ (accessed on 1 January 2023).
23. Lin, T.-H.; Jiang, J.-R. Credit Card Fraud Detection with Autoencoder and Probabilistic Random Forest. *Mathematics* **2021**, *9*, 2683. [CrossRef]
24. Du, H.; Lv, L.; Guo, A.; Wang, H. AutoEncoder and LightGBM for Credit Card Fraud Detection Problems. *Symmetry* **2023**, *15*, 870. [CrossRef]
25. Awoyemi, J.O.; Adetunmbi, A.O.; Oluwadare, S.A. Credit card fraud detection using machine learning techniques: A comparative analysis. In Proceedings of the 2017 International Conference on Computing Networking and Informatics (ICCNI), Lagos, Nigeria, 29–31 October 2017; pp. 1–9.
26. Pumsirirat, A.; Yan, L. Credit Card Fraud Detection using Deep Learning based on Auto-Encoder and Restricted Boltzmann Machine. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 18–25. [CrossRef]
27. Zamini, M.; Montazer, G. Credit Card Fraud Detection using autoencoder based clustering. In Proceedings of the 2018 9th International Symposium on Telecommunications (IST), Tehran, Iran, 17–19 December 2018; pp. 486–491.
28. Randhawa, K.; Loo, C.K.; Seera, M.; Lim, C.P.; Nandi, A.K. Credit Card Fraud Detection Using AdaBoost and Majority Voting. *IEEE Access* **2018**, *6*, 14277–14284. [CrossRef]
29. CreditCardFraud. Available online: https://www.kaggle.com/mlg-ulb/CreditCardFraudfraud (accessed on 1 January 2023).
30. Tomek I. Two modifications of CNN. *IEEE Trans. Syst. Man Cybern.* **1976**, *6*, 769–772.
31. HSN Consultants, Inc. *Card Fraud Losses Reach 22.80 Billion*; Technical Report 1118; The Nilson Report: Oxnard, CA, USA, 2017.
32. Plaksiy, K.; Nikiforov, A.; Miloslavskaya, N. Applying Big Data Technologies to Detect Cases of Money Laundering and Counter Financing of Terrorism. In Proceedings of the 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), Barcelona, Spain, 6–8 August 2018; pp. 70–77.
33. Zong, W.; Zhou, F.; Pavlovski, M.; Qian, W. Peripheral Instance Augmentation for End-to-End Anomaly Detection Using Weighted Adversarial Learning. In *Database Systems for Advanced Applications. DASFAA 2022*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2022; Volume 13246.
34. Pang, G.; Shen, C.; Hengel, A. Deep Anomaly Detection with Deviation Networks. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19), Anchorage, AK, USA, 4–8 August 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 353–362.
35. Huang, Z.; Zhang, B.; Hu, G.; Li, L.; Xu, Y.; Jin, Y. Enhancing unsupervised anomaly detection with score-guided network. *arXiv* **2021**, arXiv:2109.04684.

36. Kumar, N.; Shaju, S.J.; Kayathwal, K.; Agarwal, K.; Singh, A.; Chaurasia, D.; Asthana, S.; Arora, A. Intent2vec: Representation learning of cardholder and merchant intent from temporal interaction sequences for fraud detection. In Proceedings of the IJCAI-21 Workshop on Applied Semantics Extraction and Analytics (ASEA), Virtual, 21–23 August 2021.
37. Zhou, Y.; Song, X.; Zhang, Y.; Liu, F.; Zhu, C.; Liu, L. Feature Encoding With Autoencoders for Weakly Supervised Anomaly Detection. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 2454–2465. [CrossRef] [PubMed]
38. Carneiro, E.M.; Forster, C.H.Q.; Mialaret, L.F.S.; Dias, L.A.V.; da Cunha, A.M. High-Cardinality Categorical Attributes and Credit Card Fraud Detection. *Mathematics* **2022**, *10*, 3808. [CrossRef]
39. Alwadain, A.; Ali, R.F.; Muneer, A. Estimating Financial Fraud through Transaction-Level Features and Machine Learning. *Mathematics* **2023**, *11*, 1184. [CrossRef]
40. Dalal, S.; Seth, B.; Radulescu, M.; Secara, C.; Tolea, C. Predicting Fraud in Financial Payment Services through Optimized Hyper-Parameter-Tuned XGBoost Model. *Mathematics* **2022**, *10*, 4679. [CrossRef]
41. Pedregosa F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Louppe, G.; Prettenhofer, P.; Weiss, R.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
42. Thomas, T.; Vijayaraghavan, A.P.; Sabu, E. *Machine Learning Approaches in Cyber Security Analytics*; Springer: Singapore, 2020.
43. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006; 738p.
44. MacKay, D. *Information Theory, Inference, and Learning Algorithms*; Cambridge University Press: Cambridge, UK, 2003.
45. Elhassan, A.T.; Aljourf, M.; Al-Mohanna, F.; Shoukri, M. Classification of Imbalance Data using Tomek Link (T-Link) Combined with Random Under-sampling (RUS) as a Data Reduction Method. *Glob. J. Technol. Optim.* **2016**, *1*, 1–11.
46. Garcia, V.; Mollineda, R.A.; Sanchez, J.S. Index of Balanced Accuracy: A Performance Measure for Skewed Class Distributions. In *Pattern Recognition and Image Analysis. IbPRIA 2009*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5524.
47. Olso, R.S.; Bartley, N.; Urbanowicz, R.J.; Moore, J.H. Evaluation of a tree-based pipeline optimization tool for automating data science. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), Denver, CO, USA, 20–24 July 2016; ACM: New York, NY, USA, 2016; pp. 485–492.
48. Feurer, M.; Klein, A.; Eggensperger, K.; Springenberg, J.T.; Blum, M.; Hutter, F. *Auto-Sklearn: Efficient and Robust Automated Machine Learning*; Springer International Publishing: New York, NY, USA, 2019; pp. 113–134.
49. Ruchay, A. The Classification of Fraudulent Bank Transactions. Available online: https://github.com/ruchaya/CreditCardFraud (accessed on 1 January 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.