Article

Credit Card Fraud Detection Using a Hybrid Machine Learning Algorithm

Shubham Singh *, Harshita Nimje , Ajinkya Fulpatile , Rahul Neware

Posted Date: 22 February 2024

doi: 10.20944/preprints202402.1206.v1

Keywords: Machine Learning; Fraud Detection; Deep Learning; Credit Card Fraud; CN; KNN

Article

Credit Card Fraud Detection Using a Hybrid Machine Learning Algorithm

Shubham Singh [1,*], Harshita Nimje [1], Ajinkya Fulpatile [1] and Rahul Neware [2]

[1] School of Computer Science, Vellore Institute of Technology, Bhopal-466114, India
[2] ICAR-Indian Agricultural Statistical Research Institute, New Delhi India
* Correspondence: shubham.singh020403@gmail.com

Abstract: The use of credit cards and online banking is expanding exponentially. As more individuals use debit cards, credit cards, and internet banking, the likelihood of falling victim to various forms of fraud also rises. Due to ignorance, credit card company customers have provided their card details, personal information, and one-time password to an unknown bogus caller on several occasions in the recent past. One of the biggest issues facing financial institutions and customers alike is credit card theft. It may result in large financial losses and harm to the financial institution's image. While there are several ways to identify credit card fraud, one of the best is machine learning.By using past data, machine learning algorithms may be trained to recognize trends and abnormalities that point to fraud. The state-of-the-art in machine learning for credit card fraud detection is reviewed in this study. Numerous research on the same subject have been conducted in the past with a variety of well-known machine-learning techniques. However, with a grasp of the notion of hybrid machine learning, we will now examine how well hybrid machine learning performs on the same problem statement.

Keywords: machine learning; fraud detection; deep learning; credit card fraud; CNN; KNN

Introduction

There have always been people searching for new ways to gain illegal access to another person's financial information since the beginning of the digital process. This has become a major worry in the modern day because all transactions can now be easily made online by simply entering the credit card information. One of the biggest issues facing financial institutions and customers alike is credit card theft. It may result in large financial losses and harm to the financial institution's image. Losses from credit card fraud amount to billions of dollars annually, making it an increasingly serious issue. Financial organizations find it more and more challenging to identify and stop fraud as fraudsters are always coming up with new ways to steal credit card information and conduct fraud.

Keeping with the details, The Nilson Report [nila] states that the global losses from credit card, debit card, and prepaid card fraud in 2015 were $16.31 billion. According to a new analysis by The Nilson analysis [nila], the gross fraud loss in 2018 was $22.8 billion, 4% more than in 2015, and is predicted to rise by even more in the upcoming years. In 2012, the gross fraud amount was $5.6 billion, according to Statista [sta]. In 2018, the fraud loss amounted to $9.1 billion, or around two-fifths of the entire loss. Specifically, Card-Not-Present (CNP) frauds, or those committed online or over the phone, account for 70% of these incidents; counterfeits account for 20%, and losses resulting from misplaced cards account for the remaining 10%.

 doi:10.20944/preprints202402.1206.v1

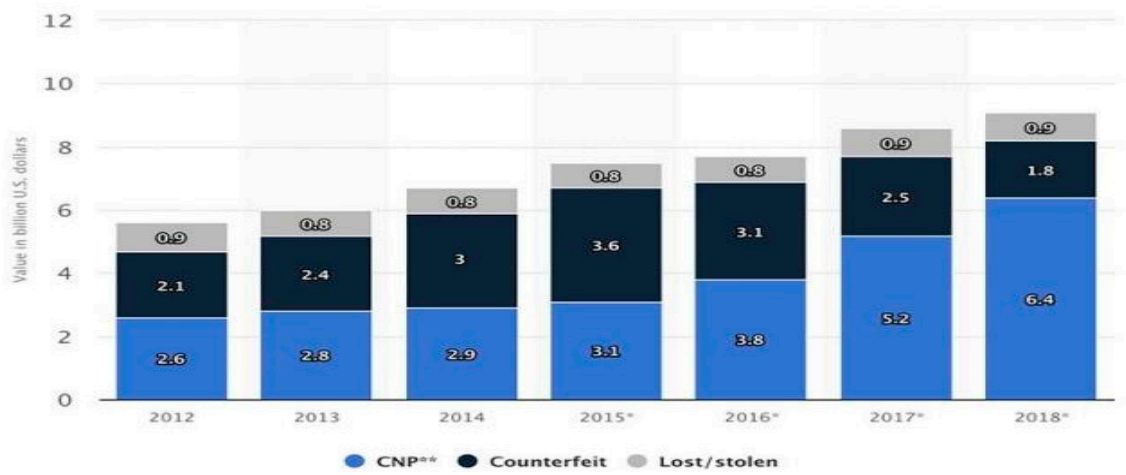

Figure 1. Graph showing the number of cases of credit card fraud.

There has been a rise in credit card fraud and scams in tandem with rising credit card usage. According to a National Crime Records Bureau (NCRB) study, 3,432 instances of credit and debit card fraud were reported in India in 2021—a almost 20% rise from 2020.

One of the most prevalent types of fraud in the United States in 2021 was credit card fraud, as reported by the Federal Trade Commission (FTC) in close to 390,000 cases. However, that number only scratches the surface of the issue.

The payments industry monitor Nilson Report published a prediction in December 2022 that states that over the following ten years, card fraud losses in the United States would hit $165.1 billion, affecting all age groups and all states. According to Insider Intelligence, card-not-present fraud, which includes online, over-the-phone, and mail-order transactions, is the only kind of credit card fraud that is expected to cause $5.72 billion in losses for the United States in 2022.

A team of data scientists will investigate the data and construct models as part of the Credit Card Fraud Detection using Machine Learning technique, which will produce the greatest results in preventing revealing and fraudulent transactions. This procedure may be compared to a cross between artificial intelligence and conventional data analysis. To do this, all of the pertinent details from card users' transactions—such as Client Behavioral Patterns, Provider, Amount, Product Category, User Zone, Date, and so forth—are combined. Transactions that are almost identical to the original or that copy a transaction have both been accomplished using clone transactions. This might happen if a corporation repeatedly sends the same invoice to different departments inside a partner company in an effort to collect payment from that partner. The optimum option is one where the system can distinguish between a transaction that was made accidentally and one that was fraudulent. We now need to choose the approaches we will employ for the forecasts after identifying all the significant factors. Certain parameters in these aspects may be of the categorical type [now, when you have categorical data, we may apply the following methods: logistic regression, CNN model, SVM, Decision Tree, linear regression, Random Forest technique, gradient boosting]. Other parameters may consist of numerical data [numerical data is now suitable for CNN, SVM, Random Forest, Linear Regression, Logistic Regression, Decision Trees, and Gradient Boosting models]. In this case, clone transactions created by human error will be distinguished from actual fraudulent activity more successfully by utilizing certain machine learning techniques that can operate with both categorical and numerical data. These techniques will be selected to create hybrid machine learning models. In order to create a hybrid model that outperforms any single model, we will first grasp the principles of hybrid machine learning and its benefits in this work.

Literature review

Credit card fraud detection is a critical domain in financial security, aiming to protect users and merchants from unauthorized transactions. The exploration of various methods to enhance the accuracy and efficiency of fraud detection systems has been a focus of numerous studies.

 doi:10.20944/preprints202402.1206.v1

In a study conducted by Katiyara et al. [1], the authors focused on deep learning advancements in credit card fraud detection. Their comparative study, utilizing a Kaggle dataset, showcased the superiority of CNN and LSTM. According to the stated accuracy values for these models, the ANN achieved an accuracy of 94.76%, the CNN achieved an accuracy of 95.89%, the autoencoder achieved an accuracy of 32.15%, the LSTM achieved an accuracy of 99.92%, and the gradient boosting model earned an accuracy of 94.00%, underscoring the efficacy of deep learning in enhancing credit card transaction security.

Susie et al. [2] introduced the xFraud framework for explainable fraud transaction detection in online retail platforms. xFraud detector+ AUC is 0.9074 (with 8 machines) and 0.8892 (with 16 machines) and GEM AUC is 0.8961 (with 8 machines) and 0.8938 (with 16 machines). The framework exhibited exceptional accuracy and precision in fraud detection, emphasizing its robustness and efficiency in real-world scenarios.

In their 2007 study [3], compared three classification methods decision tree, neural networks, and logistic regression in the context of credit card fraud detection. Utilizing real-time transaction data, the study concluded that neural networks and logistic regression outperformed the decision tree method, highlighting the potential for an accurate and user-friendly credit card risk monitoring system.

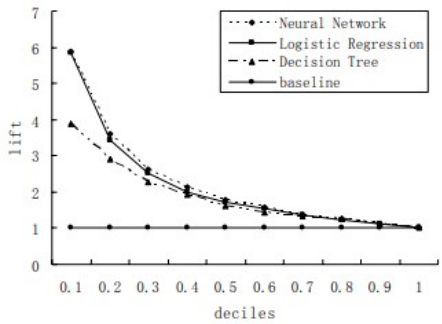

Figure 2. Different Machine Learning Model Curves on the scale of Deciles vs lift.

The study presented at the International Conference on Machine Learning in 2018, authored by Biao et al. [4], evaluates various fraud detection methods. These methods encompass Random Forest, Naive Bayes, MLP, AdaBoost, XGBoost, LightGBM, SMOTBoost, RUSBoost, Self-Paced Ensemble, and the proposed Deep Boosting Decision Trees (DBDT) - with variations DBDT-SGD and DBDT-Com. While specific accuracy values are not disclosed, the paper concentrates on assessing the performance of these methods in fraud detection, shedding light on the effectiveness of diverse approaches in identifying fraudulent activities.

Dornadula and Sa [5] presented a novel method using clustering and sliding windows to analyze past transactions and extract behavioral patterns for fraud detection. The study addressed imbalanced datasets using SMOTE, revealing the impact of the technique on improving model performance. Experimental results show Random Forest's high accuracy (0.9994) on the original dataset, while Logistic Regression and Decision Tree excel post-SMOTE, emphasizing the technique's impact on improving model performance for imbalanced datasets.

In the 2019 study on anomaly detection by Pang et al [6], Deviation Networks (DevNet) are introduced alongside competing methods for evaluation. DevNet demonstrates superior performance, ranking highest on eight and nine datasets for AUC-ROC and AUC-PR, respectively, and second on the census dataset for AUC-ROC. Notably, DevNet achieves significant average improvements compared to other methods, with AUC-ROC enhancements ranging from 3% to 29%, and AUC-PR improvements ranging from 21% to 309%. These findings underscore DevNet's efficiency in optimizing anomaly scores, resulting in enhanced precision and recall, contrasting with competing methods' weaker capabilities due to indirect learning approaches.

Maniraj et al. [7], investigated fraud detection in credit card transactions. Employing the Local Outlier Factor and Isolation Forest Algorithm, the study achieved remarkable accuracy, exceeding

 doi:10.20944/preprints202402.1206.v1

99.6% with a tenth of the dataset. However, precision slightly declined to 33% when analyzing the entire dataset. This research highlights the efficacy of machine learning and data science techniques in detecting fraudulent activities, contributing significantly to enhancing security in electronic payments.

Recent research has turned to Hidden Markov Models (HMMs), introducing novel features to categorize transactions across dimensions. This study, conducted by Lucas et al. [8], demonstrated that the highest PR-AUC reported in the paper is 0.98 for the Random Forest classifier with the proposed feature set.

Kayode Olaleye Ayorinde [9], explored credit card fraud detection through machine learning models like Logistic Regression, Decision Tree, Random Forest, XGBoost, K-Means Clustering, and Autoencoder in Keras. The study employs a dataset of 284,807 transactions from European cardholders, aiming to propose a robust methodology for credit card fraud detection. Notably, accuracy levels vary among models, with k-nearest neighbor achieving 97.69%, Random Forest at 99.21%, and ANN reaching 99.92%. The research contributes to the evolving landscape of fraud prevention in credit card transactions.

Mekterović [10], listed and examined various credit card fraud detection methods. They conducted a preliminary experiment assessing models like logistic regression, multilayer perceptron, and random forest, measuring accuracy through metrics such as recall, precision, F1 score, and average precision. Results highlighted the superiority of newly developed models, notably the random forest, which consistently outperformed existing ones across multiple metrics.

Table 1. Conclusion made by Igor Mekterović, Mladen Karan, Damir Pintar, and Ljiljana Brkić, Ileberi et al. [11], explained various machine learning classifiers including Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), Artificial Neural Network (ANN), and Naive Bayes (NB) were utilized for fraud detection. The study incorporated the Genetic Algorithm (GA) for feature selection. These classifiers achieved accuracy scores ranging from 95.50% to 99.98%, showcasing their effectiveness in detecting credit card fraud.

Classifier	Recall	Precision	F1	AP
Baseline (RE)	0.564	0.378	0.453	0.189
LR.5.A.sm	0.454	0.766	0.570	0.388
LR.5.B.sm	0.454	0.763	0.569	0.387
LR.5.C.sm	0.451	0.759	0.566	0.385
MLP.5.A.sm	0.454	0.786	0.576	0.393
MLP.5.B.sm	0.441	0.707	0.543	0.354
MLP.5.C.sm	0.464	0.739	0.570	0.391
RF.5.A.sm	0.517	0.932	0.665	0.512
RF.5.B.sm	0.517	0.932	0.665	0.512
RF.5.C.sm	0.519	0.934	0.667	0.515

Xiang et al. [12] present a semi-supervised graph neural network approach to address limitations in existing fraud detection techniques. The GTAN approach outperforms other methods with an accuracy of 98.5%. The paper uses credit card transaction records and two publicly supervised fraud detection datasets, YelpChi and Amazon, to evaluate and compare the approach's effectiveness.

Nguyen et al. [13] introduced a pioneering approach termed posterior detection with future information. The study emphasized the importance of incorporating "future" information from subsequent transactions, showcasing improved fraud detection accuracy with the Bi-LSTM model with a past-present-future sequence length of 4-1-2 achieved an AUCPR of 0.367 ± 0.008 at the transaction level, while the Random Forest baseline achieved an AUCPR of 0.163 ± 0.006.

Lu et al. [14] introduced the BRIGHT framework, utilizing graph neural networks for real-time fraud detection in e-commerce marketplaces. The framework demonstrated superior performance, emphasizing efficiency and precision in real-time inference. The model, called LNN (GCN), achieved an average precision of 44.2% and an Area Under the Receiver Operating Characteristic Curve (ROC

 doi:10.20944/preprints202402.1206.v1

AUC) of 0.9276. These results outperformed the baseline models, including LightGBM (FE) and LightGBM (Bench).

Nugent's document [15], submitted in November 2022, focused on privacy-preserving fraud detection using homomorphic encryption. The study employed XGBoost and a feedforward classifier neural network, achieving better performance with encrypted inference, highlighting the significance of maintaining privacy without compromising accuracy.

Addressing the challenge of highly imbalanced datasets, Tung-Yu Wu and You-Ting Wang [16] proposed an anomaly detection framework. The experimental results reveal the effectiveness of the proposed approach, with an accuracy of 0.906. These results highlight the capability of the framework in accurately detecting credit card fraud. Utilizing deep neural networks trained in an unsupervised and adversarial manner, the study demonstrated state-of-the-art performances on benchmark datasets, emphasizing the effectiveness of their approach.

Table 2. XGBoost and NN table.

Model	Metric Selected For	AUC ROC	Average Precision	Fraudulent	Legitimate
XGBoost	AUC ROC	0.989	0.703	0.88	0.9689
	Average Precision	**0.986**	**0.774**	**0.89**	**0.981**
	Inspection	0.982	0.784	0.81	0.9977
NN	AUC ROC	0.981	0.744	0.68	0.9997
	Average Precision	**0.981**	**0.739**	**0.77**	**0.9992**
	Inspection	0.982	0.724	0.77	0.9993

Vetrivendhan and Kumar [17] proposed a Cognitive Convolution Neural Network (CCNN) for credit card fraud detection. The study surpasses traditional models such as Logistic Regressions, k-nearest Neighbors, Decision Trees, and Support Vector Machines in accuracy and complexity. Analyzing the CCFD dataset, the authors employ feature engineering and diverse machine learning classifiers. The CCNN model achieves a remarkable 98.22% to 98.47% accuracy in detecting transaction fraud with an oversampling model. The research emphasizes the limitations of current fraud detection systems, advocating for advanced machine learning techniques to adapt to evolving fraud patterns in the credit card industry.

Wang et al. [18] address class imbalance issues using the train-with-generation method with a geometric progression and evaluate model performance using various metrics. Notably, high performance is achieved for continuous columns, with average shape and pair trend scores reaching approximately 0.90 and 0.98, respectively. Implementation of the train-with-generation method significantly improves scores for categorical columns, reaching 0.78 and 0.88, respectively. Overall, the model demonstrates improved accuracy, with average shape and pair trend scores for all columns reaching 0.88 and 0.92, respectively.

In conclusion, the literature on credit card fraud detection presents a diverse range of approaches, from traditional machine learning algorithms to advanced deep learning models and innovative techniques for handling imbalanced datasets and ensuring privacy. These studies contribute valuable insights and advancements, offering a comprehensive understanding of the evolving landscape of fraud detection in financial transactions.

PROJECT DISCUSSION:

Introduction:

The challenge of credit card fraud poses a significant threat that demands urgent attention and effective solutions. To address this issue, it is crucial to devise an approach that not only outperforms conventional models but is also user-friendly. The primary objective of this section is to process the data meticulously and construct a superior model compared to the basic ones, ensuring both enhanced performance and ease of use. In pursuit of this goal, a hybrid machinelearning approach

 doi:10.20944/preprints202402.1206.v1

will be employed. This hybrid model combines the strengths of K-Nearest Neighbors (KNN) and Convolutional Neural Network (CNN) algorithms. The subsequent comparison with the fundamental model aims to showcase the efficacy and user-friendliness of the proposed hybrid solution in credit card fraud detection.

Data Source:

The dataset used for this hybrid machine-learning model for fraudulent detection was obtained from Kaggle.com. Comprising 31 attributes and 284,808 rows, the dataset includes 28 numeric variables transformed through PCA for confidentiality. Key attributes include "Time" (elapsed seconds between transactions), "Amount" (transaction amount), and "Class" (binary variable denoting fraud/non-fraud). Here in this project, we used pandas to import this credit card fraud dataset.

Data Analysis and Processing:

Before programming the hybrid model, thorough data analysis and processing are imperative to identify and rectify errors. The process includes checking data dimensions, identifying null values, and handling any discrepancies. The analysis starts by checking the variable type of each column so that further processes can be done accordingly. Here in this process of knowing the variable type of column, pandas info() is been used. The below table shows the variable type of each column.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 284807 entries, 0 to 284806
Data columns (total 31 columns):
Time      284807 non-null float64
V1        284807 non-null float64
V2        284807 non-null float64
V3        284807 non-null float64
V4        284807 non-null float64
V5        284807 non-null float64
V6        284807 non-null float64
V7        284807 non-null float64
V8        284807 non-null float64
V9        284807 non-null float64
V10       284807 non-null float64
V11       284807 non-null float64
V12       284807 non-null float64
V13       284807 non-null float64
V14       284807 non-null float64
V15       284807 non-null float64
V16       284807 non-null float64
V17       284807 non-null float64
V18       284807 non-null float64
V19       284807 non-null float64
V20       284807 non-null float64
V21       284807 non-null float64
V22       284807 non-null float64
V23       284807 non-null float64
V24       284807 non-null float64
V25       284807 non-null float64
V26       284807 non-null float64
V27       284807 non-null float64
V28       284807 non-null float64
Amount    284807 non-null float64
Class     284807 non-null int64
dtypes: float64(30), int64(1)
memory usage: 67.4 MB
```

Figure 3. Dataset Structure.

The Observation from the table shows every column in the data base is “float64” and only ‘Class’ column is ‘int64’. The next step in the analysis was to get the count of total fraud record, data was containing.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 22 February 2024 doi:10.20944/preprints202402.1206.v1

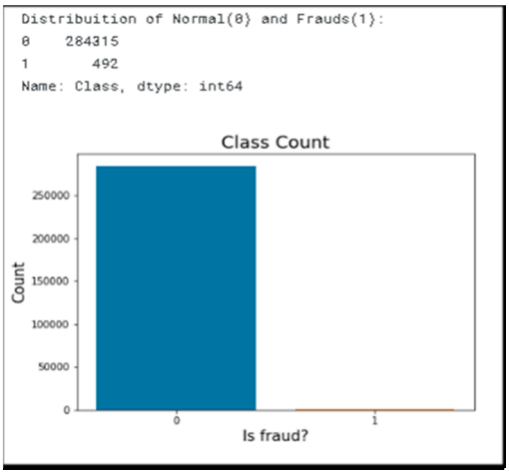

Figure 4. Count graph of number of fraud and non-fraud .

From above plotting, data set records large number of normal data and only 492 records of fraud transcations data. Now, using the fraud and normal transaction, the maximum and minimum transaction statistics can we know for this list comprehension was used on the 'Amount' column , the statistics is been shown in the below figures.

```
Fraud transaction statistics
count     492.000000
mean      122.211321
std       256.683288
min         0.000000
25%         1.000000
50%         9.250000
75%       105.890000
max      2125.870000
Name: Amount, dtype: float64

Normal transaction statistics
count    284315.000000
mean         88.291022
std         250.105092
min           0.000000
25%           5.650000
50%          22.000000
75%          77.050000
max       25691.160000
Name: Amount, dtype: float64
```

Figure 5. Description of normal and fraud transactions.

The analysis made yet shows a clear sign that the data set is not scale. To get the correct analysis the data set should be standardized. The next process will be to normalize the data.

Data Normalization and Analysis:

To ensure uniformity among variable ranges, normalization is performed using the StandardScaler module. The perks of doing normalization by the process of StandardScaler will make the data in the range of 0 to 1. Libraries of sickit learn are imported for this which are then applied on every column.

When calculating the standard score of a sample x, use the formula

$$z = \frac{(x-u)}{s}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 22 February 2024 doi:10.20944/preprints202402.1206.v1

where u is the training sample mean, or zero if with_mean=False, and s is the training sample standard deviation, or one if with_std=False. Making an object, fitting and modifying the pertinent column, and eliminating unnecessary columns from the dataset are all steps in this procedure. The heatmap, which illustrates the relationship together with its magnitude, will be used to display the relationship between the columns now that the data has been normalized. The heatmap() function of the Seaborn library was utilized, providing the dataset as input and creating the heatmap.

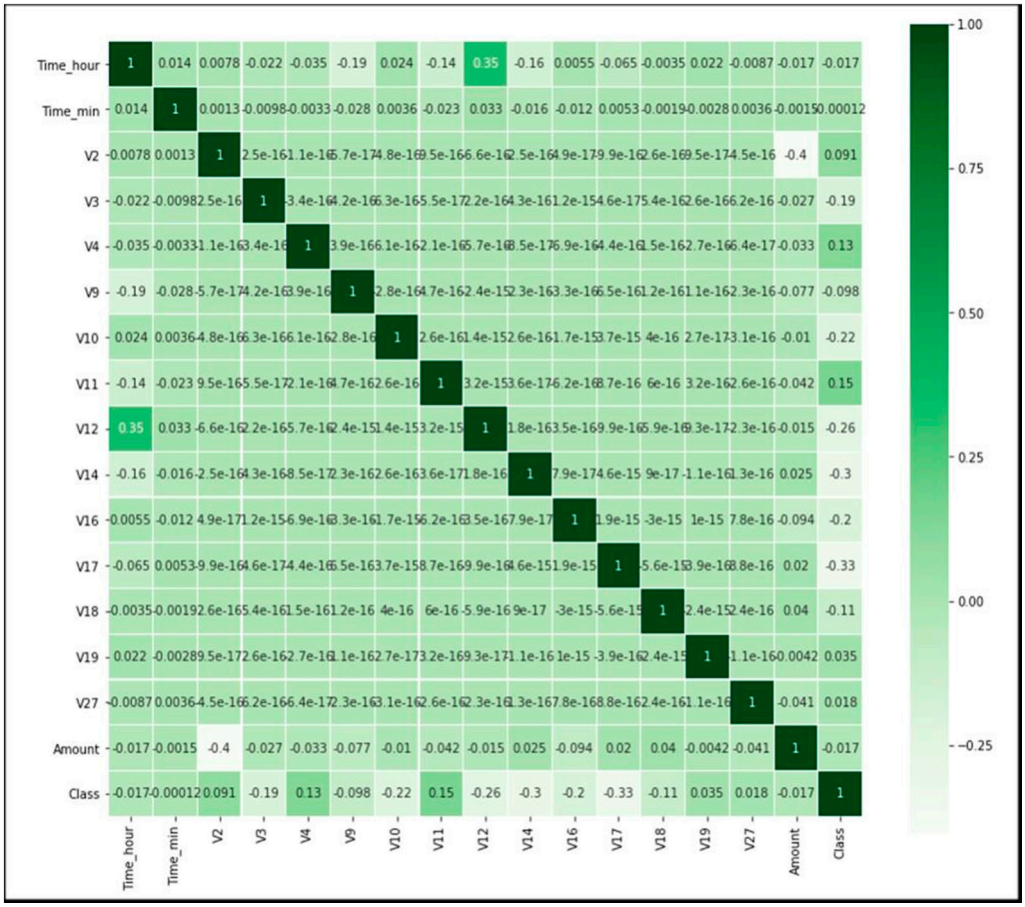

Figure 6. Heatmap of credit card dataset. Where darker the color of the box more, stronger the relation.

Data Modeling:

Now after the part of Analysis and processing. This part will focus on the part where Data is divided into independent (X) and dependent (y) parts. Utilizing the sci-kit-learn library, the dataset is split into training and testing sets with a 0.3 ratio. Pseudocode involves using train_test_split and separating columns into X and y.

Here from now on, the aim will be to use different classification techniques and to compare them with the hybrid model made from the combined strength of Random forest and CNN.

What is a Hybrid Machine Learning Algorithm?

Hybrid machine learning combines different approaches such as supervised and unsupervised learning, reinforcement learning, or deep learning to address complex problems. It can help to overcome the limitations of individual machine learning algorithms. Hybrid machine-learning architectures can combine the strengths of different machine-learning algorithms to overcome these limitations.

 doi:10.20944/preprints202402.1206.v1

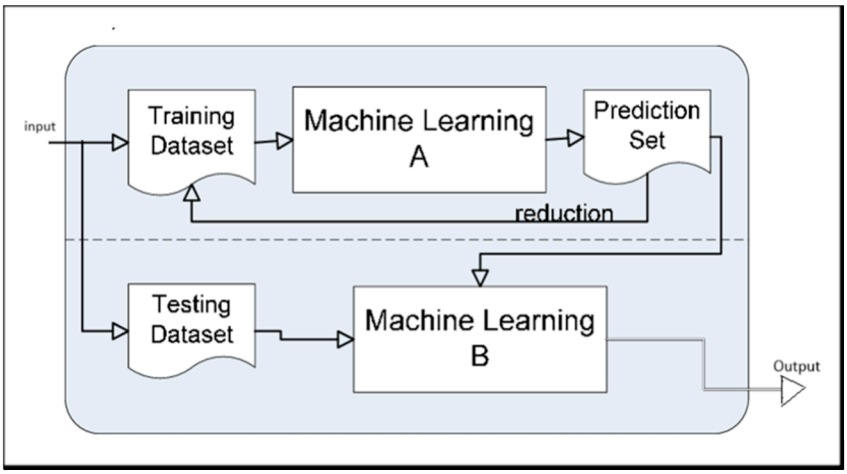

Figure 7. Basic flow of hybrid machine learning.

It allows for a more comprehensive data analysis and can lead to more accurate predictions or insights. Hybrid machine learning can be used to solve a wide variety of machine learning problems. It is a powerful tool for improving the performance, efficiency, and interpretability of machine learning models. However, it is important to carefully consider the specific needs of your problem and the challenges of hybrid machine learning before using them.

Why Hybrid Architecture is Effective for Credit Card Fraud Detection?

Hybrid architectures are well-suited for credit card fraud detection because they can be easily updated to adapt to new fraud patterns. This is important because credit card fraudsters are constantly developing new techniques to avoid detection.

Hybrid architectures can frequently outperform normal machine learning techniques or deep learning algorithms. This is because hybrid architectures can use the strengths of several machine learning methods to overcome their limitations.

When working with huge datasets, hybrid architectures can be more efficient, because hybrid architecture can break the problem into smaller subtasks, which can then be solved by individual machine learning algorithms that are more efficient for particular tasks.

Hybrid architectures can be more interpretable. This is because normal machine learning algorithms are often easier to interpret than deep learning algorithms. By combining a deep learning model with a traditional machine learning algorithm, it is possible to create a hybrid architecture that is both accurate and interpretable.

Hybrid Model Making:

This section details the creation of a hybrid model incorporating both random forest and Convolutional Neural Network (CNN) methods. The CNN model is configured with layers such as input, MaxPooling1D, Flatten, and Dense.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 22 February 2024 doi:10.20944/preprints202402.1206.v1


```
Epoch 1/10
6231/6231 [==============================] - 31s 5ms/step - loss: 0.0036 - accuracy: 0.9994
Epoch 2/10
6231/6231 [==============================] - 18s 3ms/step - loss: 0.0032 - accuracy: 0.9994
Epoch 3/10
6231/6231 [==============================] - 19s 3ms/step - loss: 0.0032 - accuracy: 0.9994
Epoch 4/10
6231/6231 [==============================] - 18s 3ms/step - loss: 0.0030 - accuracy: 0.9994
Epoch 5/10
6231/6231 [==============================] - 18s 3ms/step - loss: 0.0031 - accuracy: 0.9994
Epoch 6/10
6231/6231 [==============================] - 19s 3ms/step - loss: 0.0029 - accuracy: 0.9994
Epoch 7/10
6231/6231 [==============================] - 17s 3ms/step - loss: 0.0029 - accuracy: 0.9994
Epoch 8/10
6231/6231 [==============================] - 18s 3ms/step - loss: 0.0029 - accuracy: 0.9994
Epoch 9/10
6231/6231 [==============================] - 19s 3ms/step - loss: 0.0029 - accuracy: 0.9994
Epoch 10/10
6231/6231 [==============================] - 18s 3ms/step - loss: 0.0027 - accuracy: 0.9994
2671/2671 [==============================] - 4s 1ms/step
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d_1 (Conv1D)           (None, 27, 32)            128

 max_pooling1d_1 (MaxPoolin  (None, 13, 32)            0
 g1D)

 flatten_1 (Flatten)         (None, 416)               0

 dense_2 (Dense)             (None, 128)               53376

 dense_3 (Dense)             (None, 1)                 129

=================================================================
Total params: 53633 (209.50 KB)
Trainable params: 53633 (209.50 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 8. CNN details of Epochs and Model Structure.

Simultaneously, a RandomForestClassifier object with 100 trees (n_estimators = 100) is instantiated. Random forest uses the concept of Decision trees on each tree and the most frequently returns prediction made by each Decision tree. Now this hybrid model works as follow, Predictions from the CNN model, using "binary_crossentropy" loss and "adam" optimizer for 10 Epochs, are sent to the random forest with the testing dataset so that Random Forest could make predictions to achieve the final Predictions and this model will be our hybrid model. TensorFlow and sci-kitlearn libraries are employed for CNN and random forest modeling, respectively. This model performed very well with an accuracy of 99.981% showing very few mistakes which can be shown from the Confusion matrix;

```
Confusion Matrix:
[[85308     0]
 [   16   119]]
```

Figure 9. Hybrid model, (Here 85308 is the Correctly predicted non-Fraud value 119 is the.

Correctly predicted fraud value and 18 is the wrong value that model predicted.) *Comparing Other Model:*

 doi:10.20944/preprints202402.1206.v1

Now this section will aim to make a more model with different machine learning models and to make a comparison between and with the hybrid model. These models will include the SVM, KNN, Logistic regression, and Naïve Bayes.

SVM model:

A supervised machine learning method using linked learning algorithms is called support vector machine. It demonstrates expertise in linear classification and is applied to both regression and classification analysis. Notably, by creating boundaries between classes, it expands its capacity to include non-linear classification mistakes, In order to minimize classification mistakes, these margins are purposefully designed to maximize the distance between the margin and the classes. The Support Vactor Machine model, which scored an astounding 99.94% accuracy. 51 cases were misclassified, nevertheless.

```
Correctly Classified Instances       85391               99.9403 %
Incorrectly Classified Instances        51                0.0597 %
Kappa statistic                          0.8388
Mean absolute error                      0.0006
Root mean squared error                  0.0244
Relative absolute error                 16.4433 %
Root relative squared error             54.1917 %
Total Number of Instances            85442

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.764    0.000    0.930      0.764    0.839      0.843    0.882     0.711     1
                 1.000    0.236    1.000      1.000    1.000      0.843    0.882     1.000     0
Weighted Avg.    0.999    0.235    0.999      0.999    0.999      0.843    0.882     0.999

=== Confusion Matrix ===

     a     b   <-- classified as
   133    41 |     a = 1
    10 85258 |     b = 0
```

Figure 10. SVM details about the Confusion Matrix, mean absolute error, root mean absolute error, relative absolute error, etc.

Logistic Regression:

The relationship between a dependent qualitative variable (binary or binomial logistic regression) or a variable with three values or more (multinomial logistic regression) and one or more independent explanatory variables, whether qualitative or quantitative, is the basis for assessments made using the logistic regression model, a statistical technique.

With 75 cases of misclassification, the most current model developed using the Logistic Regression model has an accuracy of 99.92%.

Naïve Bayes:

Naïve Bayes is a classification technique that relies on conditional probability of occurrence for clustering and classifications. It views the existence of a characteristic inside a class as independent of the existence of any other feature.

With an accuracy of 97.77%, R's second model, Naïve Bayes, misclassified 2,051 transactions—33 fraudulent as nonfraudulent and 2018 nonfraudulent as fraudulent. The accuracy of the Naïve Bayes model developed using Weka differs somewhat; it is 97.73%, and there are 1,938 occurrences of misclassification.

 doi:10.20944/preprints202402.1206.v1


```
Confusion Matrix and Statistics

                Reference
Prediction       Not Fraudulent Fraudulent
  Not Fraudulent          89684         33
  Fraudulent               2018        139

               Accuracy : 0.9777
```

Figure 11. Naïve Bayes details.

KNN:

The K-Nearest Neighbor algorithm (KNN) is a supervised machine learning technique applicable to both classification and regression scenarios. Here for this problem value of k was taken as 3, which resulted in 99.83% where it managed to correctly identify 91,719 transactions and missed 155 transactions.

```
Confusion Matrix and Statistics

                Reference
Prediction       Not Fraudulent Fraudulent
  Not Fraudulent          91702        155
  Fraudulent                  0         17

               Accuracy : 0.9983
```

Figure 12. KNN details.

Conclusion

So now on comparing, you can see the hybrid model (accuracy of 99.981%) performed much better than the other machine learning methods (highest accuracy was 99.94%). Now our work was not just to show that the hybrid model works on a particular dataset, but we can conclude that for this problem statement, the Hybrid model strengthened with CNN and the Random Forest model will always work better. As datasets related to this problem statement will always be solved using classification models and regression models, thus CNN and random forest models are best to use for the hybrid model. From this, we conclude that the hybrid model will be the best option for credit card fraud detection.

 doi:10.20944/preprints202402.1206.v1

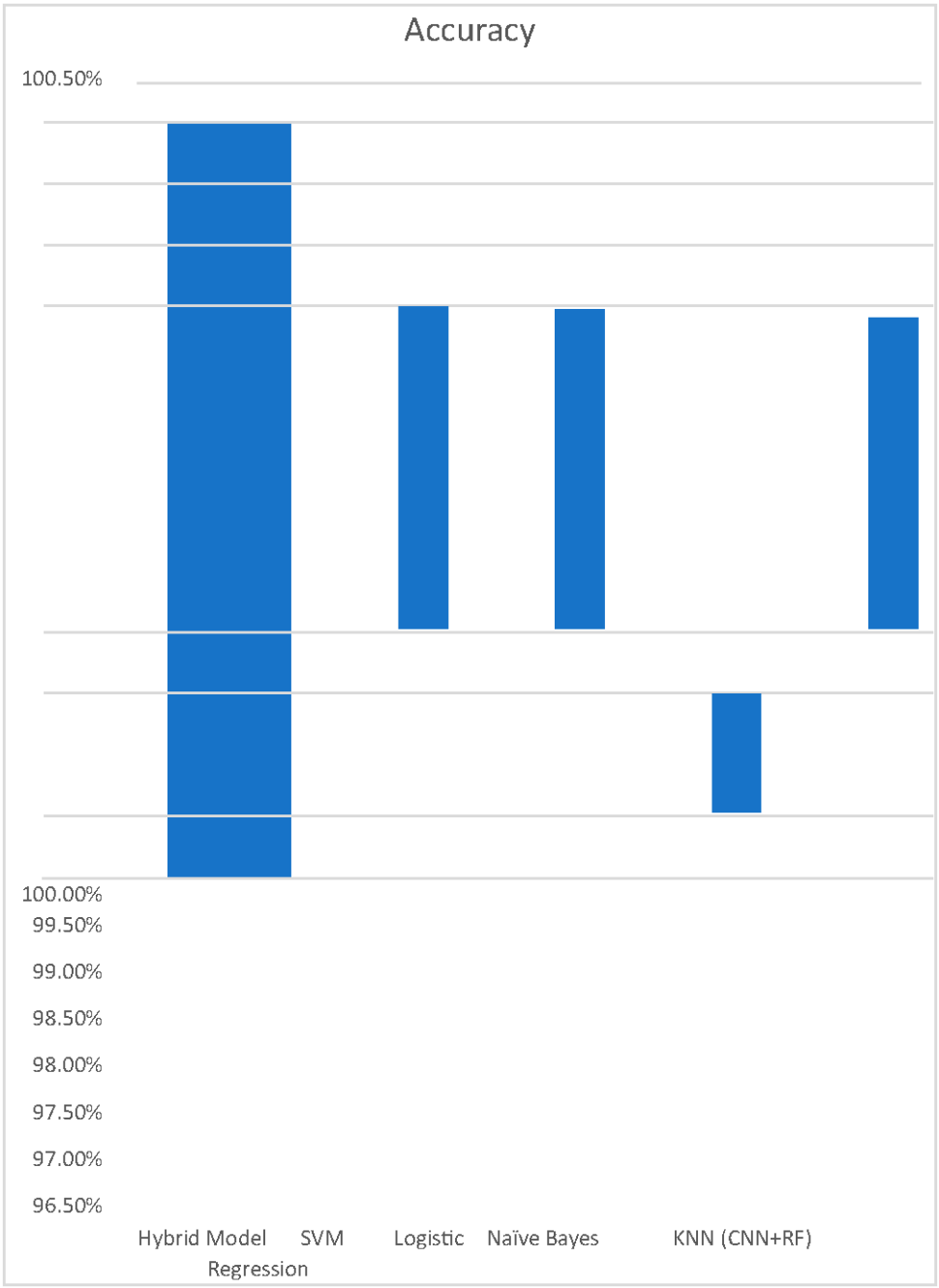

Figure 13. Accuracy Chart of Different Models.

References

1. Mehbodniya, A., Alam, I., Pande, S., Neware, R., Rane, K. P., Shabaz, M., & Madhavan, M. V. (2021). Financial fraud detection in healthcare using machine learning and deep learning techniques. Security and Communication Networks, 2021, 1-8.
2. Zorion, P. K., Sachan, L., Chhabra, R., Pandey, V., & Fatima, D. H. (2023). Credit Card Financial Fraud Detection Using Deep Learning. Available at SSRN 4629093.
3. Rao, S. X., Zhang, S., Han, Z., Zhang, Z., Min, W., Chen, Z., ... & Zhang, C. (2020). xFraud: explainable fraud transaction detection. arXiv preprint arXiv:2011.12193. https://10.14778/3494124.3494128
4. Shen, A., Tong, R., & Deng, Y. (2007, June). Application of classification models on credit card fraud detection. In 2007 International conference on service systems and service management (pp. 1-4). IEEE. https://doi.org/10.1109/ICSSSM.2007.4280163
5. Xu, B., Wang, Y., Liao, X., & Wang, K. (2023). Efficient fraud detection using deep boosting decision trees. Decision Support Systems, 114037. https://arxiv.org/pdf/2302.05918

6. Dornadula, V. N., & Geetha, S. (2019). Credit card fraud detection using machine learning algorithms. Procedia computer science, 165, 631-641. https://doi.org/10.1016/j.procs.2020.01.057
7. Pang, G., Shen, C., & van den Hengel, A. (2019, July). Deep anomaly detection with deviation networks. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 353-362). https://doi.org/10.1145/3292500.3330871
8. Maniraj, S. P., Saini, A., Ahmed, S., & Sarkar, S. (2019). Credit card fraud detection using machine learning and data science. International Journal of Engineering Research, 8(9), 110-115. https://www.researchgate.net/publication/336800562_Credit_Card_Fraud_Detection_using_Machine_Learning_and_Data_Science
9. Lucas, Y., Portier, P. E., Laporte, L., He-Guelton, L., Caelen, O., Granitzer, M., & Calabretto, S. (2020). Towards automated feature engineering for credit card fraud detection using multi-perspective HMMs. Future Generation Computer Systems, 102, 393-402. https://doi.org/10.1016/j.future.2019.08.029
10. Ayorinde, K. (2021). A Methodology for Detecting Credit Card Fraud. Minnesota State University, Mankato. https://cornerstone.lib.mnsu.edu/cgi/viewcontent.cgi?article=2167&context=e tds
11. Mekterović, I., Karan, M., Pintar, D., & Brkić, L. (2021). Credit card fraud detection in card-not-present transactions: Where to invest? Applied Sciences, 11(15), 6766. https://doi.org/10.3390/app11156766
12. Ileberi, E., Sun, Y., & Wang, Z. (2022). A machine learning-based credit card fraud detection using the GA algorithm for feature selection. Journal of Big Data, 9(1), 1-17. https://doi.org/10.1186/s40537-022-00573-8
13. Xiang, S., Zhu, M., Cheng, D., Li, E., Zhao, R., Ouyang, Y., ... & Zheng, Y. (2023, June). Semi-supervised credit card fraud detection via attribute-driven graph representation. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 37, No. 12, pp. 14557-14565).
14. Dastidar, K. G., Granitzer, M., & Siblini, W. (2022, May). The Importance of Future Information in Credit Card Fraud Detection. In International Conference on Artificial Intelligence and Statistics (pp. 10067-10077). PMLR.
15. Lu, M., Han, Z., Rao, S. X., Zhang, Z., Zhao, Y., Shan, Y., ... & Jiang, J. (2022, October). BRIGHT-Graph Neural Networks in Real-Time Fraud Detection. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management (pp. 3342-3351). https://doi.org/10.1145/3511808.3557136
16. Nugent, D. (2022). Privacy-Preserving Credit Card Fraud Detection using Homomorphic Encryption. arXiv preprint https://arXiv:2211.06675
17. Wu, T. Y., & Wang, Y. T. (2021). Locally interpretable one-class anomaly detection for credit card fraud detection. In 2021 International Conference on Technologies and Applications of Artificial Intelligence (pp. 25-30). IEEE. https://doi.org/10.1109/TAAI54685.2021.00014
18. Vetrivendan, L.., & Kumar, G.. (2023). CCNN: An Artificial Intelligent based Classifier to Credit Card Fraud Detection System with an Optimized Cognitive Learning Model. International Journal on Recent and Innovation Trends in Computing and Communication, 11(5s), 159–171. https://doi.org/10.17762/ijritcc.v11i5s.6640