**A Deep Learning Ensemble with Data Resampling for Credit Card Fraud Detection**

**Alternative Title:**

Credit card Fraud Detection using Machine Learning and Deep Learning Algorithms.

**Aim:**

People can use credit cards for online transactions as it provides an efficient and easy-to-use facility. With the increase in usage of credit cards, the capacity of credit card misuse has also enhanced. Credit card frauds cause significant financial losses for both credit card holders and financial companies. The main aim is to detect fraudulent transactions using credit cards with the help of ML algorithms and deep learning algorithms.

**Abstract:**

Credit card fraud remains a pervasive issue, leading to substantial financial losses for both financial institutions and cardholders. To combat this threat effectively, this study presents a novel approach to credit card fraud detection using a deep learning ensemble coupled with data resampling techniques. The proposed system combines multiple deep learning models to enhance the classification of fraudulent transactions, while employing resampling methods to address the class imbalance prevalent in credit card transaction data. Through extensive experimentation and evaluation on diverse datasets, our ensemble demonstrates notable improvements in fraud detection accuracy, outperforming single-model approaches and conventional sampling techniques. The results reveal the system's robustness in identifying fraudulent activities while minimizing false alarms, providing a valuable tool for financial security and risk mitigation in today's digital transaction landscape.

**Existing System:**

The relevant literature present many machines learning based approaches for credit card detection, such as Decision Tree classifier, K-Nearest neighbors, Random Forest classifier, Support Vector classifier, Logistic Regression and XG Boost, which results low accuracy. In 2020, there were 393,207 cases of CCF out of approximately 1.4 million total reports of identity theft. CCF is now the second most prevalent sort of identity theft recorded as of this year, only following government documents and benefits fraud. In 2020, there were 365,597 incidences of fraud perpetrated using new credit card accounts. The number of identity theft complaints has climbed by 113% from 2019 to 2020, with credit card identity theft reports increasing by 44.6%. Payment card theft cost the global economy $24.26 billion last year. With 38.6% of reported card fraud losses in 2018, the United States is the most vulnerable country to credit theft.

**Problem Definition:**

By proposing Machine learning Algorithms, based approaches for credit card detection, such as Extreme Learning Method, Decision Tree classifier, K-Nearest neighbors, Random Forest classifier, Support Vector classifier, Logistic Regression and XG Boost. The model results leds to low accruracy.

**Proposed System:**

Deep learning (DL) algorithms applied applications in computer network, intrusion detection, banking, insurance, mobile cellular networks, health care fraud detection, medical and malware detection, detection for video surveillance, location tracking, Android malware detection, home automation, and heart disease prediction. we explore DL Algorithms to identify credit card thefts in the banking industry in this model. It uses a number of deep learning algorithms for detecting CCF. However, in this model, we choose the CNN model and its layers to determine if the original fraud is the normal transaction of qualified datasets.

**Modules:**

* Dataset Collection
* Algorithm
* Detection

**Dataset Collection:**

The credit card dataset is accessible for research purposes. The dataset holds transactions made by a cardholder over a time period, Disclosing a consumer’s transaction details is considered a problem of confidentiality, the main component analysis is applied to the majority of the dataset’s features using principal component analysis (PCA). PCA is a standard and widely used technique in the relevant literature for reducing the dimensionality of such datasets, increasing interpretability but at the same time minimizing information loss. It does so by creating new uncorrelated variables that successively maximize variance. The detail of the dataset containing time, V1, V2, V3. V28 as PCA applied features, amount, and class labels.

**Data Preprocess:**

We propose to alter the DL algorithm of the CNN model by adding the additional layers for features extraction and the classification of credit card transactions as fraudulent or otherwise. In this model, the main aim is to detect fraudulent transactions using credit cards with the help of deep learning algorithms. In First process, the imbalanced CCF dataset is transformed into a balanced dataset by removing non fraudulent transactions from the dataset. In a real-world transaction, fraudulent and non-fraudulent classes are not balanced due to the nature of the problem. For instance, if one million transactions are performed in a day, only a few can be fraudulent. The convolutional neural network (CNN) model with layers architecture is applied to the balanced dataset to validate the proposed model. The model is trained over number of epochs. The CNN layers architecture obtained above 90.00% in training and validation accuracy respectively. The accuracy and loss of CNN model using the balanced CCF dataset.

**Algorithm:**

Convolutional neural network (CNN): In deep learning, a convolutional neural network (CNN / ConvNet) is a class of deep neural networks, most commonly applied to analyze visual imagery. Now when we think of a neural network we think about matrix multiplications but that is not the case with ConvNet. It uses a special technique called Convolution. Now in mathematics convolution is a mathematical operation on two functions that produces a third function that expresses how the shape of one is modified by the other. Object detection is widely used for image processing and classification, estimating time series and detecting differences. Layers in the CNN Model: Here are six distinct layers in the CNN models are Input layer, Convo layer (Convo C ReLU), Pooling layer, Fully connected layer (FC), SoftMax /logistic layer, Output layer. Creation of the Model, The pipeline of CNN model over keras includes convo layer, max pooling layer, dropout layer, convo layer, max pooling layer, dropout layer along with two fully connected layers sequentially, depicts input neural network and output of dropout layer. Compile the Model: Categorical Cross-Entropy: We build binary cross-entropy at prior portions and in ML. At that time, we used definite cross-entropy.

PERFORMANCE-EVALUTION MEASURES: Traditional methods of estimating ML classifiers can use confusion metrics, precision, recall and F1 score relating to the difference between the rock bottom dataset truth and the model's prediction where TP, TN, FP, and FN denote true positive, true negative, false positive and false negative, respectively.

**Prediction:**

The imbalanced CCF dataset is transformed into a balanced dataset by removing non fraudulent transactions from the dataset. In a real-world transaction, fraudulent and non-fraudulent classes are not balanced due to the nature of the problem. For instance, if one million transactions are performed in a day, only a few can be fraudulent. Our proposed model has 20 layers: a convolutional layer with a kernel size of 32 X 2 and a ReLU activation function, followed by a batch normalization layer and a dropout layer with a dropout rate of 0.2. Then, we add another convolutional layer with a kernel size of 64 X 2 and a ReLU activation function, followed by a batch normalization layer and a dropout layer with a dropout rate of 0.5. Then, we add another convolutional layer with a kernel size of 64 X 2 and a ReLU activation function, followed by a batch normalization layer and a dropout layer with a dropout rate of 0.5. Then, we add another convolutional layer with a kernel size of 64X 2 and a ReLU activation function, followed by a batch normalization layer and a dropout layer with a dropout rate of 0.25. Then, we add a attened layer with a kernel size of 64 X2 and a ReLU activation function, followed by a dense layer and a dropout layer with a dropout rate of 0.5, followed by 3 dense layers. The first dense layer has a ReLU activation function of (100). The second dense layer has a ReLU activation function of (50). The third dense layer has a ReLU activation function of (25). Finally, we add a dense layer for classification with a sigmoid activation function. At n-number epochs, the accuracy is above 90.00%..

**Hardware Requirements:**

* Hard Disk : 500GB and Above
* RAM : 4GB and Above
* Processor : I3 and Above

**Software Requirements:**

* Operating System : Windows 10 (64 bit)
* Software : Python-3.6.3
* Tools : Anaconda

**Conclusion:**

CCF is an increasing threat to financial institutions. Fraud sterst end to constantly come up with new fraud methods. A robust classifier can handle the changing nature of fraud. Accurately predicting fraud cases and reducing false-positive cases is the foremost priority of a fraud detection system. The type of input data is a dominant factor that drives different ML methods. For detecting CCF, the number of features, number of transactions, and correlation between the features are essential factors in determining the model's performance. DL methods, such as CNNs and their layers, are associated with the processing of text and the baseline model.

Using these methods for the detection of credit cards yields better performance than traditional algorithms. Comparing all the algorithm performances side to side, the CNN with layers and the baseline model is the top method with an accuracy of above 90.00%.

**Future Enhancement:**

Numerous sampling techniques are used, CCF Detection Using State-of-the-Art ML and DL Algorithms to increase the performance of existing examples, but they significantly decrease on the unseen data. The performance on unseen data increased as the class imbalance increased. Future work associated may explore the use of more state of art deep learning methods to improve the performance of the model proposed in this study.

**Architecture diagram:**

Pre-processed Dataset

Feature Extraction

Convolutional Neural Networks

Real time

User input

User interface

Prediction



Model compile

Raw dataset

Trained Model