**ONLINE PAYMENT FRAUD DETECTION USING CNN**

**SCOPE:**

Fraud is one of the major ethical issues in transaction sector. In this era we can see many innovative financial services like ATMs, online banking, etc. Besides, along with the rapid advances of e-commerce, the use of credit cards has become a convenient and necessary part of financial life. A credit card is a payment card supplied to customers as a system of payment. Using a third-person credit card or its information without the knowledge of that person is referred to as credit card fraud. Application and Behavioral fraud are major types of fraud where people are easily tricked and lose their money. The same user may submit multiple applications which may lead to identical fraud. Application fraud takes place when fraudsters apply for new cards from a bank or issuing companies using false or others' information. In this project, we will be applying some supervised and unsupervised algorithms and will classify the credit card dataset. We will use CNN and correlate the data train and get the model's accuracy. Keywords: CCFD, CNN, Logistic Regression, Feature extraction.

**PURPOSE:**

This Project is focused on online payment fraud detection in real-world scenarios. The main aim of the project is to create a fraud detection system. Our vision is to create a detection system that detects fraudulent transactions to help people and create awareness about this deceit. . We are going to apply machine learning algorithms to past data and predict the possibility of a transaction being a fraud transaction. Purchasing items from e-commerce sites and paying via credit card has become common. To solve this problem, we will be using some machine learning methods like CNN the dataset will be divided into a training set and test set. The algorithms used in the experiment are KNN,SVM and CNN. To detect fraud in a transaction we need to perform and use complex algorithms and functions. In this project, we will try to train our dataset to get max accuracy to detect fraud transactions.

**TOOLS USED:**

**Spyder -version 5**

Spyder is a free and open source scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts. It features a unique combination of the advanced editing, analysis, debugging, and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and beautiful visualization capabilities of a scientific package.

**Vscode**

**Visual Studio Code** is a text editor which supports HTML formatting along with providing HTML extensions that saves time, are efficient, and are free to use, CNN .

**CNN tools:**

**Keras**

* Keras is an open-source deep learning framework for python
* It has been developed by artificial intelligence researchers.
* Keras is an API design for human beings, not machines.
* It minimizes the number of user actions required for common use cases and provides clear and actionable error messages.

**Tensorflow**

* TensorFlow is a Google product, which is one of the most famous deep learning tools widely used in the area of machine learning and deep neural network
* It can easily run On multiple CPUs as well as GPUs as well as on mobile operating systems.
* It can operate in various programming languages such *as* Java, C++, or Python.

**Pandas**

Pandas is an open source python packages that is most widely used for dat science/data analysis and machine learning tasks.

**Sklearn**

scikit-learn (formerly scikits.learn and also known as sklearn) is a [free software](https://en.wikipedia.org/wiki/Free_software) [machine learning](https://en.wikipedia.org/wiki/Machine_learning) [library](https://en.wikipedia.org/wiki/Library_(computing)) for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [programming language](https://en.wikipedia.org/wiki/Programming_language). It features various [classification](https://en.wikipedia.org/wiki/Statistical_classification), [regression](https://en.wikipedia.org/wiki/Regression_analysis) and [clustering](https://en.wikipedia.org/wiki/Cluster_analysis) algorithms including [support-vector machines](https://en.wikipedia.org/wiki/Support_vector_machine), [random forests](https://en.wikipedia.org/wiki/Random_forests), [gradient boosting](https://en.wikipedia.org/wiki/Gradient_boosting), [*k*-means](https://en.wikipedia.org/wiki/K-means_clustering) and [DBSCAN](https://en.wikipedia.org/wiki/DBSCAN), and is designed to interoperate with the Python numerical and scientific libraries [NumPy](https://en.wikipedia.org/wiki/NumPy) and [SciPy](https://en.wikipedia.org/wiki/SciPy).

**TECHNOLOGIES USED:**

Python

Flask

Html,Css

**PROOF OF CONTENT :**

METHODOLOGY

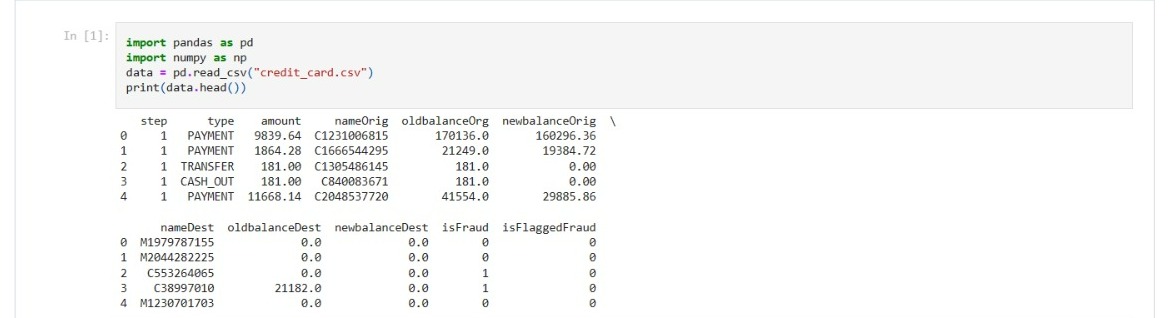
*A. Model Architecture*

After importing the dataset and analyzing the dataset and performing data exploration and statistical analysis we further balance the data and separate the dataset to test and train and build a CNN model. and then we have split the dataset into fraud and non-fraud data and created a balance data for checking fraud transactions.

B. Algorithm

Step 1: Dataset is imported

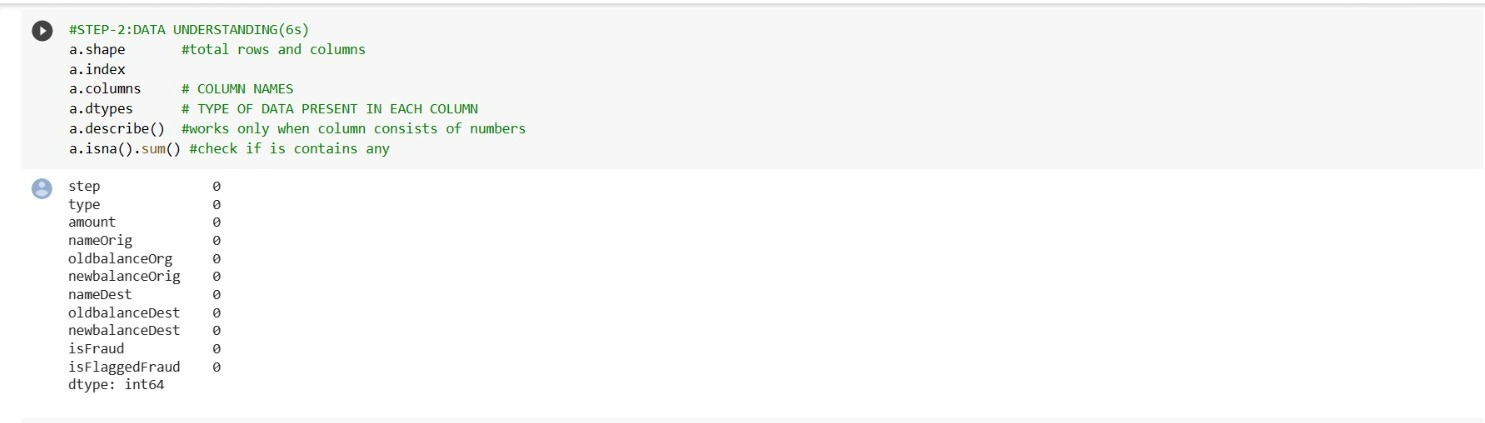
Dataset is imported



STEP 2: DATA VISUALIZATION

Analyzing the dataset and performing data exploration and statistical analysis .

| **Column Name** | **Description** |
| --- | --- |
| step | represents a unit of time where 1 step equals 1 hour |
| type | type of online transaction |
| amount | the amount of the transaction |
| nameOrig | customer starting the transaction |
| oldbalanceOrg | balance before the transaction |
| newbalanceOrig | balance after the transaction |
| nameDest | recipient of the transaction |

**

Step3: Data Preprocessing



1.As a part of data refining we removed various columns whose information is unwothy.

2.we made sure that no strings are present in the data,before giving to machine learning models.

3.using label encoding strings are assigned with different integers from 1 to 5.

STEP 4: Splliting the data for training and testing

**

80% data is allocated for training the model.

20% data is allocated for testing the data.

STEP 5: TRAINING and TESTING

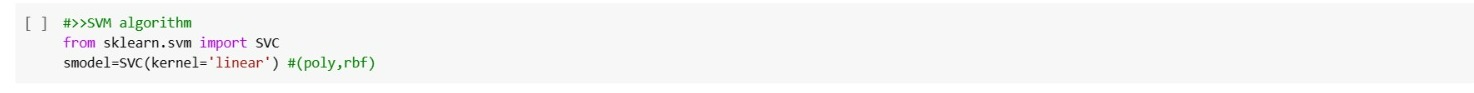
Implemented the project with different machine learning models

1. Implementation with KNeighborsClassifier algorithm

**

After model evaluation with KNeighborsClassifier algorithm the accuracy for the model obtained is 93.5764654

2.implementation with SVM algorithm



After model evaluation with KNeighborsClassifier algorithm the accuracy for the model obtained is 95.5764654

STEP 6: MODEL BUILDING

Implementing with convolutional neural network (deep learning algorithm)



Convolutional Neural Network consists of multiple layers like the input layer, Convolutional layer, Pooling layer, and fully connected layers.



The Convolutional layer applies filters to the input image to extract features, the Pooling layer downsamples the image to reduce computation, and the fully connected layer makes the final prediction. The network learns the optimal filters through backpropagation and gradient descent.

Activation functions used in our data are relu and sigmoid in the convolution neural network.

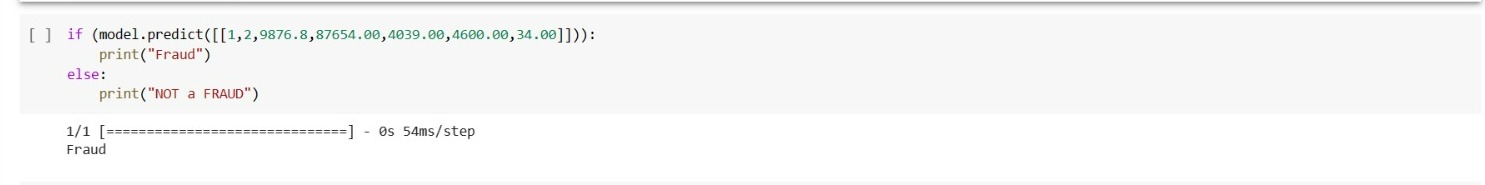
STEP 7: Model evaluation

Model evaluation is done by running the epochs. The number of epochs is a hyperparameter that defines the number times that the learning algorithm will work through the entire training dataset.

The dataset is trained with 20 epochs.



STEP 8: FRAUD PREDICTION



Tranasction details are given to the trained deep learning model and fraud

transactions are identified.

**REPOSITORY DETAILS:**

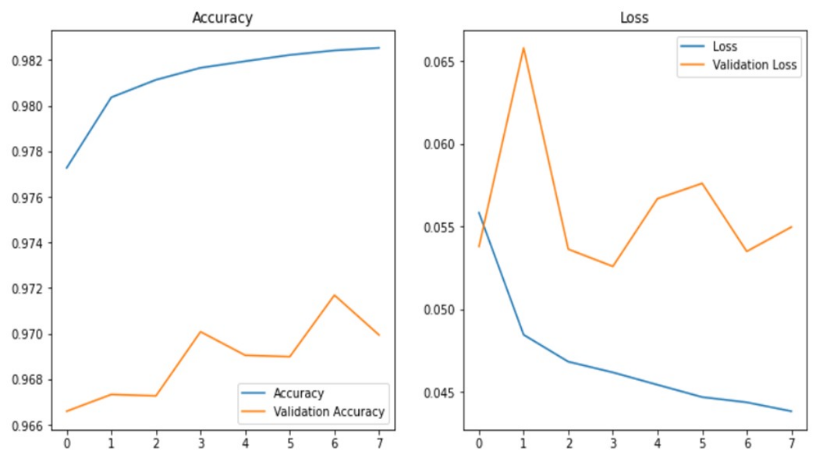
DATASET ADDRESS: <https://www.kaggle.com/datasets/ealaxi/paysim1?resource=download>

**OUTCOME:**

The accuracy in predicting the fraud transaction obtained with an accuracy of 99.866544.

The loss obtained while calculating the difference between machine predicted data and the original data is rectified .

The graphs related to accuracy and loss is shown be



**CONCLUSION**

Accuracies obtained in machine learning algorithms

KNN=93.467257824

SVM=95.4676493

CNN = 99.65348

In this Credit Card Fraud Detection System project, machine learning algorithms were briefly compared in fraud detection, also a detailed comparative analysis is provided. It is seen that various algorithms give different accuracy in performance. Even though there are numerous fraud discovery approaches we can't state that this specific method distinguishes the fraud. At the point when we look at the precision, recall, and F1-score are 0.8701, 0.5826, and 0.679166 using Logistic Regression accuracy is 0.999185. Accuracy of 0.9539 and loss of-0.1108 using CNN are obtained. Hence we will analyze the possibility of a transaction being a fraud transaction.