**CREDIT CARD FRAUD DETECTION USING TENSORFLOW AND DEEP LEARNING REPORT**

**NAME : AYUSHI PANDEY**

**SECTION : KM073**

**REG NO: 11809430**

**ROLL NO: A29**

**COURSE CODE: INT 248**

**Github: https://github.com/Ayushi7877/credit-card-fraud-detection/blob/main/INT248.ipynb**

**INTRODUCTION**

Fraud detection is a set of activities that are taken to prevent money or property from being obtained through false pretenses.

Fraud can be committed in different ways and in many industries. The majority of detection methods combine a variety of fraud detection datasets to form a connected overview of both valid and non-valid payment data to make a decision. This decision must consider IP address, geolocation, device identification, “BIN” data, global latitude/longitude, historic transaction patterns, and the actual transaction information. In practice, this means that merchants and issuers deploy analytically based responses that use internal and external data to apply a set of business rules or analytical algorithms to detect fraud.

**Credit Card Fraud Detection with Machine Learning** is a process of data investigation by a Data Science team and the development of a model that will provide the best results in revealing and preventing fraudulent transactions. This is achieved through bringing together all meaningful features of card users’ transactions, such as Date, User Zone, Product Category, Amount, Provider, Client’s Behavioural Patterns, etc. The information is then run through a subtly trained model that finds patterns and rules so that it can classify whether a transaction is fraudulent or is legitimate.

**The Techniques of Credit Card Fraud and Prevention**

| **Rank** | **Category** | **# of Reports** |
| --- | --- | --- |
| 1 | Internet Services | 62,942 |
| 2 | Credit Cards | 51,129 |
| 3 | Healthcare | 47,410 |
| 4 | Television and Electronic Media | 38,336 |
| 5 | Foreign Money Offers and Counterfeit Check Scams | 27,443 |
| 6 | Computer Equipment and Software | 18,350 |
| 7 | Investment-Related | 14,884 |

**PROPOSED ARCHITECTURE:**

Here we are using autoencoder model which is basically unsupervised training for fraud detection.

Importing the libraries

Feature normalization for finding min – max score (sigmoid activation)

Feature normalization z score tanh

Training and testing the time series dataset where we use 75% for training level and 25% for testing.

Data exploration

Build graph for fc layers

Getting auto encoder embedding

Building binary classifier for predicting fraud using auto encoder

For fraud case

For non fraud case

Display fraud score

Visualizing predictions

Test model

Train and validate the model

Impelementing the model (using autoencoder & unsupervised learning)

**DATA SET USED**

**Link path: /creditcard.csv.zip**

The dataset contains transactions made by credit cards in September 2013 by European cardholders.

This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

It contains only numerical input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2, … V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependant cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

**RESULT AND EXPERIMENTAL ANALYSIS**

We find that using autoencoder as unsupervised learning and training the dataset and we see that here 1 hidden layer is hidden

Here we get auc score as 0.940650.

Then we visualize the predictions and get fraud score (mse) distribution for non fraud cases.

Here we visualize prediction for fraud cases and get fraud score.

Our precision of accuracy above threshold is 7.86% of fraud in test set is 0.132%.

Observation: our precision increased by a factor of 60 from 0.132% to 7.86% however detection prediction is still low (below 8%) but this is mainly due to the overall percentage of fraud cases is really too low.

If we use autoencoder and build a binary classifier that predicts fraud we get auc score as 0.9668895919781596.

**Screenshots**

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

**CONCLUSION AND FUTURE SCOPE**

Research related to Fraud Detection has been around forover 20 years now and has used various methods from manual checking to customer end authentication. Machinelearning models have also had wide successes in this area.Deep learning models have been recently adopted in manyapplications enabled by the rise in higher computation power

and cheap computing cost. This paper provides an empirical investigation comparingvarious machine learning and deep learning models on different data sets for the detection of fraudulent transaction.

The main aim of this study is to find insights of which methods would best suitable for which type of datasets. As nowadays, many companies are investing in new techniques to improve their business this paper could potentially help practitioners and companies to better understand how

different methods work on certain types of datasets. Our study reveals that to detect fraud, the best methods with larger datasets would be using SVMs, potentially combined with CNNs to get a more reliable performance. For the smaller datasets, ensemble approaches of SVM,

Random Forest and KNNs can provide good enhancements .connvolutional Neural Networks (CNN) usually, outperforms other deep learning methods such as Autoencoders, RBM

and DBN.

The idea of implementing neural networks for fraud

detection is worth further exploration. The main problem

with obtaining good NN models is the availability of

appropriate training data. We are currently looking into more

data sets and availability of larger sized fraud transactions to

be able to train larger models.

We are also looking into other models of deep learning

networks. In particular Generative Adversarial Networks

(GANs) have been used for anomaly detection in the field of

cybersecurity to detect cyber-attacks. Some research work

has used GANs to simultaneously train both the generator

and discriminator on healthy eyes data to identify anomalies

in the human eyes. This could be extended to the application

of fraud detection.

The idea of implementing neural networks for fraud detection is worth further exploration. The main problem with obtaining good NN models is the availability of appropriate training data. We are currently looking into more data sets and availability of larger sized fraud transactions to be able to train larger models. We are also looking into other models of deep learning networks. In particular Generative Adversarial Networks (GANs) have been used for anomaly detection in the field of cybersecurity to detect cyber-attacks. Some research work has used GANs to simultaneously train both the generator and discriminator on healthy eyes data to identify anomalies in the human eyes. This could be extended to the application of fraud detection.

**REFERENCES**

Dataset from Kaggle

[www.analyticsvidhya.com](http://www.analyticsvidhya.com) for reference

deep learnng from scratch -O’reilly

deep learning by jan goodfellow.