iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VIII **Month of publication:** Aug 2023

DOI: https://doi.org/10.22214/ijraset.2023.55152

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VIII Aug 2023- Available at www.ijraset.com

Credit Card Fraud Detection Using Autoencoder & Decoder ML

Arshiya Mohammad [1], V Anil Santosh[2], D D D Suribabu[3]
[1]*M-Tech student Department of Computer Science and Engineering, International School of Technology and Sciences (For Women), Andhra Pradesh, India*
[2]*Associate Professor Department of Computer Science and Engineering, International School of Technology and Sciences (For Women), Andhra Pradesh, India*
[3]*Associate Professor & Head of Department of Computer Science and Engineering, International School of Technology and Sciences (For Women), Andhra Pradesh, India*

Abstract: ***In recent years credit card fraud has become one of the growing problems. It is vital that credit card companies can identify fraudulent credit card transactions, so that customers are not charged for the items that they did not purchase. The reputation of companies will heavily damage and endangered among the customers due to fraud in financial transactions The fraud detection techniques were increasing to improve accuracy to identify the fraudulent transactions. This project intends to build an unsupervised fraud detection method using autoencoder. An Autoencoder with four hidden layers which has been trained and tested with a dataset containing a European cardholder transaction that occurred in two days with 284,807 transactions from September 2013.***

Keywords: Credit Card, Fraud Detection, Autoencoder Network, Unsupervised Learning

I. INTRODUCTION

A credit card is a thin handy plastic card that contains identification information such as a signature or picture, and authorizes the person named on it to charge purchases or services to his account - charges for which he will be billed periodically. They have a unique card number which is of utmost importance. Its security relies on the physical security of the plastic card as well as the privacy of the credit card number. There is a rapid growth in the number of credit card transactions which has led to a substantial rise in fraudulent activities. Credit card fraud is a wide-ranging term for theft and fraud committed using a credit card as a fraudulent source of funds in each transaction. Generally, statistical methods and many data mining algorithms are used to solve this fraud detection problem. Most of the credit card fraud detection systems are based on artificial intelligence, Meta learning and pattern matching. Fraud detection is a binary classification problem in which the transaction data is analyzed and classified as "legitimate" or "fraudulent". Credit card fraud detection techniques are classified in two general categories: fraud analysis (misuse detection) and user behavior analysis (anomaly detection).

A. Problem Statement

The Credit Card Fraud Detection Problem includes modeling past credit card transactions with the knowledge of the ones that turned out to be a fraud. This model is used to identify whether a new transaction is fraudulent or not. Our aim here is to detect 100% of the fraudulent transactions while minimizing the incorrect fraud classifications.

II. LITERATURE REVIEW

Illegal use of a credit card or its information without the knowledge of the owner is referred to as credit card fraud. Different credit card fraud tricks belong mainly to two groups of application and behavioral fraud [3]. Application fraud takes place when fraudsters apply for new cards from banks or issuing companies using false or other's information. Multiple applications may be submitted by one user with one set of user details (called duplication fraud) or different users with identical details (called identity fraud). Behavioral fraud, on the other hand, has four principal types: stolen/lost card, mail theft, counterfeit card and „card holder not present" fraud. Stolen/lost card fraud occurs when fraudsters steal credit card or get access to a lost card.

Mail theft fraud occurs when the fraudster gets a credit card in mail or personal information from the bank before reaching the actual cardholder [3]. In both counterfeit and „card holders not present" frauds, credit card details are obtained without the knowledge of card holders.

In the former, remote transactions can be conducted using card details through mail, phone, or the Internet. In the latter, counterfeit cards are made based on card information. Based on statistical data stated in [1] in 2012, the high-risk countries facing credit card fraud threat is illustrated in Fig.1. Ukraine has the most fraud rate with a staggering 19%, which is closely followed by Indonesia at 18.3% fraud rate. After these two, Yugoslavia with the rate of17.8% is the riskiest country. The next highest fraud rate belongs to Malaysia (5.9%), Turkey (9%) and finally the United States. Other countries that are prone to credit card fraud with the rate below than 1% are not demonstrated.

III. PROPOSED METHODOLOGY

The model needs to classify the incoming transactions into fraudulent or normal transactions. There are several methods to build a binary classifier. We are proposing to use Autoencoder which are unsupervised learning model which reconstructs the compressed input for better classification and reduce the noise in the input data.

Autoencoders are neural networks. Neural networks are composed of multiple layers, and the defining aspect of an autoencoder is that the input layers contain exactly as much information as the output layer. The reason that the input layer and output layer have the exact same number of units is that an autoencoder aims to replicate the input data. It outputs a copy of the data after analyzing it and reconstructing it in an unsupervised fashion. The data that moves through an autoencoder is not just mapped straight from input to output, meaning that the network does not just copy the input data. There are three components to an autoencoder: an encoding (input) portion that compresses the data, a component that handles the compressed data (or bottleneck), and a decoder (output) portion. When data is fed into an autoencoder, it is encoded and then compressed down to a smaller size. The network is then trained on the encoded/compressed data and it outputs a recreation of that data. The autoencoders reconstruct each dimension of the input by passing it through the network. It may seem trivial to use a neural network for the purpose of replicating the input, but during the replication process, the size of the input is reduced into its smaller representation. The middle layers of the neural network have a fewer number of units as compared to that of input or output layers. Therefore, the middle layers hold the reduced representation of the input. The output is reconstructed from this reduced representation of the input.

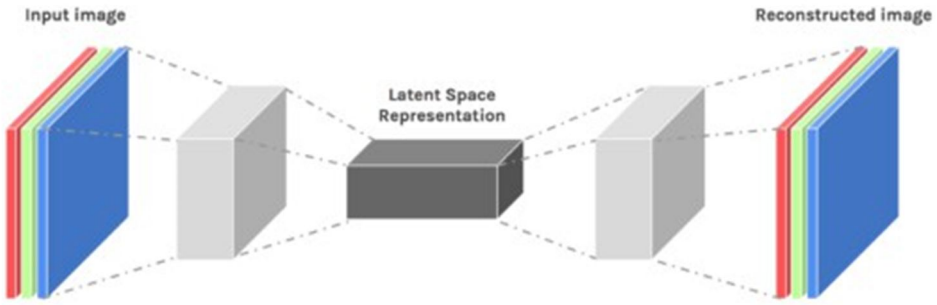


Fig 1. Autoencoder

A. Autoencoder Architecture

An autoencoder can essentially be divided up into three different components:

The encoder, a bottleneck, and the decoder.

The encoder portion of the autoencoder is typically a feedforward, densely connected network. The purpose of the encoding layers is to take the input data and compress it into a latent space representation, generating a new representation of the data that has reduced dimensionality. The code layers, or the bottleneck, deal with the compressed representation of the data. The bottleneck code is carefully designed to determine the most relevant portions of the observed data, or to put that another way the features of the data that are most important for data reconstruction. The goal here is to determine which aspects of the data need to be preserved and which can be discarded. The bottleneck code needs to balance two different considerations: representation size (how compact the representation is) and variable/feature. relevance. The bottleneck performs element-wise activation on the weights and biases of the network. The bottleneck layer is also sometimes called a latent representation or latent variables. The decoder layer is what is responsible for taking the compressed data and converting it back into a representation with the same dimensions as the original, unaltered data. The conversion is done with the latent space representation that was created by the encoder. The most basic architecture of an autoencoder is a feed-forward architecture, with a structure much like a single layer perceptron used in multilayer perceptron's. Much like regular feed-forward neural networks, the auto-encoder is trained using backpropagation.

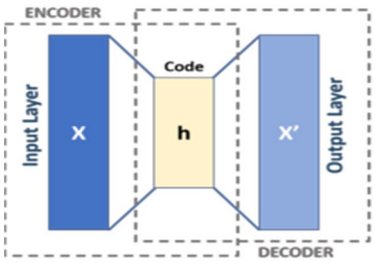


Fig 2. Autoencoder Architecture

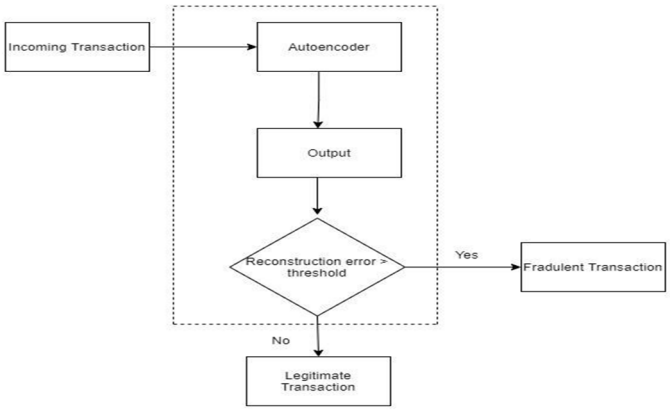


Fig 3. System Architecture

IV. EXPERIMENT ANALYSIS

This project can run on commodity hardware. We ran the entire project on an Intel 8th generation I5 processor with 8 GB Ram,2GB Graphics Card. First part is the training phase which takes 20-25 mins of time and the second part is the testing part which only takes a few seconds to make predictions.

- RAM: 4 GB
- Storage: 500 GB
- CPU: 2 GHz or faster
- Architecture: 32-bit or 64-bit
- Python 3.5 in Google Colab is used for data pre-processing, model training and prediction.
- Operating System: Windows 7 and above or Linux based OS or MAC OS.
- Data Loading
- Class wise Analysis
- Data Modelling
- Model Training
- Model Evaluation
- Web App

Fig 4. Class Wise Analysis

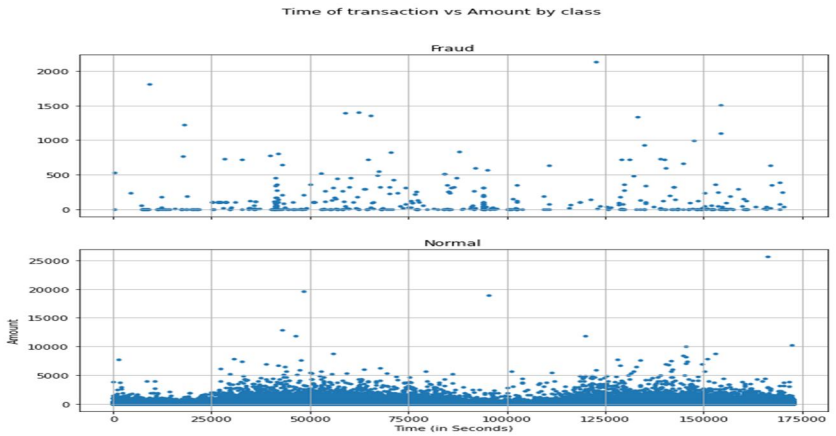


Fig 5. The relation between time of transaction versus amount by fraud and normal class

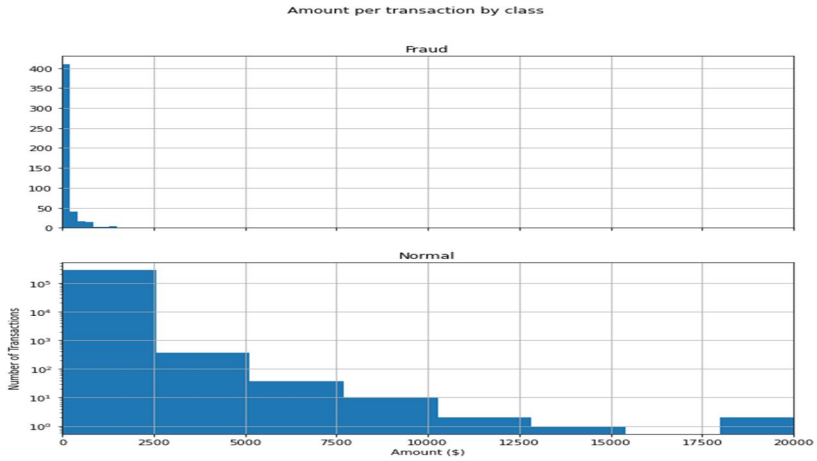


Fig 6. The amount per transaction by fraud and normal class.

A. Data Modelling

Removal of the “Time” attribute since it has no contribution towards the prediction of the class. Division of train and test data in the existing dataset, with 80% training data and 20% testing data.

```
"""###Data Modelling"""
data = df.drop(['Time'], axis =1)
X_train, X_test = train_test_split(data, test_size=0.2, random_state=42) X_train = X_train[X_train.Class == 0]
X_train = X_train.drop(['Class'], axis=1)
y_test = X_test['Class']
X_test = X_test.drop(['Class'], axis=1) X_train = X_train
X_test = X_test print(X_train.shape)
#(227451,29)
print(X_test.shape)
#(56962,29)
print(y_test.shape)
#(56962)
scaler = StandardScaler().fit(X_train.Amount.values.reshape(-1,1))
X_train['Amount'] = scaler.transform(X_train.Amount.values.reshape(-1,1))X_test['Amount'] =
    scaler.transform(X_test.Amount.values.reshape(-1,1)) X_train.shape
#(227451,29)
```

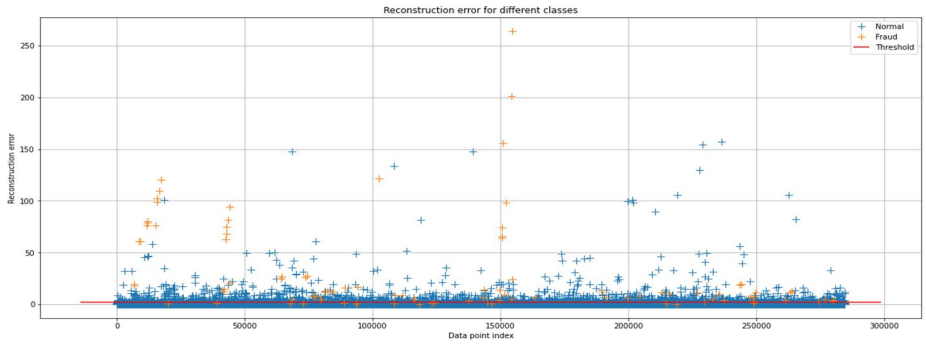
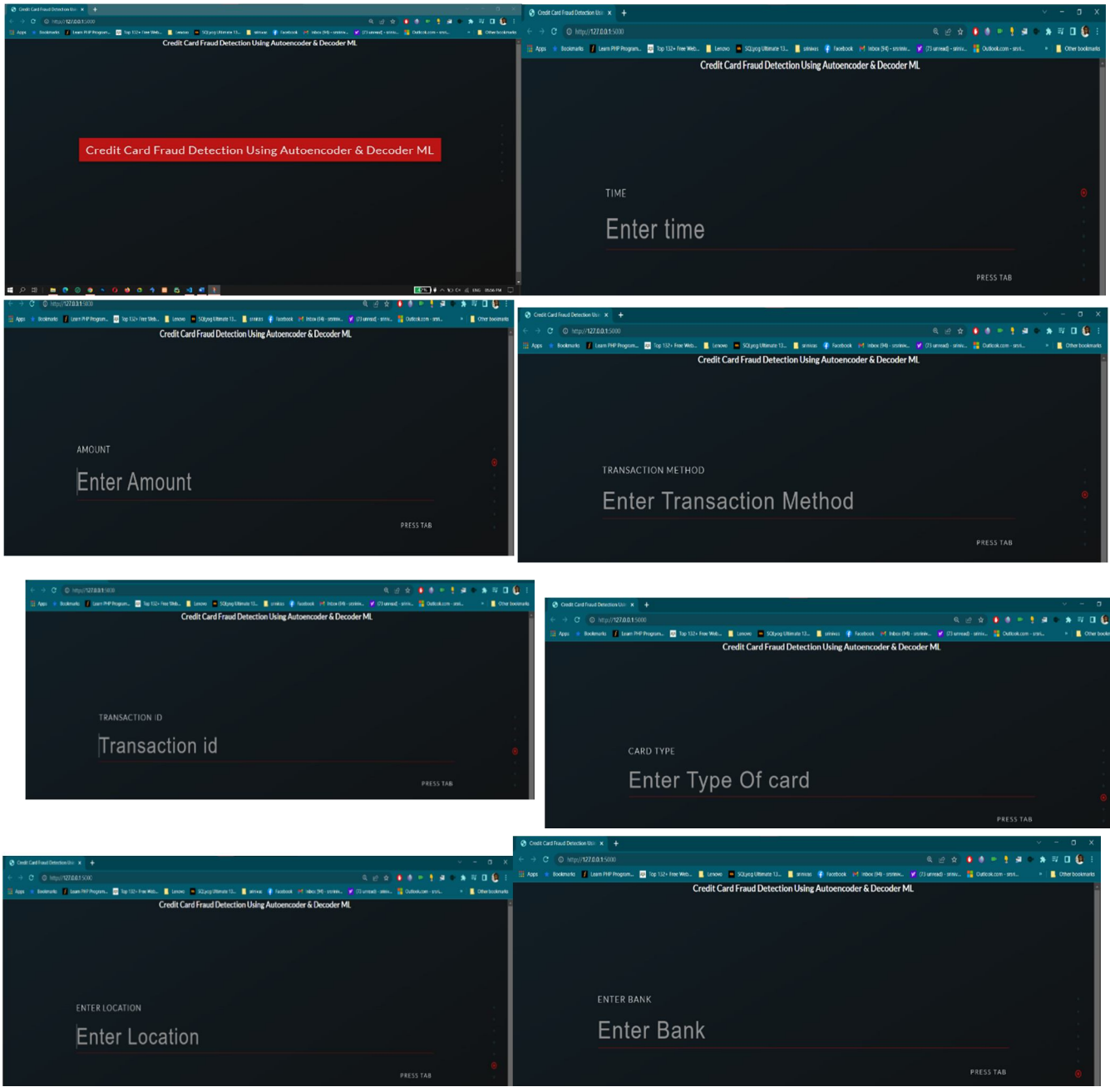
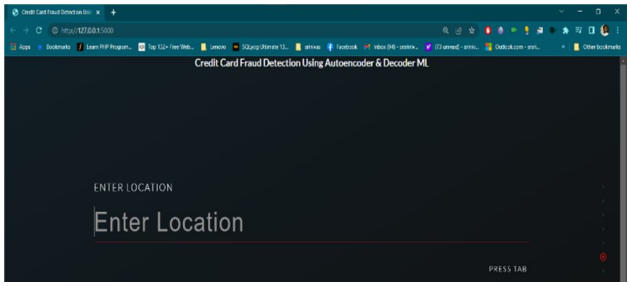
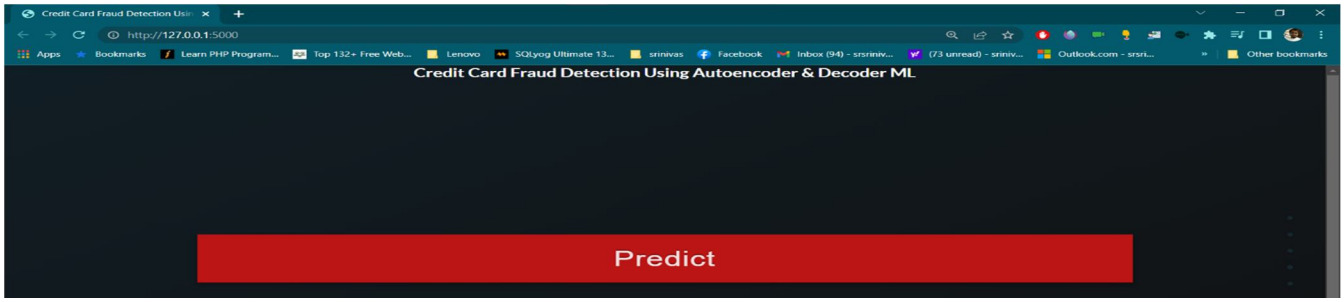


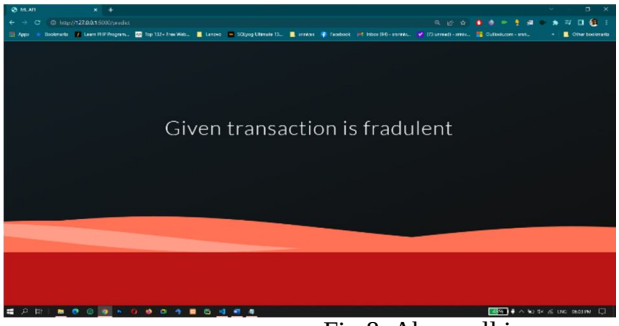
Fig 7. Reconstruction error for different classes

B. Web Pages









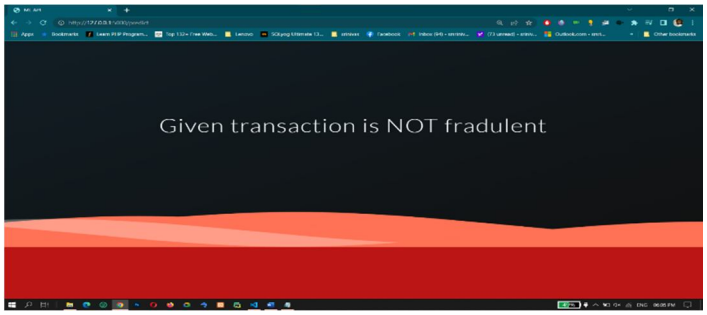


Fig 8. Above all images are step by step procedure web pages

V. CONCLUSIONS & FUTURE SCOPE

In this project we have used an autoencoder to encode the given data and then decode it into original data and then calculated the reconstruction error to classify into normal or fraudulent transactions. We have also saved the trained autoencoder model and then loaded with pickle into the flask application.

Deployment of this model into the cloud applications like Heroku

Extend the model for other datasets

Create an API which takes transactions into it and predicts the type of transaction.

REFERENCES

[1] KhyatiChaudhary, JyotiYadav, BhawnaMallick, " A review of Fraud Detection Techniques: Credit Card", International Journal of Computer Applications Volume 45– No.1 2012.

[2] Michael Edward Edge, Pedro R, Falcone Sampaio, "A survey of signature based methods for financial fraud detection", journal of computers and security, Vol. 28, pp 3 8 1 – 3 9 4, 2009.

[3] Linda Delamaire, Hussein Abdou, John Pointon, "Credit card fraud and detection techniques: a review", Banks and Bank Systems, Volume 4, Issue 2, 2009.

[4] Salvatore J. Stolfo, David W. Fan, Wenke Lee and Andreas L. Prodromidis; "Credit Card Fraud Detection Using Meta-Learning: Issues and Initial Results"; Department of Computer ScienceColumbia University; 1997.

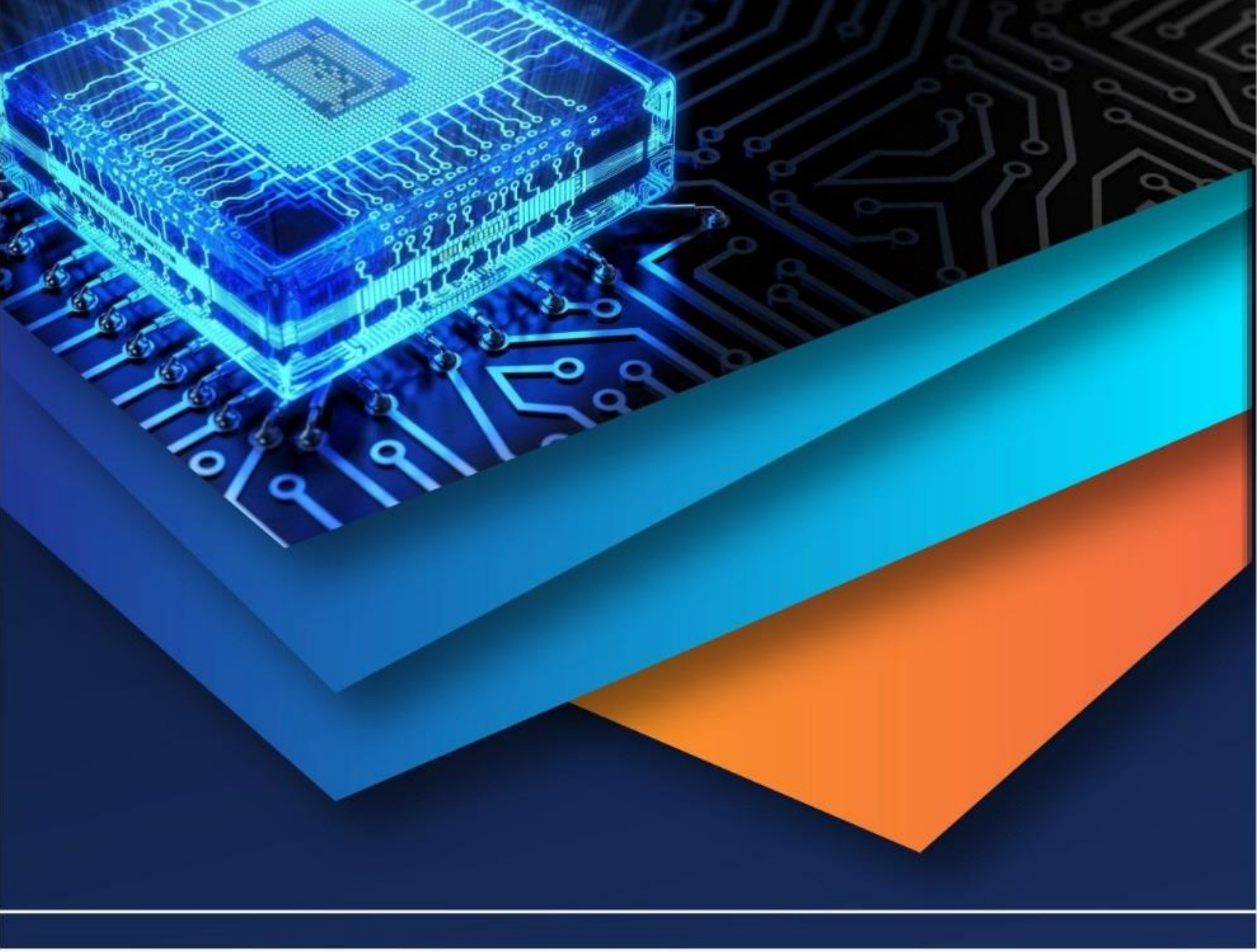
[5] Maes S. Tuyls K. Vanschoenwinkel B. and Manderick B.; "Credit Card Fraud Detection Using Bayesian and Neural Networks"; Vrije University Brussel – Belgium; 2002.

[6] Andreas L. Prodromidis and Salvatore J. Stolfo; "Agent-Based Distributed Learning Applied to Fraud Detection"; Department of Computer Science- Columbia University; 2000.

[7] Salvatore J. Stolfo, Wei Fan, Wenke Lee and Andreas L. Prodromidis; "Cost-based Modeling for Fraud and Intrusion Detection: Results from the JAM Project"; 0-7695-0490-6/99, 1999 IEEE.

[8] Soltani, N., Akbari, M.K., SargolzaeiJavan, M., "A new user-based model for credit card fraud detection based on artificial immune system," Artificial Intelligence and Signal Processing (AISP), 2012 16th CSI International Symposium on., IEEE, pp. 029-033, 2012.

[9] S. Ghosh and D. L. Reilly, “Credit card fraud detection with a neural-network”, Proceedings of the 27th Annual Conference on System Science, Volume 3: Information Systems: DSS/ KnowledgeBased Systems, pages 621-630, 1994. IEEE Computer Society Press.
[10] Masoumeh Zareapoor, Seeja.K.R, M.Afshar.Alam, ”Analysis of Credit Card Fraud Detection Techniques: based on Certain Design Criteria”, International Journal of Computer Applications (0975 – 8887) Volume 52– No.3, 2012.
[11] Holland, J. H. “Adaptation in natural and artificial systems.” Ann Arbor: The University of Michigan Press. (1975).
[12] E. Aleskerov, B. Freisleben, B. Rao, „CARDWATCH: A Neural Network-Based Database Mining System for Credit Card Fraud Detection“, the International Conference on Computational Intelligence for Financial Engineering, pp. 220-226, 1997.
[13] SushmitoGhosh, Douglas L. Reilly, Nestor, “Credit Card Fraud Detection with a NeuralNetwork”, Proceedings of 27th Annual Hawaii International Conference on System Sciences, 1994.
[14] Moody and C. Darken, “Learning with localized receptive fields.” in Proc. of the 1988 Connectionist Models Summer School, D.S. Touretzky, G.E. Hinton and T.J. Sejnowski, eds., Morgan Kaufmann Publishers, San Mateo, CA, 1989, pp. 133-143.
[15] S.J. Nowlan, “Max likelihood competition in RBP networks,” Technical Report CRG-TR-90- 2, Dept. of Computer Science, University of Toronto, Canada, 1990. 22
[16] Krenker, M. Volk, U. Sedlar, J. Bester, A. Kosh, "Bidirectional Artificial Neural Networks for Mobile-Phone Fraud Detection,” Journal of Artificial Neural Networks, Vol.31, No. 1, pp. 92-98, 2009.
[17] MubeenaSyeda, Yan-Qing Zbang and Yi Pan,” Parallel granular neural networks for fast credit card fraud detection”, international conference on e-commerce application, 2002.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY