National College of Ireland

Credit Card Fraud Detection: A Hybrid Approach

MSc Research Project
Programme Name

Damilare Kolawole
Student ID: X21235571

School of Computing
National College of Ireland

Supervisor: Vikas Tomer

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Damilare Abel Kolawole		
Student ID:	X21235571		
Programme:	Data Analytics & Programming	Year:	2023
Module:	Msc Research Project		
Supervisor:	Vikas Tomer		
Submission Due Date:	31-01-2024		
Project Title:	Credit Card Fraud Detection: A Hybrid Approach		
Word Count:	5751	**Page Count**	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: DAMILARE KOLAWOLE

Date: 31-01-2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Credit Card Fraud Detection: A Hybrid Approach

Damilare Abel Kolawole
X21235571

Abstract

The advancement of technology and the widespread use of online transactions have had a tremendous influence on the financial system, resulting in an increase in credit card-related fraud. This research looks at the effectiveness of a Hybrid Deep Learning Approach, especially an Autoencoder-Long Short-Term Memory (LSTM) model, in dealing with the problem of unbalanced datasets in credit card transactions. The study delves into two critical questions: first, how to effectively train a deep learning model on imbalanced datasets where legitimate transactions far outnumber fraudulent ones, thereby benefiting financial institutions, businesses, and cardholders; Second, it compares the proposed Hybrid Deep Learning Approach to current models in credit card fraud detection, with the goal of improving detection systems for different stakeholders. The research focuses on the unbalanced nature of credit card transaction datasets by using the Synthetic Minority Over-sampling Technique (SMOTE) for dataset balancing and feature selection. The hybrid deep Learning Approach incorporates an autoencoder to compress and extract key features, followed by an LSTM model to capture temporal relationships and sequential patterns in the data. This method improves anomaly detection by successfully discriminating irregular sequences. The results show that the hybrid model outperformed current methods in credit card fraud detection. The use of autoencoder-LSTM architecture allows the model to recognize abnormalities with greater precision and accuracy. Furthermore, visual representations such as ROC curves and confusion matrices demonstrate the model resilience, with higher Area Under the Curve (AUC) ratings.

Keywords: Credit Card Fraud, Machine Learning, Deep Learning, Class Imbalance, Detection, SMOTE.

1 Introduction

The rapid growth of technology in recent years has profoundly impacted several businesses, most notably the financial industry. This transition is seen in the rise of Bitcoin, IoT (Internet of Things), and other decentralized digital currencies, which are progressively posing a threat to traditional financial institutions. Despite this digital revolution, the transition to online commerce has resulted in an increase in fraudulent activities, notably card-related fraud. According to the Federal Trade Commission (FTC), consumer fraud losses were more than $8.8 billion, a 30 percent rise over the previous year (Jay, 2023). Due to the proliferation of e-commerce, internet technology, and mobile devices, the widespread usage of credit cards in online purchases has become the standard. However, the frightening rise in credit card theft cases has spurred academic researchers to dive further into this topic. According to the Global

Fraud and Payment Report, nearly 34% of all card transactions in 2022 will be fraudulent. Credit card fraud continues to create significant losses for both individuals and companies. To tackle this, academics have investigated several machine-learning algorithms with the goal of improving credit card fraud detection systems as a viable solution to this continuing issue (Varmedja *et al.*, 2019). The incorporation of outlier identification methods helps improve fraud detection models. The effectiveness of fraud detection systems may be considerably increased by applying these algorithms and encouraging cooperation across varied sectors. Improving real-time credit card fraud detection models seems to be a potential strategy to addressing fraud detection challenges.Click here to enter text. (Pitsane, Hope and T Janse van, 2022).

1.1 Classification of Credit Card Fraud

Credit cards are classified into numerous types, three of which will be explained below:

- Application Fraud.
- Lost/Stolen Card
- Merchant Collision

- Application fraud is a fraudulent activity in which a cardholder obtains a new card from a financial institution or card issuer by use of faked or stolen personal data. This sort of fraudulent activity may manifest in two situations: duplicate fraud, which occurs when a user submits an incorrect set of facts, and identity theft, which involves the use of another person's identifying information (referred to as identity fraud). In both cases, fraudulent methods were used to obtain an illegal credit card.

- Lost/Stolen Card fraud refers to circumstances in which an unauthorized individual uses a lost or stolen credit card to conduct fraudulent transactions.

- Merchant Collision occurs when an individual is inadvertently charged numerous times by a merchant for a single transaction. This circumstance has the potential to result in financial losses for both the company and the client because of the repeated charges.

To identify and classify credit card fraud, researchers have improved machine learning algorithms; but, as technology advances, the detection system will need ongoing improvements. Neural network-based deep learning is a growing area in machine learning. The ability of deep neural networks (DNNs) to identify card fraud at a level that is equivalent to human performance is becoming more widely acknowledged. Credit card transaction analysts work in a dynamic environment where clients buying habits are always changing. As these developments take place, fraudsters are always coming up with new strategies . (Habibpour *et al.*, 2023).

1.2 Justification

This study is motivated by the urgent need to address the complexity and dynamic nature of credit card theft. Traditional approaches could struggle to adapt to evolving strategies, resulting in significant financial losses. The hybrid approach uses deep learning algorithms, such autoencoders and reinforcement learning. Through sequential analysis and the acquisition of representations, this combination allows the system to understand complex and ever-changing fraud patterns, allowing it to detect new fraudulent behaviors and provide accurate detections automatically. As a result, the hybrid approach provides a clever and flexible way to improve fraud detection effectiveness, reduce false positives, and eventually reduce monetary losses suffered by people and companies.

1.3 Research Question

RQ1: How can a Hybrid Deep Learning Approach be effectively trained to address imbalanced datasets in credit card transactions, where legitimate transactions outnumber fraudulent ones, to improve Credit Card Fraud Identification and Detection, thereby benefiting financial institutions, businesses, and cardholders?

RQ2: What is the evaluation of the proposed Hybrid deep learning approach, compared to the existing ones in credit card fraud detection hence improving detection systems for financial institutions, merchants, businesses, and cardholders?

1.4 Research Structure

This research is organized as follows: The second section discusses existing credit card fraud detection literature. The third section presents a research methodology to answer the research question with a detailed explanation of each step. The fourth section is the design specification, the five section is discuss the implementation while the sixth section talks about the experiments done and finally the conclusion.

2 Related Work

Credit card fraud detection is still a major worldwide problem for both consumers and financial institutions. Conventional rule-based systems and machine learning algorithms struggle to keep up with the ever-evolving fraudulent methods. A subtype of machine learning called deep learning, is based on artificial neural networks, has become more popular for handling difficult problems like fraud detection. However, for a variety of reasons, academic researchers are particularly interested in the nuances of credit card fraud identification. Interestingly, there is a huge bias in credit card fraud datasets, with a large proportion of valid transactions over fraudulent ones. Because of this skewed distribution, standard classifiers have difficulty correctly identifying instances of minority classes (Hlosta *et al.*, 2013). Researchers have found certain typical issues with credit card fraud, which will be discussed below.

2.1 Machine Learning for Fraud Detection

(Saheed *et al.*, 2020) The impact of credit card theft on consumers and financial institutions has been growing. To improve detection accuracy, the author uses a Genetic Algorithm (GA) as a feature selection approach to focus the identification of credit card fraud at the application level. However, there is still room for development in terms of assessing and improving the most advanced fraud detection technology (Liou *et al.*, 2018) In order to resolve class imbalance, the author investigates unbalanced data classification and highlights the use of oversampling techniques. The limits of popular oversampling techniques, particularly their effect on introducing noise into artificial minority class data, are not thoroughly examined in this work, however. Clear understanding of the advantages and disadvantages of the suggested method would be possible via a more in-depth comparison with well-established oversampling techniques such as SMOTE, ADASYN, and ensemble approaches.
Effective transaction data analysis is essential to preventing credit card fraud, but it is often hampered by dataset imbalance or skewness. A major problem in machine learning is imbalanced data, which affects model performance. The SMOTE Technique and the unsupervised machine learning technique CT-GAN (Conditional Generative Adversarial Network) are the two methods the author uses in this research to address dataset skewness. Three classifier models are used: Random Forest, MultiLayer Perceptron, and Isolation Forest. The performance measure for both approaches is AUPRC (Area Under the Precision-Recall Curve). The results show that the CT-GAN approach performs better than two of the three models, showing potential for handling problems with unbalanced data. Furthermore, 86 percent of credit card fraud detections are made using the Isolation Forest model, which makes it stand out (Duggal, 2022). A study on deep neural network on credit card fraud detection for tackling uncertainties was done by (Habibpour *et al.*, 2023), The author offers three uncertainty quantification (UQ) strategies for card fraud detection using transaction data, including Monte Carlo dropout, ensemble, and ensemble Monte Carlo dropout. To analyze the prediction uncertainty estimations, the research applies a UQ confusion matrix and many performance criteria. The experimental results show that the ensemble approach is very successful at capturing uncertainty associated with produced forecasts. Furthermore, the suggested UQ approaches provide useful insights into point forecasts, improving the whole fraud prevention process. (Bandr, 2023) explores the benefits and drawbacks of existing Deep Neural Network (DNN)-based fraud detection techniques, evaluating how well they can handle inconsistent data and sequential patterns. It also looks at how important attention techniques are for improving model performance and spotting important fraudulent transactions, including LSTM-attention. The use of forensic techniques into the identification of credit card fraud is an interesting feature. The study examines how current forensic procedures conform to or deviate from the suggested LSTM-attention methodology, emphasizing the model's practicability and suitability for use in actual forensic situations.
Carcillo et al. (Islam *et al.*, 2023) To improve the effectiveness of the fraud detection system, a hybrid model was constructed by integrating supervised and unsupervised approaches. The authors used genuine and annotated datasets of false identification to test their approach. The limitation of this study is that the problem of data imbalance was not addressed.

2.2 Class Imbalance

Researchers have investigated strategies such as sampling and optimization to decrease class imbalance, acknowledging the difficulty of classifying genuine credit card transactions as fraudulent. These researcher try to solve the imbalance problem and improve classification algorithms in fraud detection. (Ullastres, 2022a). (Thabtah *et al.*, 2020) In order to solve class imbalance in fraud detection, the research examined several approaches and carried out a thorough study of the problem. They looked at methods such as thresholding, cost-sensitive learning, undersampling, oversampling, and SMOTE. They attempted to identify the advantages and disadvantages of various approaches via a comparative analysis. The research also sought to determine the effect of dataset imbalance on classifier accuracy. In order to do this, they applied the Naive Bayes technique to datasets with varying levels of skewness and then examined the results.

(Patil, 2021) The authors provide a unique strategy that combines supervised machine learning algorithms including Logistic Regression, Random Forest, and XGBoost with Conditional Tabular Generative Adversarial Networks to solve the class imbalance via data augmentation (CT-GAN). SelectKBest is a feature selection strategy used to identify the most important features to further explore. Machine learning methods trained on both imbalanced and balanced datasets are used to evaluate the suggested approach. Following implementation of the suggested method, the Random Forest model excels. (Deshan *et al.*, 2021) conducted extensive analysis of the European dataset to identify credit card fraud. To overcome the data's fundamental class imbalance, they adopted a stratified splitting technique to guarantee a representative distribution of classes in both the training and testing sets. They employed SMOTE (Synthetic Minority Over-sampling Technique) to mitigate the impact of class imbalance during model training, mainly on the training set. SMOTE is a well-known and regularly used sampling method that employs interpolation to produce synthetic examples of the minority class. By supplementing the data with synthetic samples, SMOTE is able to provide a more evenly distributed training dataset for the models.

Table 1: Review Summary on Credit Card Fraud Detection

Author	Model	Transaction Data	Metrics Used	Results	Limitation
(Fanai and Abbasimehr, 2023)	Deep Autoencoders	European Cardholder dataset, German Credit Dataset	AUC-PR Precision F1 score AUC-ROC	56% 68% 62% 72%	Class imbalance was not addressed in this study which could lead to model instability
(Ullastres, 2022b)	Ensembling Learning	Simulated Credit Card Transactions generated using Sparkov	AUC-PR MCC F1 score	73% 71% 70%	The author focuses on tree-based ensemble classifiers and did not address the issue of Class imbalance.
(Zhang *et al.*, 2021)	Homogeneity-oriented behavior analysis (HOBA)	Real-life dataset	Accuracy F1 score Precision	75% 47% 35.24% 71.68%	Class Imbalance in a real-life dataset should have been addressed properly
(Chalwadi, 2021)	Neural Network MLP	European credit card transaction data	Accuracy Precision Recall F1-Score	99.75% 95.91% 50.81% 66.43%	The researcher did not address how class imbalance influences the training of the Neural Network MLP classifier and whether this affects the model capacity to

					effectively identify fraudulent transactions.
(Misra *et al.*, 2020)	Autoencoders	European Dataset	Accuracy Precision Recall F1-Score	99% 85% 80% 82%	The author did not address class imbalance
(Fiore *et al.*, 2019)	GAN	Simulated Data	Accuracy F1 score Precision	99% 81% 94%	Class imbalance was not addressed properly
(Zhang and Trubey, 2019)	SVM and RF	US transaction data	Adjusted R^2	49%	the researcher does not go into detail on how they dealt with data quality problems or preprocessing stages, which have a substantial impact on model outputs.

3 Research Methodology

Credit card theft has changed dramatically over the years, giving fraudsters more tools with which to carry out breaches, sometimes without the cardholders knowledge. The unlawful charges and significant financial losses associated with these illicit activities are often missed until cardholders get their billing statements. To prevent such illicit operations, strong fraud detection systems and ongoing monitoring are essential, since the complexities of credit card theft is growing. This study will use the Knowledge Discovery in Databases (KDD) approach to solve these issues.

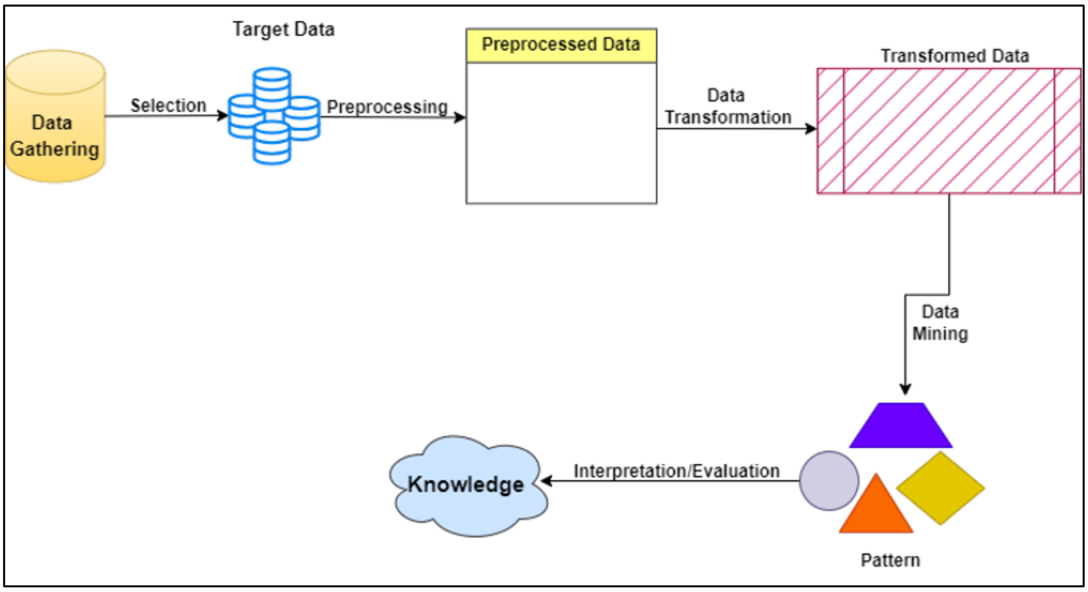

Figure 1: KDD Design Flow

The practice of extracting meaningful and previously undiscovered information, patterns, or insights from massive and complicated datasets is known as KDD technique. The iterative nature of the Knowledge Discovery in Databases (KDD) process allows for data integration, refinement of mined data, improvement of assessment criteria, and data transformation, resulting in a wide variety of relevant outputs. KDD has also advocated for the use of data analytics tools and processes that allow for the discovery of patterns and connections in data.('olaitanvictoriaolanlokun.pdf', no date).

3.1 Understanding of the Research Problem

This research explores the widespread problem of credit card theft and looks at the significant effects it has on both people and companies. Our research question is formed on the basis of literature done in this study. Our goal is to develop a hybrid model that can identify fraudulent transactions by using transaction.

3.2 Data Collection and Exploration

This section collects data and performs an exploratory data analysis on it. In this study, a dataset provided by Kaggle. To protect the privacy of people, the Kaggle repository makes data that is publicly available and anonymizes any personal information that may be exposed. Credit card transactions that took place over the course of two days in September 2013 are included in this repository of data. Using principal component analysis (PCA) for dimensionality reduction as well as ensuring secrecy, it consists of thirty characteristics, with twenty-eight of them being coded as V1 to V28. The remaining characteristics are continuous, with the exception of Amount and Time. While the 'Amount' feature displays the total amount of a credit or debit transaction, the 'Time' feature indicates the amount of time that has passed between the last three transactions. The dataset is huge, with 284807 records and 30 distinct attributes.

3.3 Data Preprocessing

When dealing with datasets that are inconsistent, missing, or noisy, databases often struggle because of their enormous size, which often surpasses several terabytes. Complicating matters further is the fact that such datasets are often obtained from a myriad of sources. Data quality issues are the primary cause of the low quality of mined results.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 284807 entries, 0 to 284806
Data columns (total 33 columns):
 #   Column      Non-Null Count   Dtype
---  ------      --------------   -----
 0   Time        284807 non-null  float64
 1   V1          284807 non-null  float64
 2   V2          284807 non-null  float64
 3   V3          284807 non-null  float64
 4   V4          284807 non-null  float64
 5   V5          284807 non-null  float64
 6   V6          284807 non-null  float64
 7   V7          284807 non-null  float64
 8   V8          284807 non-null  float64
 9   V9          284807 non-null  float64
 10  V10         284807 non-null  float64
 11  V11         284807 non-null  float64
 12  V12         284807 non-null  float64
 13  V13         284807 non-null  float64
 14  V14         284807 non-null  float64
 15  V15         284807 non-null  float64
 16  V16         284807 non-null  float64
 17  V17         284807 non-null  float64
 18  V18         284807 non-null  float64
 19  V19         284807 non-null  float64
 20  V20         284807 non-null  float64
 21  V21         284807 non-null  float64
 22  V22         284807 non-null  float64
 23  V23         284807 non-null  float64
 24  V24         284807 non-null  float64
 25  V25         284807 non-null  float64
 26  V26         284807 non-null  float64
 27  V27         284807 non-null  float64
 28  V28         284807 non-null  float64
 29  Amount      284807 non-null  float64
 30  Class       284807 non-null  int64
 31  Time_min    284807 non-null  int64
 32  Time_hour   284807 non-null  int64
dtypes: float64(30), int64(3)
memory usage: 71.7 MB
```

books/Desktop/Dami Project NCI/New files /MSC Thesis pr.ipynb 5/27

MSC Thesis pr - Jupyter Notebook

671:

	Time	V1	V2	V3	V4	
count	284807.000000	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848
mean	94813.859575	1.759061e-12	-8.251130e-13	-9.654937e-13	8.321385e-13	1.649
std	47488.145955	1.958696e+00	1.651309e+00	1.516255e+00	1.415869e+00	1.380
min	0.000000	-5.640751e+01	-7.271573e+01	-4.832559e+01	-5.683171e+00	-1.137
25%	54201.500000	-9.203734e-01	-5.985499e-01	-8.903648e-01	-8.486401e-01	-6.915
50%	84692.000000	1.810880e-02	6.548556e-02	1.798463e-01	-1.984653e-02	-5.433
75%	139320.500000	1.315642e+00	8.037239e-01	1.027196e+00	7.433413e-01	6.119
max	172792.000000	2.454930e+00	2.205773e+01	9.382558e+00	1.687534e+01	3.480

8 rows × 33 columns

Figure 2: Data Structure

3.3.1 Class Imbalance Handling

Class imbalance is one major problem of credit card dataset as the number of normal transactions is more than that of the fraudulent ones. This is addressed using Synthetic Minority Over-sampling Technique (SMOTE). SMOTE is a method used to handle class imbalances in datasets when one class in the dataset outweighs the other. The dataset used for this research work is highly imbalanced SMOTE will be used to address this issue (Chawla *et al.*, 2002).

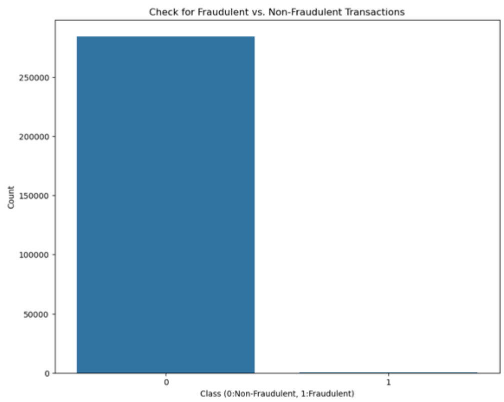

Figure 3: Normal vs Fraud

3.4 Modelling

Applications of several reinforcement models to pre-processed data are required at this crucial step of the KDD process. An innovative strategy that tackles the limitations of conventional fraud detection will be developed via the use of recurrent neural networks (such as LSTM or GRU) and autoencoders, which were suggested in the literature before. This research will primarily make use of these two techniques. The hybrid model, the training and optimization, the evaluation, and the selection of the model (RNN and autoencoders) are the four components that make up this stage. Hybrid Deep Learning Approach (autoencoder and LSTM) selection was made based on the strength of this machine learning models and it performance from literature. While Random Forest is a powerful algorithm, the decision to opt for a hybrid deep learning approach was driven by the need to explore advance techniques that can address the evolving nature of credit card fraud.

The uniqueness of the project lies in the adoption of the hybrid deep learning approach. (the autoencoder and LSTM). The aim is to capitalize on the strengths of autoencoders for feature extraction and long short-term memory for deep sequential learning to capture temporal dependencies. This integration allows our model to discern and intricate patterns and anomalies in credit card transactions. Additionally, the incorporation of the Synthetic Minority Over-Sampling Technique (SMOTE) to address the challenge of class imbalance in our dataset. Hence, this project leveraged on the strengths of deep learning and preprocessing techniques. The model was trained using all relevant features after a careful consideration of the feature importance it was observed that v15, v17, v24,v27 have the same distribution on fraudulent and real transactions so it is of no importance for model training and they were dropped. All other features were used from v1 – v26 asides the aforementioned (v15, v17, v24,v27).

3.4.1 Recurrent Neural Networks

Credit card transaction is seemingly sequential, one major strength of RNN is its ability to capture temporal dependencies. RNN can detect complex patterns by using the data sequence structure as this would be used to detect anomalies. However, because of the vanishing or expanding gradient issue, RNNs have had difficulty capturing long-term relationships. Their inability to store and apply knowledge over long sequences was hampered by this constraint. As a result, more complex RNN architectures, including Gated Recurrent Unit and Long Short Term Memory (LSTM), evolved (GRU).

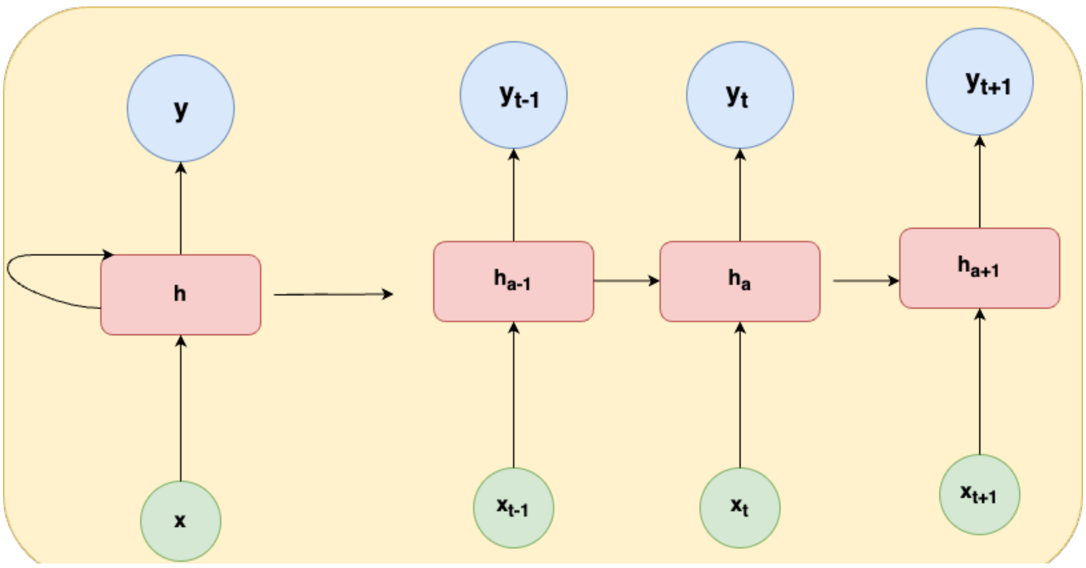

Figure 4: RNN Architecture

3.4.2Autoencoders

Autoencoders: are employed to learn meaningful representations of the input data in an unsupervised manner. By using autoencoders, the input data is encoded into a compressed form known as the latent space or bottleneck. The encoded representation preserves the most prominent characteristics of the original data, removing extraneous information and noise while retaining important patterns and structures (Baldi, 2012).

- **Encoder:** the input data xa is mapped into hidden form h by the encoder. If w_t and b_t represents the layer biases and weight then the hidden form can be expressed $h = f_E\ (w_t * x_i + b_t)$
- **Decoder:** transforms h of the reconstruction y` of the original data If $w_t$ $and$ $b_t$ represents the layer biases and weight then the hidden form can be expressed $y` = f_D\ (w_t * h + b_t)$

  Where: f_D and F_E $are\ decoder\ and\ encoder\ functions\ respectively$

  $w_t and\ b_t\ are\ weight\ and\ biases\ repectively$

The figure below is the architectural design of autoencoders.

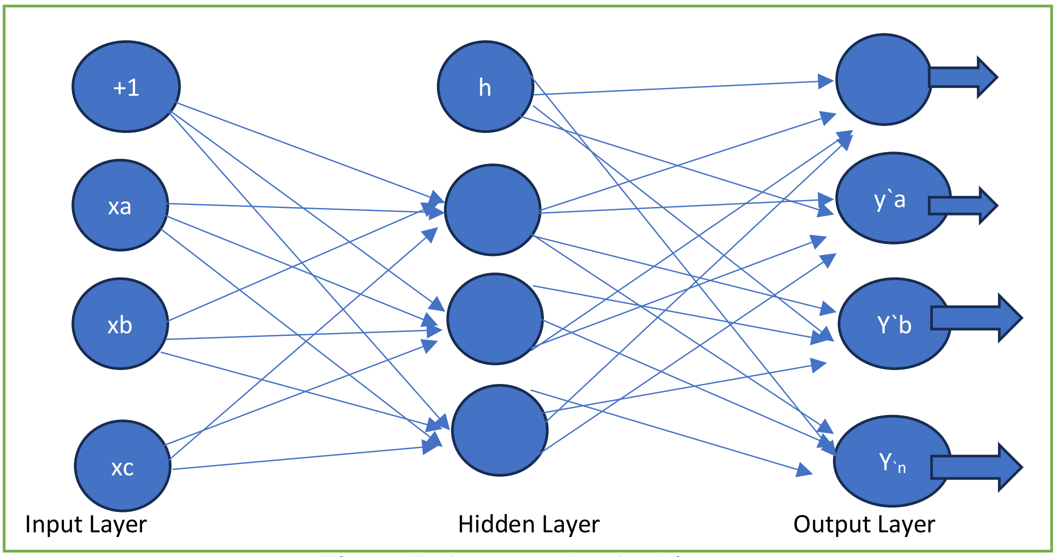

Figure 5: Autoencoder Architecture

3.4.2 Autoencoders and LSTM (Hybrid Model)

By honing the basic cell structure of Recurrent Neural Networks, Long Short-Term Memory (LSTM) has achieved remarkable success in a variety of areas, including music creation, picture captioning, voice recognition, and language translation (RNNs). Using memory components such as forget gates, input gates, and output gates, it addresses the problem of disappearing gradients in RNNs. By using these memory units, the model can efficiently manage data sequences by either retaining or discarding information.

For accurate predictions in detecting credit card fraud, it is essential to think about the transactional behaviour from beginning to end. But there would be computational inefficiencies due to a dramatic increase in data dimensionality if all transactional data were explicitly included in the prediction model. Using Autoencoders is a good way to tackle this problem. To reduce dimensionality without sacrificing model performance, Autoencoders help extract key features from upstream and downstream transactional data. By including these properties into the LSTM model's input structure, we may reduce the data dimensionality and let the model understand the impact of previous and upcoming transactions.

4 Design Specification

The figure below depicts the architectural design for our proposed work, the process flow that outlines each procedure taken. The dataset used in this study is gotten from Kaggle repository. An exploratory data analysis was done to understand the data better, after which null and duplicates was taken off, SMOTE was then used to address class imbalance and feature selection was used to extract the required features. Finally, for model training, the data was divided into train and test for model training.

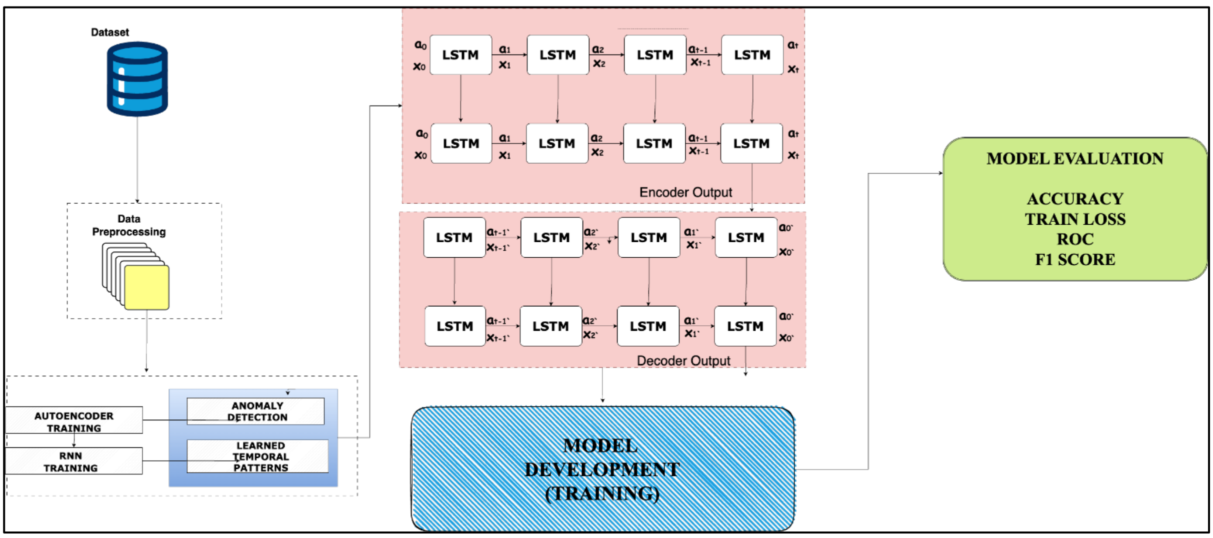

Figure 6: Design Flow for the Proposed Model

Algorithm: Workflow for the Proposed model

Input:

- $Training\ set\ x_{train}\ y_{train}\ (features\ and\ labels)$
- Test set $Test\ Set\ x_{test}\ (features)$

Output:

- $Predicted\ labels\ for\ test\ set\ x\_test$

1 $Preprocess\ the\ training\ data\ (normalize, scale, handle\ missing\ values, etc.)$
2 $use\ x_{train} to\ build\ the\ autoencoder\ training\ set\ x\_AE.$

initialize *the weight matrices of the autoencoder randomly*
put x_{AE} *into autoencoder.*
Train the autoencoder to reconstruct the input
Generate features (latent representations) z_{train} *from the autoencoder for* x_{train}
initialize *LSTM network*
Use z_{train} *and* y_{train} *to train the LSTM network*
for *each test sample* x_{test_i} ***do***
Encode x_{test_i} *using the trained autoencoder to get* z_{test_i}
use z_{test_i} *as input to the trained LSTM network*
predict the label for x_{test_i} *using the LSTM network*
end for
return the predicted labels for x_{test}

.

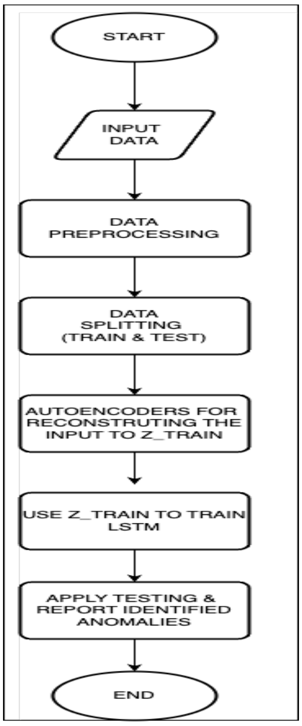

Figure 7: Flowchart For Proposed Model

5 Implementation

5.1.1 Software and Technologies Used

The software used for this research work are:

- **Programming Language used:** Python
- **IDE:** Anaconder (Jupyter Notebook)
- **Python Libraries:**

- Pandas is an analytical and data manipulation library. It provides the data structures and operations required for effectively cleaning, preprocessing, and analyzing data, especially structured data.
- NumPy is a Python library for numerical calculations. It supports arrays, matrices, and mathematical functions for performing a variety of operations on numerical data.
- Scikit-learn has a number of tools for preparing data, training models, evaluating models, and more. To divide datasets into training and testing sets, use the train test split function.
- Imbalanced-learn is a method for oversampling, SMOTE, undersampling, and other methods to manage class imbalance in classification issues.
- Google TensorFlow is an open-source machine learning framework. Keras is a TensorFlow API that is used to construct and train neural network models.
- Seaborn is a data visualization toolkit. It is developed on top of Matplotlib and adds capabilities and improves the aesthetics of visualizations.

The figure below gives depicts where SMOTE, and other libraries were imported.

```
[1]: import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from imblearn.over_sampling import RandomOverSampler
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input, LSTM, Dense
import pandas as pd
import numpy as np
from scipy import stats
import tensorflow as tf
import matplotlib.pyplot as plt
import seaborn as sns
import pickle
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, precision_recall_curve
from sklearn.metrics import recall_score, classification_report, auc, roc_curve
from sklearn.metrics import precision_recall_fscore_support, f1_score
from sklearn.preprocessing import StandardScaler
from pylab import rcParams
from keras.models import Model, load_model
from keras.layers import Input, Dense
from keras.callbacks import ModelCheckpoint, TensorBoard
from keras import regularizers

import warnings
warnings.filterwarnings('ignore')

print('Imported successfully')
```

Figure 8: Libraries Imported

```
In [9]: # transform the dataset
from imblearn.over_sampling import SMOTE
oversample = SMOTE()
X_r, y = oversample.fit_resample(X, tr_data['Class'])
# summarize the new class distribution
counter = Counter(y)
print(counter)
# scatter plot of examples by class label
for label, _ in counter.items():
    row_ix = where(y == label)[0]

Counter({0: 284315, 1: 284315})
```

Figure 9: SMOTE

5.1.2 Model Development

This section discusses how the model is developed. The autoencoder architecture is first built and the long-term short memory is then integrated. The features extracted from autoencoder is used as the input data for the LSTM model and then the model is ran for better performance.

```
In [27]: import tensorflow
from tensorflow.keras.layers import Dense,LSTM
from tensorflow.keras.models import Model
from tensorflow.keras import models,layers,activations,losses,opti
from tensorflow.keras.callbacks import EarlyStopping
n_features = len(train_data.columns)
encoder = models.Sequential(name='encoder')
encoder.add(layer=layers.Dense(units=200, activation=activations.r
encoder.add(layers.Dropout(0.1))
encoder.add(layer=layers.Dense(units=100, activation=activations.r
encoder.add(layer=layers.Dense(units=5, activation=activations.rel

decoder = models.Sequential(name='decoder')
decoder.add(layer=layers.Dense(units=100, activation=activations.r
decoder.add(layer=layers.Dense(units=200, activation=activations.r
decoder.add(layers.Dropout(0.1))
decoder.add(layer=layers.Dense(units=n_features, activation=activa

autoencoder = models.Sequential([encoder, decoder])

autoencoder.compile(
    loss=losses.MSE,
    optimizer=optimizers.Adam(),
    metrics=[metrics.mean_squared_error])

WARNING:absl:At this time, the v2.11+ optimizer `tf.keras.optimiz
```

Figure 10: Model Building

Integrating AE+LSTM

```
In [48]: # Extract encoded representations (latent space)
encoded_train_data = autoencoder.predict(x_train)
encoded_test_data = autoencoder.predict(x_test)

7121/7121 [==============================] - 3s 360us/step
1781/1781 [==============================] - 1s 318us/step
```

lhost:8888/notebooks/Desktop/Dami Project NCI/New files /MSC Thesis pr.ipynb 20/27

2/2023, 15:16 MSC Thesis pr - Jupyter Notebook

```
In [50]: import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
from tensorflow.keras.callbacks import EarlyStopping

# Reshape data for LSTM (assuming a sequence length of 10)
sequence_length = 10

def create_sequences(data, sequence_length):
    sequences = []
    for i in range(len(data) - sequence_length + 1):
        sequences.append(data[i:i+sequence_length])
    return np.array(sequences)

# Create sequences for LSTM input
train_sequences = create_sequences(encoded_train_data, sequence_le
test_sequences = create_sequences(encoded_test_data, sequence_leng

# LSTM model
lstm_model = Sequential([
    LSTM(units=64, input_shape=(train_sequences.shape[1], train_se
    # Add more LSTM layers or Dense layers if needed
    Dense(units=1, activation='sigmoid')
])

lstm_model.compile(optimizer='adam', loss='binary_crossentropy', m

# Early stopping to prevent overfitting
early_stopping = EarlyStopping(patience=3, restore_best_weights=Tr

# Train LSTM on sequences
lstm_model.fit(train_sequences, y_train[sequence_length-1:], epoch

# Evaluate the model
score = lstm_model.evaluate(test_sequences, y_test[sequence_length
print("Test Accuracy:", score[1])
```

Figure 11: LSTM Integration

The figure above shows the model developed in this research work. Altogether, three models were developed, traditional Autoencoder, LSTM and then Autoencoder and LSTM in this research work.

6 Evaluation

The aim of this section is to evaluate the effectiveness of the proposed model by performing an experiment on a dataset including credit card transaction data. The primary aim is to evaluate the effectiveness of the model in discerning genuine credit card transactions from fraudulent ones. Every algorithm was subjected to an intensive evaluation process to determine its performance.

6.1 Experiment 1: Autoencoders

The first experiment was on autoencoders, the table below shows the results gotten.

Table 2: Autoencoder Results

Precision	Accuracy	F1-score	Recall
0.106	0.987	0.100	0.106

Autoencoder has better accuracy but other evaluation metrics performance are not so well hence the need for an improved model.

6.2 Experiment 2: LSTM

The second experiment was performed using traditional long-short term memory. The results gotten are shown in the table below:

Table 3: LSTM Results

Precision	Accuracy	F1-score	Recall
0.986	0.937	0.934	0.887

LSTM performed more better than autoencoder, it has a lower accuracy but better metrics across other evaluation which makes it better than the first experiment done.

6.3 Experiment 3: Autoencoder and LSTM (Proposed Model)

The last experiment done was using the features extracted from autoencoders to train LSTM. The model aims to leverage on autoencoder ability for anomaly detection and LSTM for temporal dependencies. The results gotten is shown in the table below:

Table 4: AE+LSTM Results

Precision	Accuracy	F1-score	Recall
1.00	0.998	0.991	0.990

The results above show an improved performance across all evaluation metrics which indicates a better model.

6.4 Experiment 4: Comparism with Existing Models

Table 5: Results with Other Models

Author	Model	Transaction Data	Metrics Used	Results
(Fanai and Abbasimehr, 2023)	Deep Autoencoders	European Cardholder dataset, German Credit Dataset	AUC-PR Precision F1 score AUC-ROC	56% 68% 62% 72%
(Ullastres, 2022)	Ensembling Learning	Simulated Credit Card Transactions generated using Sparkov	AUC-PR MCC F1 score	73% 71% 70%
(Zhang *et al.*, 2021)	Homogeneity-oriented behavior analysis (HOBA)	Real-life dataset	Accuracy F1 score Precision	75% 47% 35.24% 71.68%
(Fiore *et al.*, 2019)	GAN	Simulated Data	Accuracy F1 score Precision	99% 81% 94%
(Misra *et al.*, 2020)	Autoencoders	European Dataset	Accuracy Precision Recall F1-Score	99% 85% 80% 82%
Our Model	AE+LSTM	European Dataset	Accuracy Precision Recall F1-Score ROC	99% 99% 99% 93% 87%

6.5 Discussion

To answer the research question 'How can a Hybrid Deep Learning Approach be effectively trained to address imbalanced datasets in credit card transactions, where legitimate transactions outnumber fraudulent ones, to improve Credit Card Fraud Identification and Detection, thereby benefiting financial institutions, businesses, and cardholders?' as the data is highly imbalanced, this issue was carefully looked into using Synthetic Minority Over-sampling Technique (SMOTE) was used, all irrelevant features were also dropped and the model was trained on essential features alone. The second research question was 'What is the evaluation of the proposed Hybrid deep learning approach, compared to the existing ones in credit card fraud detection hence improving detection systems for financial institutions, merchants, businesses, and cardholders?' The above experiment shows that the hybrid model has an improved performance. The integration of autoencoder and long-short term memory creates a better anomaly detection system for credit card fraud. Autoencoders takes the input data into a reduced dimensional latent space, and it preserves essential features while LSTM learns of temporal dependencies, analyse compressed features to capture detailed sequential patterns. This model is better at spotting anomalies by recognizing regular sequences. The figure below shows the ROC curve, and confusion matrix the AUC gotten from the model to further address the evaluated results.

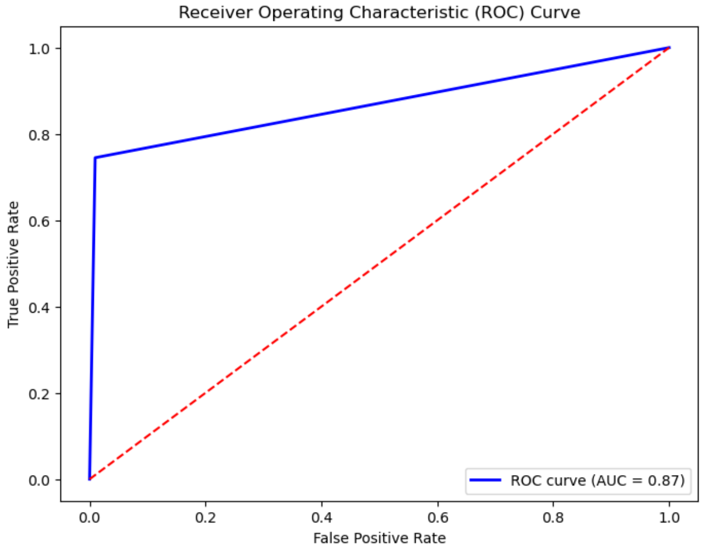

Figure 12: ROC Curve

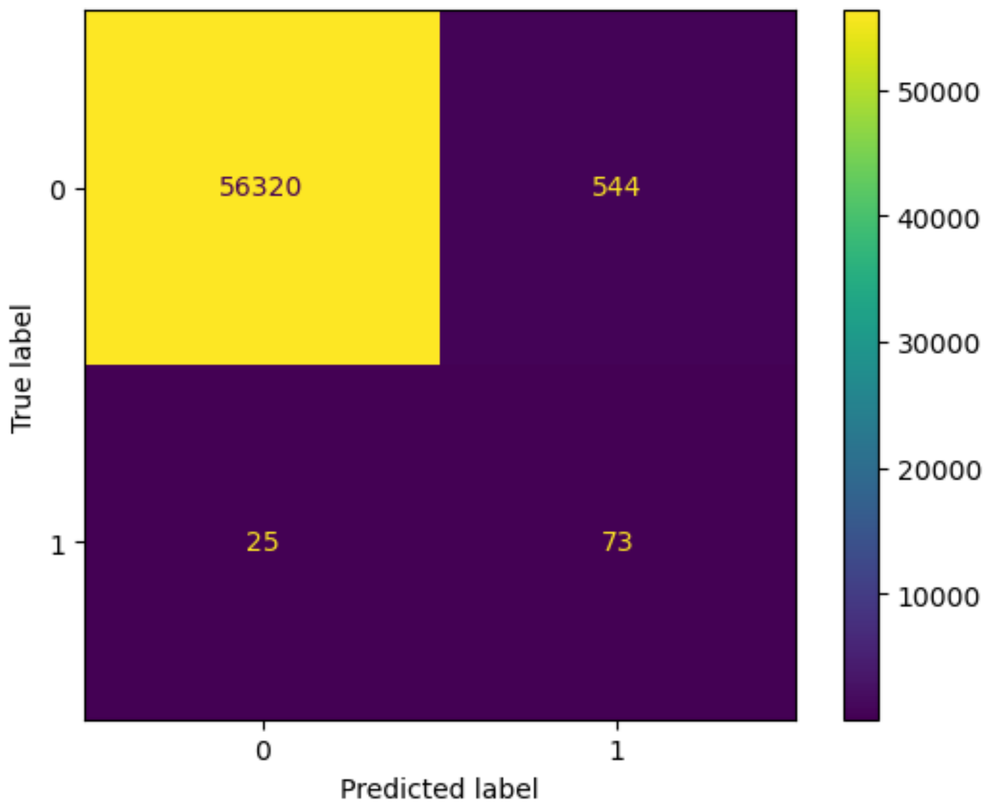

Figure 13: Confusion Matrix

7 Conclusion and Future Work

In conclusion, credit card fraud is a major area that needs to be continuously addressed, because technologies evolve, and people will continue to use digital transaction as comes convenient. Credit card data is highly imbalanced, and this research work aims to improve fraud detection accuracy in situations where legal transaction exceeds fraudulent ones. Various deep learning method such as Autoencoders, LSTM and a hybrid model (AE+LSTM) to predict if a transaction was fraudulent or not.

To improve the performance evaluation class imbalance, feature selection was used to get relevant features needed for model training. Metrics such as confusion matrix and ROC was used to evaluate this model alongside other evaluation metrices, although false positive and negatives gotten was not 0 which financial institutions needs to get when training their models. Future works can be done by adding more layers to the AE-LSTM architecture, applying attention mechanism could also improve the model to get 0 false positive for better fraud prediction. There are other methods that could be used to improve this research work. Models such as:

1. Generative Adversarial Network (GAN) with LSTM: by leveraging on GANs to generate synthetic data for the minority class to address class imbalance.
2. Variational Autoencoder with LSTM: using Variational autoencoder, the model could be enhanced using the probabilistic nature of variational autoencoders to generate better model performance.

3. Ensemble Methods: Random Forest and other classifier could be used by leveraging on the strength of each model for a better fraud detection model performance.
4. Attention Mechanism with LSTM: attention mechanism could assign different weight to various part of input sequence so the model can be trained on the relevant information during the learning process.

References

Baldi, P. (2012) ‘Autoencoders, Unsupervised Learning, And Deep Architectures.’, *Proceedings of the ICML Work shop on Unsupervised and Transfer Learning*, pp. 37-49.

Bandr, F. (2023) ‘FORENSIC CREDIT CARD FRAUD DETECTION USING DEEP NEURAL NETWORK’, *Journal of Southwest Jiaotong University*, 58(1). Available at: https://doi.org/10.35741/issn.0258-2724.58.1.33.

Chalwadi, K.R. (2021) ‘Classification of Credit Card Fraudlent transactions using Neural Network and Oversampling Technique.’

Chawla, N.V. *et al.* (2002) ‘Smote: synthetic minority over-sampling technique.’, *Journal of Artificial Intelligence Research*, (16), pp. 321-357.

Deshan, H. *et al.* (2021) ‘Decision Analysis and Prediction Based on Credit Card Fraud Data’, *The 2nd European Symposium on Computer and Communications*, pp. 20–26. Available at: https://doi.org/10.1145/3478301.3478305.

Duggal, P. (2022) ‘Predicting Credit Card Fraud Using Conditional Generative Adversarial Network’, p. 19.

Fanai, H. and Abbasimehr, H. (2023) ‘A novel combined approach based on deep Autoencoder and deep classifiers for credit card fraud detection’, *Expert Systems with Applications*, 217, p. 119562. Available at: https://doi.org/10.1016/j.eswa.2023.119562.

Fiore, U. *et al.* (2019) ‘Using generative adversarial networks for improving classification effectiveness in credit card fraud detection’, *Information Sciences*, 479, pp. 448–455. Available at: https://doi.org/10.1016/j.ins.2017.12.030.

Habibpour, M. *et al.* (2023) ‘Uncertainty-aware credit card fraud detection using deep learning’, *Engineering Applications of Artificial Intelligence*, 123, p. 106248. Available at: https://doi.org/10.1016/j.engappai.2023.106248.

Hlosta, M. *et al.* (2013) ‘Constrained Classification of Large Imbalanced Data by Logistic Regression and Genetic Algorithm’, *International Journal of Machine Learning and Computing*, pp. 214–218. Available at: https://doi.org/10.7763/IJMLC.2013.V3.305.

Islam, M.A. *et al.* (2023) ‘An ensemble learning approach for anomaly detection in credit card data with imbalanced and overlapped classes’, *Journal of Information Security and Applications*, 78, p. 103618. Available at: https://doi.org/10.1016/j.jisa.2023.103618.

Jay, M. (2023) ‘FTC Report’, *New FTC Data Show Consumers Reported Losing Nearly $8.8 Billion to Scams in 2022*, 20 July. Available at: https://www.ftc.gov/news-events/news/press-releases/2023/02/new-ftc-data-show-consumers-reported-losing-nearly-88-billion-scams-2022 (Accessed: 20 July 2023).

Liou, C.-Y. *et al.* (2018) ‘Autoencoder for words’, *Neurocomputing*, 139, pp. 84–96. Available at: https://doi.org/10.1016/j.neucom.2013.09.055.

Misra, S. *et al.* (2020) ‘An Autoencoder Based Model for Detecting Fraudulent Credit Card Transaction’, *Procedia Computer Science*, 167, pp. 254–262. Available at: https://doi.org/10.1016/j.procs.2020.03.219.

‘olaitanvictoriaolanlokun.pdf’ (no date).

Patil, T. (2021) ‘Credit Card Fraud Detection Using Conditional Tabular Generative Adversarial Networks (CT-GAN) and Supervised Machine Learning Techniques.’, p. 23.

Pitsane, M.Y., Hope, M. and T Janse van, R. (2022) ‘Improving Accuracy of Credit Card Fraud Detection Using Supervised Machine Learning Models and Dimension Reduction’, *31/12/2022* [Preprint]. Available at: https://doi.org/doi.org/10.59200/ICONIC.2022.032.

Saheed, Y.K. *et al.* (2020) ‘Application of GA Feature Selection on Naive Bayes, Random Forest and SVM for Credit Card Fraud Detection’, in *2020 International Conference on Decision Aid Sciences and Application (DASA)*. *2020 International Conference on Decision Aid Sciences and Application (DASA)*, Sakheer, Bahrain: IEEE, pp. 1091–1097. Available at: https://doi.org/10.1109/DASA51403.2020.9317228.

Thabtah, F. *et al.* (2020) ‘Data imbalance in classification: Experimental evaluation’, *Information Sciences*, 513, pp. 429–441. Available at: https://doi.org/10.1016/j.ins.2019.11.004.

Ullastres, E.F. (2022a) ‘Credit Card Fraud Detection using Ensemble Learning Algorithms’, p. 26.

Ullastres, E.F. (2022b) ‘Credit Card Fraud Detection using Ensemble Learning Algorithms’.

Varmedja, D. *et al.* (2019) ‘Credit Card Fraud Detection - Machine Learning methods’, in *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*. *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, East Sarajevo, Bosnia and Herzegovina: IEEE, pp. 1–5. Available at: https://doi.org/10.1109/INFOTEH.2019.8717766.

Zhang, X. *et al.* (2021) ‘HOBA: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture’, *Information Sciences*, 557, pp. 302–316. Available at: https://doi.org/10.1016/j.ins.2019.05.023.

Zhang, Y. and Trubey, P. (2019) ‘Machine Learning and Sampling Scheme: An Empirical Study of Money Laundering Detection’, *Computational Economics*, 54(3), pp. 1043–1063. Available at: https://doi.org/10.1007/s10614-018-9864-z.