COVENTRY

UNIVERSITY

COURSE: M.Sc. COMPUTER SCIENCE

MODULE CODE:7091CEM

STUDENT ID:13143956

SUBMITTED BY

: RAJINDER KAUR

SUBMITTED TO:

: Prof. MARK ELSHAW

**7091CEM**

**MACHINE LEARNING**

**AND BIG DATA**

**Table of Contents**

[Introduction 3](file:///C:\Users\rajin\Downloads\Machine%20Learning%20and%20Big%20Data%20(SUKH)%20(1).docx#_Toc130579589)

[a) Big Data for Machine Learning: 3](file:///C:\Users\rajin\Downloads\Machine%20Learning%20and%20Big%20Data%20(SUKH)%20(1).docx#_Toc130579590)

[b) Architecture: 9](file:///C:\Users\rajin\Downloads\Machine%20Learning%20and%20Big%20Data%20(SUKH)%20(1).docx#_Toc130579591)

[c) Visualization of data: 11](file:///C:\Users\rajin\Downloads\Machine%20Learning%20and%20Big%20Data%20(SUKH)%20(1).docx#_Toc130579592)

[d) Machine Learning System Application: 19](file:///C:\Users\rajin\Downloads\Machine%20Learning%20and%20Big%20Data%20(SUKH)%20(1).docx#_Toc130579593)

[e) Conclusion and Reflection: 21](file:///C:\Users\rajin\Downloads\Machine%20Learning%20and%20Big%20Data%20(SUKH)%20(1).docx#_Toc130579594)

[Reference 22](file:///C:\Users\rajin\Downloads\Machine%20Learning%20and%20Big%20Data%20(SUKH)%20(1).docx#_Toc130579595)

# **Introduction**

This report aims to evaluate datasets, select suitable architectures, visualise the data, implement and evaluate machine learning system applications, and reflect processes. This particular big data-related dataset has been uploaded here properly to perform different kinds of operations. The entire task has been done in the “*Jupyter Notebook*” platform using “python programming language”. There are different types of data visualizations and machine learning approaches used to perform the detection of the fraud in the online payment system. The application of the machine learning system, preprocessing, architecture selection, and data visualization were successful.

# **a) Big Data for Machine Learning:**

***Dataset Uploading***

The development of a machine learning system to identify online payment fraud using a specific data set has been described in this report (Alwadain *et al.* 2023). According to the task, the report is divided into five phases.

Table

Description automatically generated

**Figure 1: Importation of libraries and uploading the dataset**

(Source: Developed in Jupyter Notebook)

A dataset that has been taken from “*Kaggle.com*” regarding “Online Payment Fraud Detection” has been chosen as the dataset for this task and the source of the data is “*https://www.kaggle.com/datasets/rupakroy/online-payments-fraud-detection-dataset?resource=download*”.

***Evaluation of the dataset***

The dataset has also been evaluated through its quality and quantity to perform all the tasks properly and efficiently here. This dataset has described the fraud data of the payment system and the values present in the dataset are appropriate to solve the entire task.

Table

Description automatically generated

**Figure 2: Showing the shape and head of the dataset**

(Source: Developed in Jupyter Notebook)

This picture shows the shape of the fraud dataset which has 6362620 rows and the number of columns present in the dataset is 11 (Chaquet-Ulldemolins *et al.* 2022). The head data has been shown in this picture of the first ten rows.

Graphical user interface, text

Description automatically generated

**Figure 3: Showing the information of the dataset**

(Source: Developed in Jupyter Notebook)

This picture represents the information about the fraud dataset which has 11 columns. The data types of each variable and the index number of each column have also been described here properly (Emmanuel *et al.* 2022). There are three different types of data types present in this fraud dataset such as ‘object’, ‘integer’ and ‘float’.

Graphical user interface, table

Description automatically generated

**Figure 4: Showing the statistical description of the dataset**

(Source: Developed in Jupyter Notebook)

The payment fraud data has been described here properly to calculate the different kinds of operations such as max, min, and standard deviation of each column. The code that has been used for this particular operation is “df\_payment\_fraud.describe()”.

Graphical user interface, text, application

Description automatically generated

**Figure 5: Checking the null values present in the dataset**

(Source: Retrieved from Jupyter Notebook)

The above picture represents whether any null value present in the dataset or not and then it also checks the uniqueness of this particular fraud dataset. There are no null values present in each column of the dataset.

A picture containing text

Description automatically generated

**Figure 6: Checking the duplicate values present in the dataset**

(Source: Retrieved from Jupyter Notebook)

This particular image represents any duplicate value present in the dataset. There is no duplicate values present in the data.

Graphical user interface, text, application

Description automatically generated

**Figure 7: Checking the numbers of values present in the is fraud column**

(Source: Retrieved from Jupyter Notebook)

This particular picture represents the number of values present inside the “is\_fraud” column. There are 6354407 values present for the zero index and 8213 values present in the first index.

***Preprocessing the dataset***

Various characteristics, such as the amount of the transaction, the type of payment, the age of the account, and the status of the fraud, are included in selected records (Fang *et al.* 2019). There are 6,362,620 rows and 11 columns in the dataset.

Text

Description automatically generated

**Figure 8: Dividation of each class in is fraud column**

(Source: Retrieved from Jupyter Notebook)

As per the above picture, each class has been divided to two different parts here to generate the best accuracy score. The dataset was evaluated and found to be suitable for machine learning systems (Han *et al.* 2020). However, preprocessing was required for the dataset. Encoding categorical features, scaling numerical features, and removing missing values were all part of preprocessing. Python's “pandas and scikit-learn” libraries are used for preprocessing.

Table

Description automatically generated

**Figure 9: Showing the prepared dataset after preprocessing**

(Source: Retrieved from Jupyter Notebook)

The above picture shows the prepared dataset that has been named as “prepared\_df”. The number of rows has been decreased to 8213 and the number of columns remains the same here.

Graphical user interface, text, application, email

Description automatically generated

**Figure 10: Showing the label encoding before performing train test splitting**

(Source: Retrieved from Jupyter Notebook)

The label encoding is one of the most important tasks here to preprocess the fraud dataset properly. This operation has been done by importing a necessary library “preprocessing” and there are two different columns which are preprocessed here “type” and “name\_Dest” (Minastireanu *et al.* 2019). The “X” and “Y” columns have also been defined here properly to split the dataset properly. The “X” contains five different columns and the “Y” contains only one column from the fraud dataset.

# **b) Architecture:**

It has been looked at different approaches like “Hadoop, Python, and Flink” when choosing an architecture. Because of its speed, fault tolerance, and scalability, we chose Apache Spark.

Text

Description automatically generated

**Figure 11: Showing the train test splitting**

(Source: Retrieved from Jupyter Notebook)

A proper model has been built as part of the system's design. The detection of online payment fraud is one use for big data machine learning systems. The system is made to predict whether a transaction is fraudulent by analyzing transaction data in real-time.

Text

Description automatically generated

**Figure 12: Showing the accuracy score by random forest classifier model**

(Source: Retrieved from Jupyter Notebook)

The best accuracy has been generated by a classifier machine learning model that is the “*Random Forest Classifier*”. There are two types of machine learning models: "XGB” and “Random Forest Classifier”. The “XGB” generates an 80% of accuracy score and the “Random forest classifier” model generates a 99% of accuracy score on the payment fraud detection data (Mittal *et al.* 2021). The best machine learning model that gives the proper prediction here is the second one. In this fraud detection technique, the checking of fraud data has been shown clearly here.

# **c) Visualization of data:**

Examining the data to discover relationships and patterns was a part of data visualization. Python's Matplotlib and Seaborn libraries were utilized for the visualization. Histograms, scatterplots, and boxplots are used here to visualise each column from the fraud payment dataset.

Chart

Description automatically generated

**Figure 13: Showing the feature analysis by histogram plotting**

(Source: Retrieved from Jupyter Notebook)

The above picture represents the feature analysis of the entire fraud dataset using seven different columns from the dataset (Rui *et al.* 2022). The columns which have been taken from the dataset to design such a plot are “step, amount, newbalanceDest, newbalanceOrig, OldbalanceOrg, IsFraud and isFlaggedFraudd”.

Graphical user interface

Description automatically generated with medium confidence

**Figure 14: Showing the feature analysis by obtaining the values**

(Source: Acquired from Jupyter Notebook)

The above picture represents to print the value of fraud detection here. There are seven different columns taken to perform this type of operation here properly. This particular operation has been performed here to generate two different values present in two different indexes.

Chart

Description automatically generated with low confidence

**Figure 15: Showing the feature analysis of k by histogram plotting**

(Source: Retrieved from Jupyter Notebook)

This figure shows the feature analysis and visualization of the rest columns present in the fraud detection dataset. These columns are “amount, isFraud, newbalanceOrig, and isFlaggedFraud”. The graph has been generated here using two values such as frequency and amount.

Chart, bar chart

Description automatically generated

**Figure 16: Showing the bar plot of cross\_tab**

(Source: Retrieved from Jupyter Notebook)

The above diagram represents cross tab calculated from the fraud detected value of is fraud column. There are five different vbalues generated by this particular column such as “CASH\_IN, CASH\_OUT, DEBIT\_type, PAYMENT, and TRANSFER” and the range of the value has been varied from 3600 to 8000.

Chart, box and whisker chart

Description automatically generated

**Figure 17: Showing the box plot by using the amount column from fraud dataset**

(Source: Acquired from Jupyter Notebook)

This picture shows the box plot by taking “isFlaggedFraud” column from the fraud dataset. The entire visualization represents the entire amount of flagged fraud detected by the countable variable.

Chart, scatter chart

Description automatically generated

**Figure 18: Showing the stripe plot by using the amount column from the fraud dataset**

(Source: Acquired from Jupyter Notebook)

This picture represents the amount plot from the fraud dataset by plotting a stripe plot. The Python library has been used to create this system.

Chart, histogram

Description automatically generated

**Figure 19: Showing the amount of fraud from the fraud dataset using histogram plotting**

(Source: Acquired from Jupyter Notebook)

This histogram has been plotted here to show the amount of fraud that has been detected by this particular system properly here.

Chart, bar chart

Description automatically generated

**Figure 20: Showing the amount of fraudsters from the fraud dataset using bar plotting**

(Source: Acquired from Jupyter Notebook)

This particular image represents the amount of fraudsters by plotting a bar graph. This graph has been generated after the calculation of twelve fraudster values.

Chart

Description automatically generated with medium confidence

**Figure 21: Showing the correlation matrix of the entire data**

(Source: Acquired from Jupyter Notebook)

This heatmap has been generated after performing each and every visualization. This particular visualization represents the relationship between each and every operation here on the basis of the fraud-detected data.

# **d) Machine Learning System Application:**

An online payment fraud detection system was implemented and evaluated as one of the machine learning system's applications. The system trained a model on a preprocessed data set before using the trained model to predict whether a transaction would be fraudulent. The system is 99 % accurate to predict fraud data. This is sufficient for identifying online payment fraud.

Graphical user interface, text, application

Description automatically generated

**Figure 22: Showing the importation of libraries for building models**

(Source: Acquired from Jupyter Notebook)

The necessary libraries for building the machine learning models have been imported properly to generate individual accuracy scores by evaluation of these models (Srokosz *et al.* 2023). There are two different models as “*Random Forest Classifier*” and “*XGB Classifier*”.

Text

Description automatically generated

**Figure 23: Showing the accuracy score by XGB Classifier Model**

(Source: Acquired from Jupyter Notebook)

This figure represents the accuracy score by predicting on the “is flag” data in test column. The accuracy score generated by this XGB classifier model is 80 %.

Text

Description automatically generated

**Figure 24: Showing the accuracy score by Random Forest Classifier Model**

(Source: Acquired from Jupyter Notebook)

The above figure shows the highest accuracy score according to the testing result in the fraud detection dataset. This particular model predicts on the test data to give an assurance of 99% and the data is fraud.

# **e) Conclusion and Reflection:**

Using a selected dataset, we developed a machine learning system to identify online payment fraud. The system is not accurate at 99% by the calculation of the accuracy score by the “*Random Forest Classifier Model*”, which is enough for this job. However, the system can be enhanced by evaluating other algorithms and adding features. The process was challenging but rewarding, and it taught me a variety of methods for building big data machine learning systems. The model predicts on the “is a fraud” column present in the dataset and then printing the data is in danger which means the data is fraud. The fraud detection procedure has been evaluated through these two model-building operations accurately.

# **Reference**

Alwadain, A., Rao, F. A., & Muneer, A. (2023). Estimating Financial Fraud through Transaction-Level Features and Machine Learning. *Mathematics, 11*(5), 1184. https://doi.org/10.3390/math11051184

Chaquet-Ulldemolins, J., Francisco-Javier Gimeno-Blanes, Moral-Rubio, S., Muñoz-Romero, S., & Rojo-Álvarez, J. (2022). On the Black-Box Challenge for Fraud Detection Using Machine Learning (II): Nonlinear Analysis through Interpretable Autoencoders. *Applied Sciences, 12*(8), 3856. https://doi.org/10.3390/app12083856

Emmanuel, I., Sun, Y., & Wang, Z. (2022). A machine learning based credit card fraud detection using the GA algorithm for feature selection. *Journal of Big Data, 9*(1)https://doi.org/10.1186/s40537-022-00573-8

Fang, Y., Zhang, Y., & Huang, C. (2019). Credit Card Fraud Detection Based on Machine Learning. *Computers, Materials, & Continua, 61*(1), 185-195. https://doi.org/10.32604/cmc.2019.06144

Han, Y., Yao, S., Wen, T., Tian, Z., Wang, C., & Gu, Z. (2020). Detection and Analysis of Credit Card Application Fraud Using Machine Learning Algorithms. *Journal of Physics: Conference Series, 1693*(1)https://doi.org/10.1088/1742-6596/1693/1/012064

Minastireanu, E., & Mesnita, G. (2019). An Analysis of the Most Used Machine Learning Algorithms for Online Fraud Detection. *Informatica Economica, 23*(1), 5-16. https://doi.org/10.12948/issn14531305/23.1.2019.01

Mittal, S., & Joshi, N. K. (2021). Optimizing High-Performance Weighted Extreme Learning Machine with DAPM for Imbalance Credit Card Fraud Detection. *Turkish Journal of Computer and Mathematics Education, 12*(11), 272-283. https://www.proquest.com/scholarly-journals/optimizing-high-performance-weighted-extreme/docview/2623921394/se-2

Rui, M. D., Firdaus, R., Jaleel, F., Pedro, N. M., Mata, M., & Li, G. (2022). Systemic Acquired Critique of Credit Card Deception Exposure through Machine Learning. *Journal of Open Innovation : Technology, Market, and Complexity, 8*(4), 192. https://doi.org/10.3390/joitmc8040192

Srokosz, M., Bobyk, A., Ksiezopolski, B., & Wydra, M. (2023). Machine-Learning-Based Scoring System for Antifraud CISIRTs in Banking Environment. *Electronics, 12*(1), 251. https://doi.org/10.3390/electronics12010251

# 