DEEP LEARNING APPROACHES FOR FRAUD DETECTION IN E – COMMERCE TRANSACTIONS

MOHAMED AZLAN AMEER OLI

UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 4

INITIAL FINDING AND RESULTS

4.1 Introduction

This chapter studies and analyze the dataset collected and shows a data visualization on analyzing the efficiency of fraudulent activities in e – commerce industries by using deep learning method. Exploratory Data Analysis (EDA) is used to show the patterns in the dataset collected and identify the meaningful results. Different kinds of approach and methods are used to visualize the fraud's datasets such as the preparations of the datasets, statistics of the data and comparison of RNN and LSTM model towards the end of this chapter.

4.2 Exploratory Data Analysis (EDA)

The EDA process is an important approach to ensure that the data's valuable insights are visualized accordingly. The visualized data shows identifying different patterns and anomalies. A comprehensive understanding of the data is obtained through a series of processes in the EDA analysis. Analyzing the available data and defining the issues are the first steps. The next steps are to arrange the data and check how many missing values or inconsistencies are in the dataset. To prevent any bias in the results, any gaps in the data must be properly filled, either by removing the records or by assigning the missing values.

To find patterns in the data, this EDA process looks at the distributions, overall collections and average of the data. Using SMOTE operations, the raw can be scaled, encoded or integrated with an artificial dataset to improve the analysis. The outcomes and patterns found in the dataset are highlighted through visualization such as graphs and charts. To increase the dependability of the analytic results, outliers must be addressed. To conclude, the results have been highlighted in summaries and graphics.

4.3 Steps of Exploratory Data Analysis (EDA)

This section discussed the steps of Exploratory Data Analysis (EDA) to conduct this study.

4.3.1 Data Collection

The "Fraudulent E-Commerce Transactions" dataset reflects transactional behaviors frequently seen on actual e-commerce websites. It covers both legitimate and fraudulent activities, as well as a wide range of customer behaviors and transaction details. The dataset is perfect for real-world applications because it includes features that are commonly collected during online purchase, such as payment methods, products purchase, customer's details, and device usage. Actual transactions logs are used to model the data to produce realistic distributions and correlations so the deep learning algorithm able to train and validate the data more accurately.

The dataset consists of 16 features and 1,472,952 transactions. Approximately 5% of these transactions are categorized as fraudulent and the remaining are non-fraudulent. This indicates an imbalance in the datasets, and it is also a situation involving actual fraud detections. To predict the accurate accuracy of the data SMOTE has been used to balance the datasets.

The features within the dataset are covered by several important categories. These include customer demographics such as customer's age, location and device used. Also, transactions related data such as transactions amount, date, payment method, products. Then, unique identifiers such as Transactions ID and Customer ID. In addition, the dataset also contains other features such as IP Address, Shipping and Billing Address, Account Age, Transaction Hours and a binary label that indicates fraud or non-fraud transactions. These categories improved the capacity to spot the trends and behavior associated with fraudulent activities and enabled EDA process analysis.

The dataset's structure makes it easier to develop the fraud detection algorithm by including the features such as Account Age, Days, Quantity, Transaction Hour, Is Fraudulent. The dataset is an important resource for analyzing the pattern of transactions in e-commerce industries and enhancing fraud prevention systems.

	Transaction ID	Customer ID	Transaction Amount	Transaction Date	Payment Method	Product Category	Quantity	Customer Age	Customer Location	Device Used	IP Address	Shipping Address	Billing Address	Is Fraudulent	Account Age Days	Transaction Hour
0	15d2e414-8735-46fc-9e02-80b472b2580f	d1b87f62-51b2-493b-ad6a-77e0fe13e785	58.09	2024-02-20 05:58:41	bank transfer	electronics	1	17	Amandaborough	tablet	212.195.49.198	Unit 8934 Box 0058\nDPO AA 05437	Unit 8934 Box 0058\nDPO AA 05437	0	30	5
1	0bfee1a0-6d5e-40da-a446-d04e73b1b177	37de64d5-e901-4a56-9ea0-af0c24c069cf	389.96	2024-02-25 08:09:45	debit card	electronics	2	40	East Timothy	desktop	208.106.249.121	634 May Keys\nPort Cherylview, NV 75063	634 May Keys\nPort Cherylview, NV 75063	0	72	8
2	e588eef4-b754-468e-9d90-d0e0abfc1af0	1bac88d6-4b22-409a-a06b-425119c57225	134.19	2024-03-18 03:42:55	PayPal	home & garden	2	22	Davismouth	tablet	76.63.88.212	16282 Dana Falls Suite 790\nRothhaven, IL 15564	16282 Dana Falls Suite 790\nRothhaven, IL 15564	0	63	3
3	4de46e52-60c3-49d9-be39-636681009789	2357c76e-9253-4ceb-b44e-ef4b71cb7d4d	226.17	2024-03-16 20:41:31	bank transfer	clothing	5	31	Lynnberg	desktop	207.208.171.73	828 Strong Loaf Apt. 646\nNew Joshua, UT 84798	828 Strong Loaf Apt. 646\nNew Joshua, UT 84798	0	124	20
4	074a76de-fe2d-443e-a00c-f044cdb68e21	45071bc5-9588-43ea-8093-023caec8ea1c	121.53	2024-01-15 05:08:17	bank transfer	clothing	2	51	South Nicole	tablet	190.172.14.169	29799 Jason Hills Apt. 439\nWest Richardtown, ...	29799 Jason Hills Apt. 439\nWest Richardtown, ...	0	158	5

Figure 4.1: Fraudulent E-Commerce Transactions Dataset

4.3.2 Import and Inspect Dataset

The initial step in Exploratory Data Analysis (EDA) is to determine the proportion of fraudulent and non-fraudulent transactions by analyzing the total number of transaction types. A significant class imbalance as shown in the bar chart below, 1,399,144 are non – fraudulent transactions and 73,838 are fraudulent ones. Only 5% of the dataset highlighted as fraud and 95% is non – fraud.

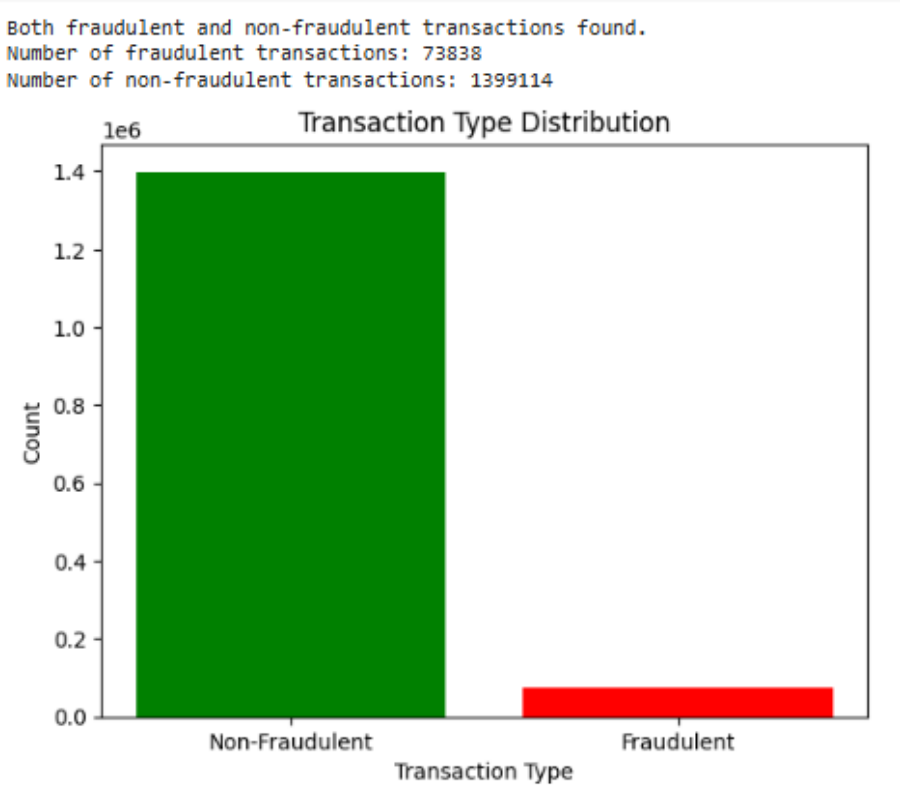

Figure 4.2: Transactions Type Distribution

Understanding this imbalance is important prior to starting the model-building process. If this imbalance data is not sufficient to address, most deep learning models fail to detect accurately because not enough data to predict accuracy. Therefore, the techniques like SMOTE is used for precision, recall and F1 – Score instead of just accuracy.

4.3.3 Demographic and Distribution Data

The term "demographic data" refers to statistical information about the characteristics of populations, such as age, gender, income, occupations, education and marital status in particular context. Demographic information is essential for data analysis to comprehend consumer behavior, segmenting target markets and risk assessment strategies (Bhatia, 2021). Data-driven decisions help certain organizations be able to meet their specific needs for their company.

The figure shows a histogram of the dataset's customer age distribution with Kernel Density Estimation (KDE) curve lay over. Based on the age distributions, slightly skewed to the right, the majority customers are in the age of 25 – 45. According to the peak age groups, 35 years old group and a few records show inaccurate age values, less than 0, which need to be analyzed further.

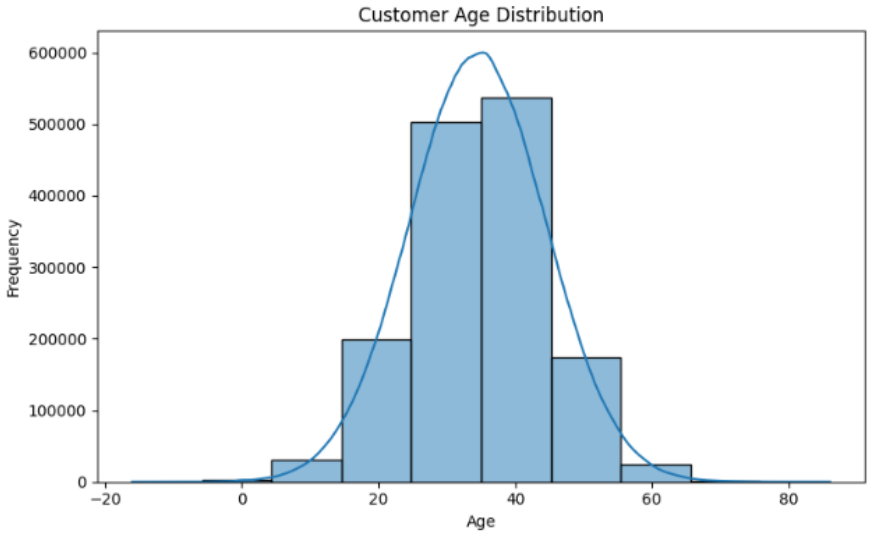

Figure 4.3: Customer Age Distribution

Next, figure 4.4 shows a pie chart that breaks down device usage into three categories: desktop, tablet, and mobile. Approximately one-third of all usage falls into each category, with mobile coming up slightly ahead at 33.4%, followed closely by tablets and desktops at 33.3% apiece. To guarantee consistency and usability across platforms, it is crucial to optimize digital experiences for all device formats, as this almost equal distribution implies that consumers access the platform or service from

all three device types in nearly similar quantities.

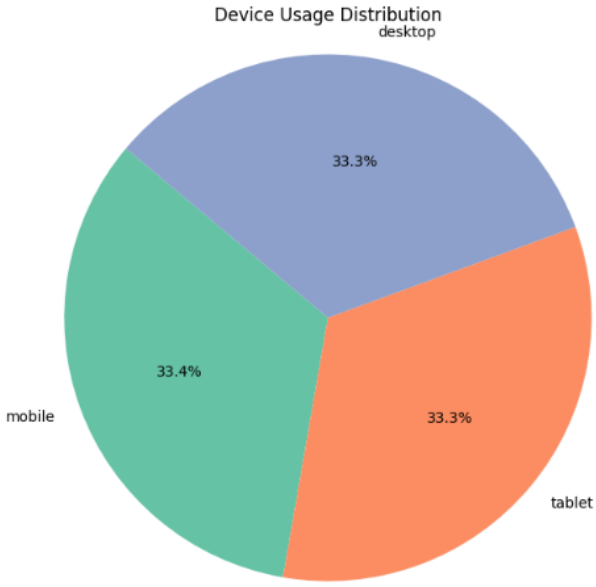

Figure 4.4: Device Usage Distribution

The distribution of the various payment methods shown in figure 4.5 that customers use such as credit card, PayPal, debit card, and bank transfer is shown in the bar chart. Each of the four methods is used with an equal rate and has a total of 360,000 transactions. These insights highlight how important it is to accept a range of payment methods to satisfy a wide range of customers' preferences and ensure a smooth shopping experience in e-commerce.

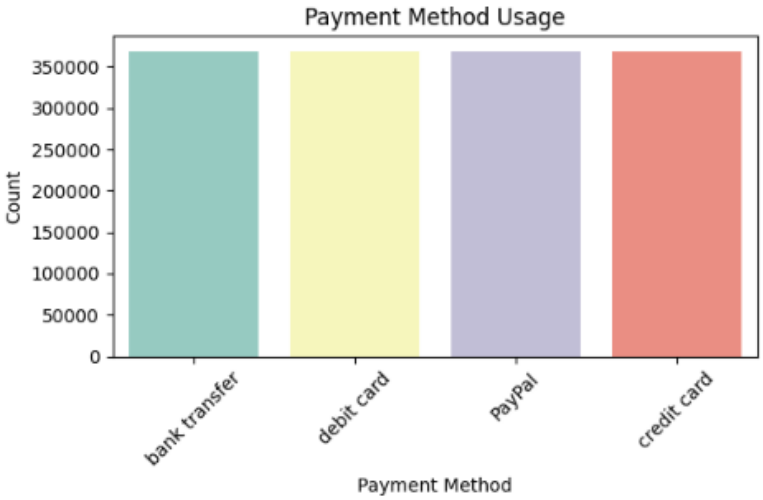

Figure 4.5: Payment Method Usage

4.3.4 Data Cleaning

The first step in the data cleaning process is to remove duplicate records of any rows with missing values in the target column, “Is Fraudulent”, to ensure that the dataset is suitable for modeling. Then, “Transaction Date” field is converted to proper date and time format. Also, the text fields such as “Payment Method”, “Product Category”, “Customer Location” and “Device Used” are standardized by converting all text into lowercase and deleting the whitespace. This will enhance grouping during the analysis and reduce inconsistencies. Finally, the output of the new cleaned data set is being displayed and confirming the process has been carried out correctly.

```
# Step 1: Drop rows with missing target or duplicate entries
df = df.drop_duplicates()
df = df.dropna(subset=['Is Fraudulent'])  # Replace with your actual target column name

# Convert 'Transaction Date' to datetime format
df['Transaction Date'] = pd.to_datetime(df['Transaction Date'])

# Standardize categorical text (lowercase)
text_columns = ['Payment Method', 'Product Category', 'Customer Location', 'Device Used']
for col in text_columns:
    df[col] = df[col].str.lower().str.strip()

df = df.reset_index(drop=True)
df.head()
rows,columns = df.shape
print(f"The dataset contains {rows} rows and {columns} columns.")
```

```
The dataset contains 1472952 rows and 16 columns.
```

Figure 4.6: Data Cleaning Code

4.3.5 Using SMOTE Model for Balancing Data

The figure shows the distributions of the transaction types after the dataset’s imbalance was adjusted by using Synthetic Minority Over – Sampling Technique (SMOTE) method. Initially, the dataset shows different in transactions that were fraudulent and non – fraudulent. After implementation of SMOTE method, the number of fraudulent transactions increase to 923,742 and non – fraudulent remained same. To lessen the bias in model training brought on by the imbalance, this adjustment was done to reach a 60:40 ratio between fraudulent and non-fraudulent classes. SMOTE enhances model performance and fairness in identifying fraudulent activity by artificially creating new samples of the minority class which is fraudulent transactions.

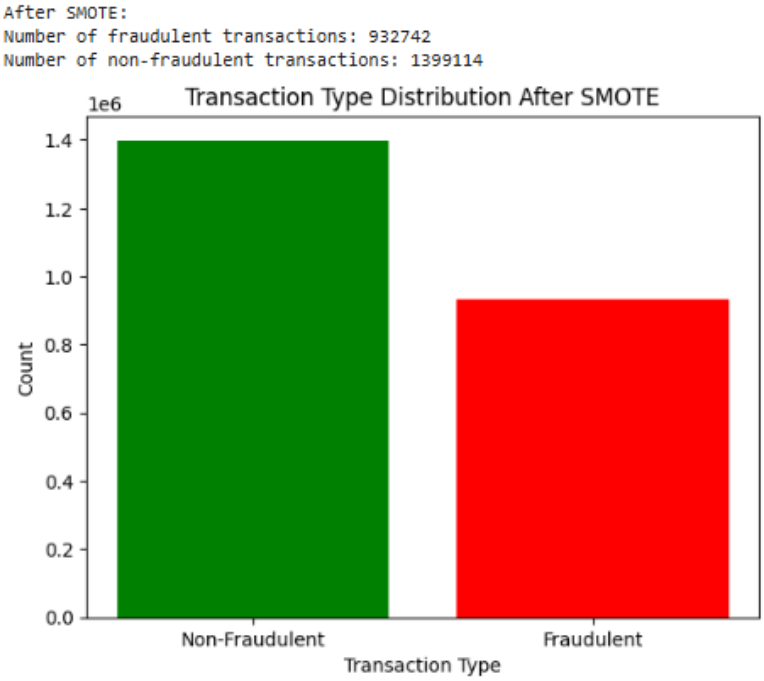

Figure 4.7: Transaction Type Distribution After SMOTE

4.4 Feature Extraction

The process of turning unprocessed data into useful inputs (features) that machine learning models may be trained on is known as feature extraction. Since well-designed features can greatly improve accuracy, generalization, and predictive performance, this step is crucial to evaluating the model's success. Features can be manually extracted using statistical methods or automatically extracted using algorithms like deep learning. A few common characteristics in fraud detection include transactions, quantity, time, customer's behavior, account age and location which give clues to analysis the fraud patterns.

Figure 4.8 shows a correlation heatmap for features derived from a resampled dataset intended for a fraud detection model. Each pair of features Pearson correlation coefficients which value range from -1 to 1 are displayed. Values near 0 indicate little to non – linear relationships, values near 1 indicates a strong and positive relationship. Meanwhile, values near -1 indicate a strong and negative relationship. For features like "Is Fraudulent" and "Transaction Amount: have a moderate positive correlation of

0.31, indicates that lager transactions amount linked as fraud. The negative, -0.29 in between “Account Age Days”, “Transaction Hours” and fraud indicating that the transactions at hours and newer account are more likely to be fraudulent. By determining which features are most helpful for fraud detections, the heatmap shows in the deep learning model’s feature selection process.

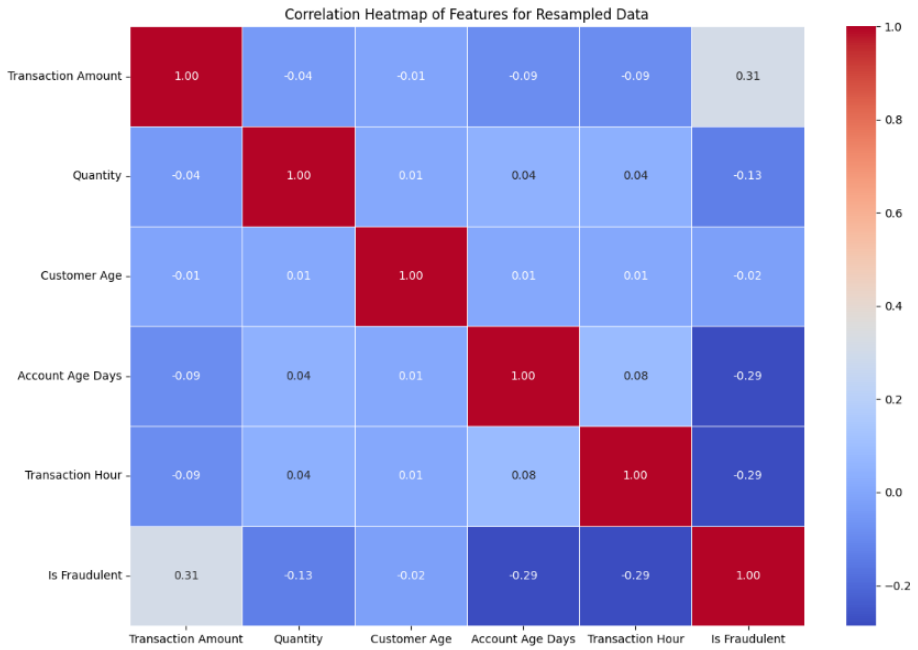

Figure 4.8: Correlation Heatmap of Features for Resampled Data

4.5 Data Modeling

Using algorithms like RNN (Recurrent Neural Network) and LSTM (Long Short-Term Memory) focus on creating models that capable of understanding time-based relationships in sequential data, such as time series or transaction logs.

4.5.1 LSTM Modeling

Figure 4.9 shows sequential neural network models using Keras. The model starts with 128 units LSTM layer that stores temporal dependencies in the input data and sends the complete output sequence back to the following layer. Next, Dropout layer with a rate of 0.4 randomly disables 40% of the neurons during training to help avoid overfitting. After processing the output from the first layer, 64 units LSTM layer

summarizes the sequence and returns final output. The model is then classified for binary classification tasks since it has a Dense layer with a single neuron and a sigmoid activation function that produces a value between 0 and 1.

```
# Build the model
model = Sequential()
model.add(LSTM(128, return_sequences=True, input_shape=input_shape))
model.add(Dropout(0.4))
model.add(LSTM(64))
model.add(Dense(1, activation='sigmoid'))
```

Figure 4.9: LSTM Modeling

4.5.2 RNN Modeling

Figure 4.10 shows sequential neural network models using Keras. The model starts with 128 units SimpleRNN layer that receives input data of the shape specified by *input_shape* and *return_sequence = True* that sends the entire sequence to the following layer. To prevent overfitting, Droupout later with rate of 0.4 randomly deactives 40% of neurons during training. The SimpleRNN of 64 units follows the output of the preceding layer and the sequence's final output. Next, binary classification tasks, Dense layer comprising a single neuron and a sigmoid activation function produces a value 0 and 1.

```
model = Sequential([
    SimpleRNN(128, return_sequences=True, input_shape=input_shape),
    Dropout(0.4),
    SimpleRNN(64),
    Dense(1, activation='sigmoid')
])
```

Figure 4.10: RNN Modeling

4.6 Model Evaluation

The model that has been successfully created previously will subsequently be processed for use with deep learning techniques. To get better accuracy, deep learning such as RNN and LSTM will be implemented in this study.

4.6.1 Initial Results of LSTM

The model evaluation results at Figure 4.11 show a reasonably good overall performance with a test accuracy of 75%. Both scikit – learn and the model's evaluation of a ROC – AUC score of 0.8316, shows the model has a high degree of discriminative power between the two classes. Then, Class 1, which is a minority class, has a lower precision of 66% compared to 76%, indicating that the model is reasonably good at detecting positives but produces more false positives.

Class 0, which is majority class has a precision of 84% and F1 – Score, 79% while class 1 which is minority class has a lower precision 0f 66% with a comparable recall of 76%. Next, a weighted average that accounts for class imbalance, the macro average F1 – Score of 76% indicates balanced performance across classes. To conclude, this model is effective at identifying fraudulent activities.

```
Test Loss: 0.4989
Test Accuracy: 0.7574
ROC-AUC Score (from sklearn): 0.8316
Test AUC (from model.evaluate): 0.8316

Classification Report:
              precision    recall  f1-score   support

           0       0.84      0.75      0.79    278751
           1       0.66      0.76      0.71    175171

    accuracy                           0.76    453922
   macro avg       0.75      0.76      0.75    453922
weighted avg       0.77      0.76      0.76    453922
```

Figure 4.11: Initial Results of LSTM

The confusion matrix at Figure 4.12 shows a binary classification models that classified “Fraud” and “Non – Fraud. The model accurately predicted 133, 998 fraud cases and 209, 801 non – fraud cases. Nevertheless, it incorrectly identified 41, 173 frauds as non – frauds which is false negatives and 68, 950 no – frauds as frauds which is false positives. Although the model’s overall number of accurate predictions is high, the high number of false positives and false negatives indicates that precision and recall could be increased.

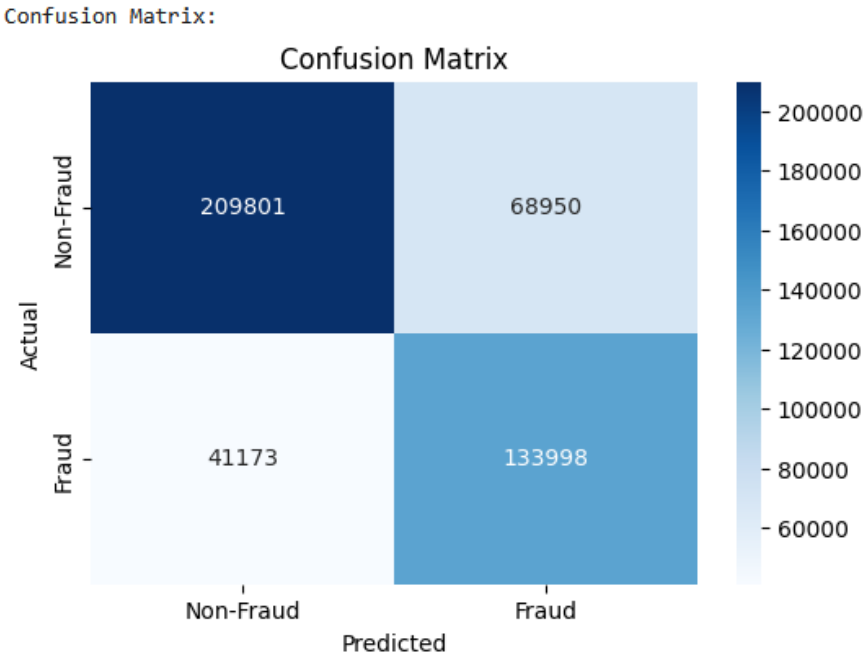

Figure: 4.12: Confusion Matrix of LSTM Model

4.6.2 Initial Results of RNN

According to the initial results of RNN model, it shows overall a good discrimination between classes with a test accuracy of 75% and a strong ROC -AUC score of approximately 82%. Meanwhile, F1 – Score shows 79% with a precision of 83% and the model did well for the non – fraud classes. Recall stays at 76% while precision falls to 66% for fraud. This indicates that the model detects majority of fraud cases but produces more false positives results. Although accuracy for fraud could be increased, the model appears to manage class imbalance well based on the balanced recall and F1 – Scores across classes.

```
Test Loss: 0.5001
Test Accuracy: 0.7577
ROC-AUC Score (from sklearn): 0.8284
Test AUC (from model.evaluate): 0.8283

Classification Report:
              precision    recall  f1-score   support

   Non-Fraud       0.83      0.76      0.79    278751
       Fraud       0.66      0.76      0.71    175171

    accuracy                           0.76    453922
   macro avg       0.75      0.76      0.75    453922
weighted avg       0.77      0.76      0.76    453922
```

Figure 4.13: Initial Results of RNN

The classification results for fraud detection are displayed in this confusion matrix. 131,843 fraud cases and 211,084 non-fraud cases were accurately predicted by the model. Nevertheless, 42,328 frauds were incorrectly classified as non-fraud which is false negatives and 67,667 non-frauds as fraud which is false positives. Although there is still a trade-off between detecting fraud and preventing false alarms, the high percentage of accurate predictions points to strong overall performance.

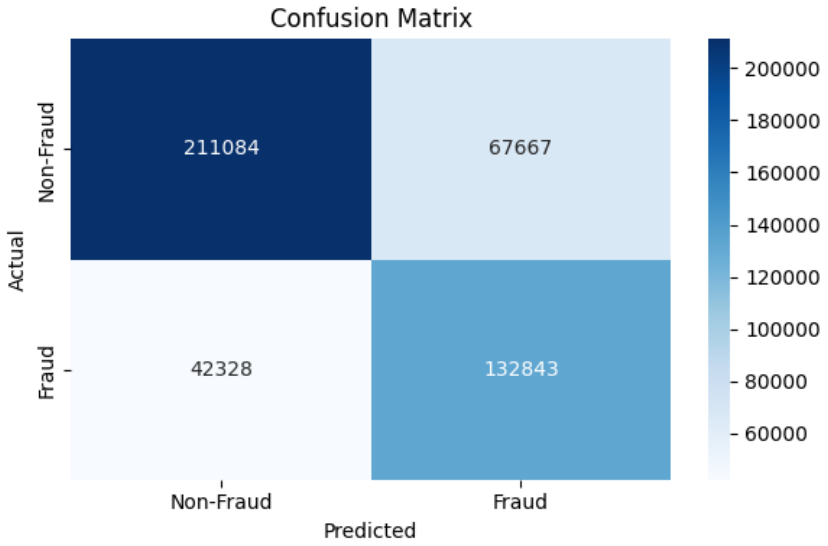

Figure 4.14: Confusion Matrix of RNN Model

4.7 Summary

This chapter also explored detailed exploratory data analysis of fraud detection in e-commerce industries by using the provided datasets. As a result of using data visualization, data cleaning process, data balancing, feature extraction, data modeling and the results of the models have been discussed in detail. The initial results of both RNN and LSTM models show almost the same accuracy.