Assesment Report

on

"Predict Credit Card Fraud: Develop a classification model to detect fraudulent transactions based on patterns in transaction amount, location, device usage, and user behavior."

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE AIML_B

By

NIRMAL SINGH(202401100400130)

Under the supervision of

"Abhishek Shukla"

KIET Group of Institutions, Ghaziabad

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University, Lucknow

(Formerly UPTU)

May, 2025

INTRODUCTION

With the rapid digitization of financial services, the number of online credit card transactions has increased exponentially. Unfortunately, this rise in digital activity has also given rise to more sophisticated fraudulent practices. Detecting such frauds in real-time is a challenging task for traditional rule-based systems.

Machine learning offers a powerful alternative by learning from past patterns and detecting anomalies in real-time. In this project, a Random Forest Classifier is employed to classify transactions based on anonymized features (V1 to V28), along with Time and Amount. The dataset used is a well-known credit card fraud detection dataset from Kaggle, containing real-world anonymized data.

METHODOLOGY

Dataset Upload

The dataset is uploaded using the Google Colab interface, allowing manual upload from the local system.

Data Preprocessing

- The dataset includes anonymized features (V1 to V28), Time, and Amount.

- The target column is Class (1 = fraud, 0 = non-fraud).

- Feature scaling is performed using StandardScaler for better performance.

Feature and Target Selection

- **Features:** All columns except Class

- **Target:** Class

Model Building

- Model: **Random Forest Classifier** with 100 decision trees (n_estimators=100)

- Data split: 70% training and 30% testing

Model Evaluation

- Metrics: **Confusion Matrix** and **Classification Report**

- The report includes accuracy, precision, recall, and F1-score.

Model Saving

- The trained model and the scaler are saved using joblib for reuse without retraining.

CODE

```python
# Step 1: Upload the dataset

from google.colab import files

import pandas as pd

import io


# Upload CSV file

uploaded = files.upload()

file_name = next(iter(uploaded))

df = pd.read_csv(io.BytesIO(uploaded[file_name]))


# Step 2: Preprocess the dataset

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import confusion_matrix, classification_report, accuracy_score, precision_score, recall_score

import seaborn as sns

import matplotlib.pyplot as plt

import joblib


# Step 3: Feature selection and scaling

X = df.drop(['Class'], axis=1)

y = df['Class']
```

```python
scaler = StandardScaler()

X_scaled = scaler.fit_transform(X)


# Step 4: Train/test split

X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3, random_state=42)


# Step 5: Train the model

model = RandomForestClassifier(n_estimators=100, random_state=42)

model.fit(X_train, y_train)


# Step 6: Predict

y_pred = model.predict(X_test)


# Step 7: Evaluation metrics

accuracy = accuracy_score(y_test, y_pred)

precision = precision_score(y_test, y_pred)

recall = recall_score(y_test, y_pred)


print(f"\nAccuracy: {accuracy:.4f}")

print(f"Precision: {precision:.4f}")

print(f" Recall: {recall:.4f}")


# Step 8: Confusion Matrix
```

```python
cm = confusion_matrix(y_test, y_pred)

plt.figure(figsize=(6,4))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Not Fraud', 'Fraud'],
yticklabels=['Not Fraud', 'Fraud'])

plt.xlabel('Predicted')

plt.ylabel('Actual')

plt.title(' Confusion Matrix - Heatmap')

plt.show()


# Optional: Detailed classification report

print("\n Classification Report:\n")

print(classification_report(y_test, y_pred))


# Step 9: Save model and scaler

joblib.dump(model, 'fraud_detector_model.pkl')

joblib.dump(scaler, 'scaler.pkl')

print("\n Model and scaler saved successfully.")
```
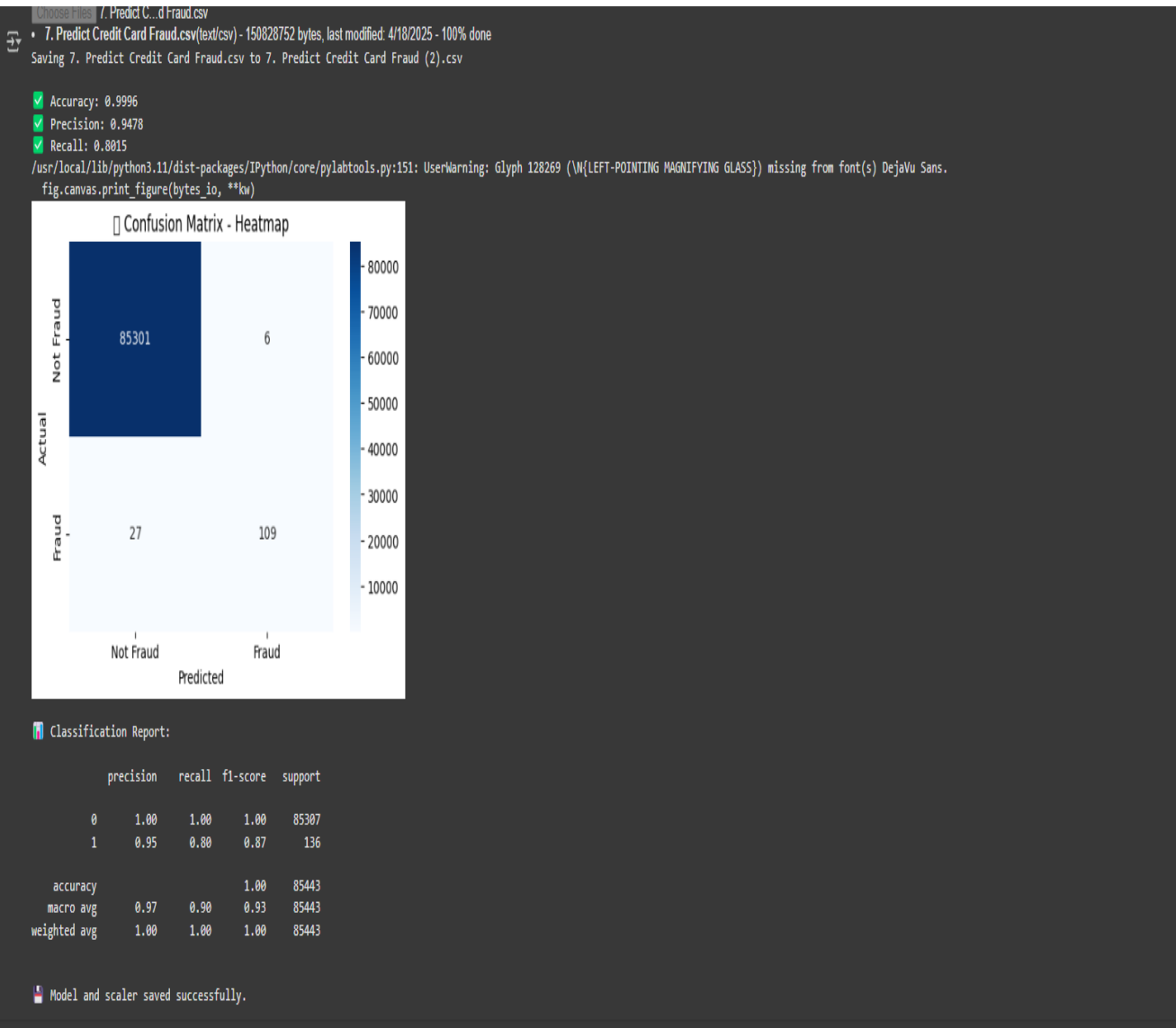
OUTPUT

• **7. Predict Credit Card Fraud.csv**(text/csv) - 150828752 bytes, last modified: 4/18/2025 - 100% done

Saving 7. Predict Credit Card Fraud.csv to 7. Predict Credit Card Fraud (2).csv

☑ Accuracy: 0.9996
☑ Precision: 0.9478
☑ Recall: 0.8015
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 128269 (\N{LEFT-POINTING MAGNIFYING GLASS}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)

🔢 Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	85307
1	0.95	0.80	0.87	136
accuracy			1.00	85443
macro avg	0.97	0.90	0.93	85443
weighted avg	1.00	1.00	1.00	85443

💾 Model and scaler saved successfully.

REFERENCES / CREDITS

Dataset Source: Kaggle Credit Card Fraud Detection Dataset

Documentation:

- scikit-learn

- pandas

- joblib

Platform Used: Google Colab