```python
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
import matplotlib.pyplot as plt
import seaborn as sns

class FraudDetection:
    def __init__(self, cc_info_path, transactions_path):
        self.cc_info_path = 'cc_info.csv'
        self.transactions_path = 'transactions.csv'
        self.cc_info = pd.read_csv(cc_info_path)
        self.transactions = pd.read_csv(transactions_path)
        self.data = None
        self.model = None

    def preprocess_data(self):
        self.cc_info = pd.read_csv('cc_info.csv')
        self.transactions = pd.read_csv('transactions.csv')
        print("Shape of Credit Card Info Data:", self.cc_info.shape)
        print("Shape of Transactions Data:", self.transactions.shape)

        self.data = pd.merge(self.transactions, self.cc_info,
on='credit_card', how='inner')
        print("\nPreview of Merged Data:")
        print(self.data.head())
        print("\nShape of Merged Data:", self.data.shape)
        print("\nChecking for missing values before dropping:")
        print(self.data.isnull().sum())
        self.data.dropna(inplace=True)
        print("\nShape after dropping missing values:",
self.data.shape)

        if self.data.empty:
            raise ValueError("The dataset is empty after
preprocessing. Please check the input data and merging process.")
        self.data['transaction_percentage'] =
self.data['transaction_dollar_amount'] /
self.data['credit_card_limit']
        print("\nPreview of Data after Feature Engineering:")
        print(self.data[['transaction_dollar_amount',
'credit_card_limit', 'transaction_percentage']].head())
        features = ['transaction_dollar_amount', 'Long', 'Lat',
'credit_card_limit', 'transaction_percentage']
        self.data = self.data[features]
        print("\nPreview of Selected Features:")
        print(self.data.head())
        scaler = StandardScaler()
        self.data_scaled = scaler.fit_transform(self.data)
        print("\nData scaling completed successfully.")

    def build_autoencoder(self, input_dim):
        self.model = Sequential([
            Dense(64, input_dim=input_dim, activation='relu'),
            Dropout(0.2),
            Dense(32, activation='relu'),
            Dense(16, activation='relu'),
            Dense(32, activation='relu'),
            Dropout(0.2),
            Dense(input_dim, activation='linear')])
        self.model.compile(optimizer='adam', loss='mse')

    def train_model(self, epochs=20, batch_size=32):
        X_train, X_val = train_test_split(self.data_scaled,
test_size=0.2, random_state=42)
        self.model.fit(X_train, X_train, epochs=epochs,
batch_size=batch_size, validation_data=(X_val, X_val), verbose=1)

    def detect_anomalies(self, threshold=0.01):
        reconstructions = self.model.predict(self.data_scaled)
        reconstruction_errors = np.mean(np.square(self.data_scaled -
reconstructions), axis=1)
        anomaly_threshold = np.percentile(reconstruction_errors, 100 *
(1 - threshold))
        self.data['anomaly'] = (reconstruction_errors >
anomaly_threshold).astype(int)
        return self.data[['anomaly', 'transaction_dollar_amount',
'Long', 'Lat', 'credit_card_limit', 'transaction_percentage']]

    def visualize_results(self):
        plt.figure(figsize=(10, 6))
        sns.scatterplot(data=self.data, x='Long', y='Lat',
hue='anomaly', palette={0: 'blue', 1: 'red'}, alpha=0.6)
        plt.title('Geographical Distribution of Anomalies')
        plt.xlabel('Longitude')


if __name__ == "__main__":
    fraud_detection = FraudDetection('cc_info.csv',
'transactions.csv')

    print("Preprocessing Data...")
    fraud_detection.preprocess_data()

    print("Building Autoencoder Model...")

fraud_detection.build_autoencoder(input_dim=fraud_detection.data_scale
d.shape[1])

    print("Training Model...")
    fraud_detection.train_model(epochs=50, batch_size=64)

    print("Detecting Anomalies...")
    results = fraud_detection.detect_anomalies(threshold=0.01)
    print("\nAnomalies Detected:")
    print(results.head())

    print("Visualizing Results...")
    fraud_detection.visualize_results()

    results.to_csv('fraud_detection_results.csv', index=False)
    print("\nResults saved to 'fraud_detection_results.csv'")

Preprocessing Data...
Shape of Credit Card Info Data: (984, 5)
Shape of Transactions Data: (294588, 5)

Preview of Merged Data:
    credit_card              date  transaction_dollar_amount
Long  \
0  1.000000e+15    9/11/2015 0:32                      43.78 -
80.174132
1  1.010000e+15  10/24/2015 22:23                     103.15 -
80.194240
2  1.020000e+15  10/26/2015 18:19                      48.55 -
80.211033
3  1.070000e+15  10/22/2015 19:41                     136.18 -
80.174138
4  1.080000e+15  10/26/2015 20:08                      71.82 -
80.238720

         Lat        city state  zipcode  credit_card_limit
0  40.267370     Houston    PA    15342              20000
1  40.180114  Washington    NH     3280              15000
2  40.313004   Charlotte    VT     5445              28000
3  40.290895      Dallas    PA    18612              10000
4  40.166719     Houston    PA    15342              10000

Shape of Merged Data: (8641, 9)

Checking for missing values before dropping:
credit_card                  0
date                         0
transaction_dollar_amount    0
Long                         0
Lat                          0
city                         0
```

```
state                       0
zipcode                     0
credit_card_limit           0
dtype: int64

Shape after dropping missing values: (8641, 9)

Preview of Data after Feature Engineering:
   transaction_dollar_amount  credit_card_limit
transaction_percentage
0                      43.78              20000
0.002189
1                     103.15              15000
0.006877
2                      48.55              28000
0.001734
3                     136.18              10000
0.013618
4                      71.82              10000
0.007182

Preview of Selected Features:
   transaction_dollar_amount       Long        Lat  credit_card_limit
\
0                      43.78 -80.174132  40.267370              20000

1                     103.15 -80.194240  40.180114              15000

2                      48.55 -80.211033  40.313004              28000

3                     136.18 -80.174138  40.290895              10000

4                      71.82 -80.238720  40.166719              10000


   transaction_percentage
0                0.002189
1                0.006877
2                0.001734
3                0.013618
4                0.007182

Data scaling completed successfully.
Building Autoencoder Model...
Training Model...
Epoch 1/50
108/108 [==============================] - 1s 5ms/step - loss: 0.6688
- val_loss: 0.3233
Epoch 2/50
108/108 [==============================] - 0s 2ms/step - loss: 0.2788
```

```
- val_loss: 0.0828
Epoch 3/50
108/108 [==============================] - 0s 3ms/step - loss: 0.1938
- val_loss: 0.0593
Epoch 4/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1467
- val_loss: 0.0439
Epoch 5/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1408
- val_loss: 0.0317
Epoch 6/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1206
- val_loss: 0.0308
Epoch 7/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1314
- val_loss: 0.0439
Epoch 8/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1218
- val_loss: 0.0272
Epoch 9/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1243
- val_loss: 0.0226
Epoch 10/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1087
- val_loss: 0.0282
Epoch 11/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1019
- val_loss: 0.0261
Epoch 12/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0938
- val_loss: 0.0207
Epoch 13/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1112
- val_loss: 0.0258
Epoch 14/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1004
- val_loss: 0.0160
Epoch 15/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0898
- val_loss: 0.0241
Epoch 16/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1025
- val_loss: 0.0182
Epoch 17/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0975
- val_loss: 0.0256
Epoch 18/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0971
- val_loss: 0.0296
```

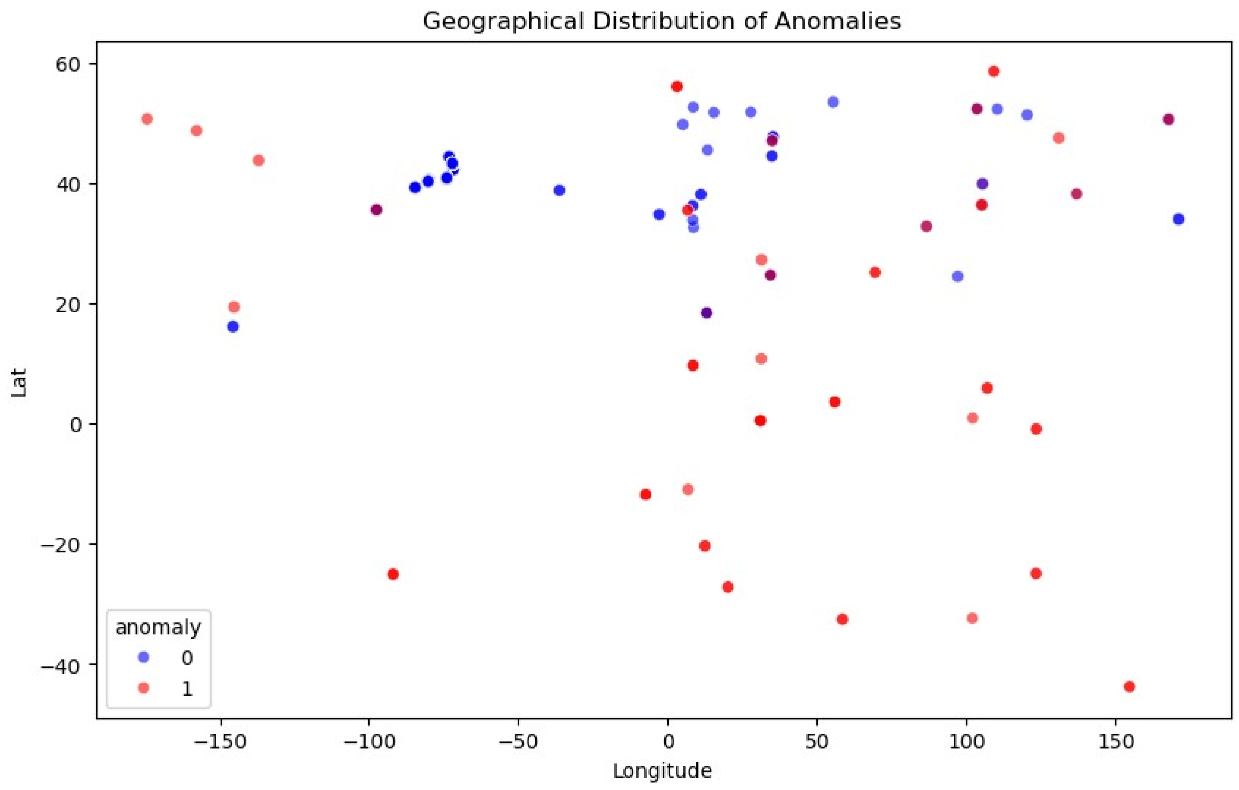
```
Epoch 19/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1036
- val_loss: 0.0185
Epoch 20/50
108/108 [==============================] - 0s 2ms/step - loss: 0.1085
- val_loss: 0.0247
Epoch 21/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0939
- val_loss: 0.0153
Epoch 22/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0943
- val_loss: 0.0200
Epoch 23/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0925
- val_loss: 0.0179
Epoch 24/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0885
- val_loss: 0.0283
Epoch 25/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0935
- val_loss: 0.0217
Epoch 26/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0904
- val_loss: 0.0185
Epoch 27/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0913
- val_loss: 0.0214
Epoch 28/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0941
- val_loss: 0.0193
Epoch 29/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0806
- val_loss: 0.0222
Epoch 30/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0887
- val_loss: 0.0203
Epoch 31/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0863
- val_loss: 0.0221
Epoch 32/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0835
- val_loss: 0.0206
Epoch 33/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0789
- val_loss: 0.0164
Epoch 34/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0790
- val_loss: 0.0153
Epoch 35/50
```

```
108/108 [==============================] - 0s 3ms/step - loss: 0.0908
- val_loss: 0.0155
Epoch 36/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0729
- val_loss: 0.0181
Epoch 37/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0733
- val_loss: 0.0144
Epoch 38/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0850
- val_loss: 0.0254
Epoch 39/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0754
- val_loss: 0.0171
Epoch 40/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0800
- val_loss: 0.0211
Epoch 41/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0742
- val_loss: 0.0218
Epoch 42/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0791
- val_loss: 0.0212
Epoch 43/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0788
- val_loss: 0.0184
Epoch 44/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0819
- val_loss: 0.0195
Epoch 45/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0805
- val_loss: 0.0191
Epoch 46/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0839
- val_loss: 0.0200
Epoch 47/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0744
- val_loss: 0.0229
Epoch 48/50
108/108 [==============================] - 0s 3ms/step - loss: 0.0812
- val_loss: 0.0215
Epoch 49/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0701
- val_loss: 0.0192
Epoch 50/50
108/108 [==============================] - 0s 2ms/step - loss: 0.0732
- val_loss: 0.0151
Detecting Anomalies...
271/271 [==============================] - 0s 1ms/step
```

```
Anomalies Detected:
   anomaly  transaction_dollar_amount       Long        Lat  \
0        0                      43.78 -80.174132  40.267370
1        0                     103.15 -80.194240  40.180114
2        0                      48.55 -80.211033  40.313004
3        0                     136.18 -80.174138  40.290895
4        0                      71.82 -80.238720  40.166719

   credit_card_limit  transaction_percentage
0              20000                0.002189
1              15000                0.006877
2              28000                0.001734
3              10000                0.013618
4              10000                0.007182
Visualizing Results...

Results saved to 'fraud_detection_results.csv'
```


Geographical Distribution of Anomalies
Lat
Longitude
anomaly