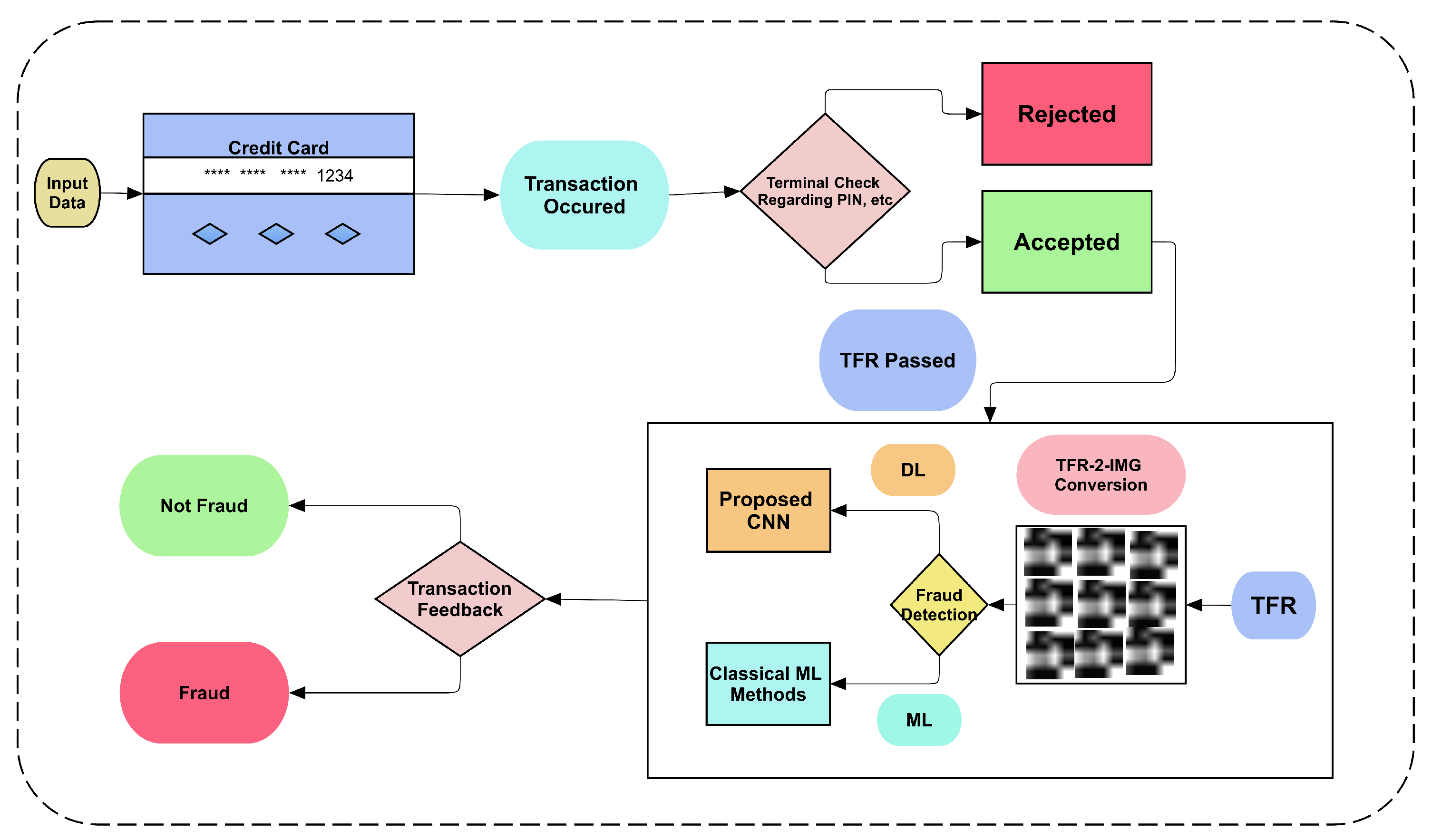
# CREDIT CARD FRAUD DETECTION



* Credit card fraud detection commonly employs a variety of machine learning algorithms. The choice of algorithm depends on the nature of the data, the size of the dataset, and the specific requirements of the detection task. Some commonly used algorithms include:
* Logistic Regression:
* Logistic Regression is a simple yet powerful algorithm for binary classification. It is easy to implement, interpret, and serves as a baseline model in many fraud detection systems.

#### Credit card fraud detection often involves the use of various libraries in Python:

* Scikit-learn:
* Scikit-learn is a popular machine learning library in Python. It provides simple and efficient tools for data mining and data analysis. In credit card fraud detection, Scikit-learn is commonly used for implementing machine learning algorithms such as decision trees, random forests, support vector machines (SVM), and more. The library offers modules for model training, evaluation, and preprocessing.

#### Pandas:

* Pandas is an open-source data manipulation and analysis library for Python. It provides easy-to-use data structures like DataFrame and Series, which are built on top of the NumPy library. Pandas is widely used for tasks related to data cleaning, exploration, and analysis in data science and machine learning.
* import pandas as pd

# Creating a DataFrame

data = {'Name': ['Alice', 'Bob', 'Charlie'],

'Age': [25, 30, 35],

'City': ['New York', 'San Francisco', 'Los Angeles']}

df = pd.DataFrame(data)

print(df)

##### Numpy:

* NumPy, which stands for Numerical Python, is a powerful open-source library in Python for numerical computing. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these elements. NumPy is a fundamental package for scientific computing in Python and is widely used in fields such as machine learning, data science, and engineering.
* import numpy as np

# Creating a NumPy array

arr = np.array([1, 2, 3, 4, 5])

# Performing a vectorized operation

arr\_squared = arr\*\*2

print(arr\_squared)

#### Matplotlib:

* import matplotlib.pyplot as plt

# Sample data

x = [1, 2, 3, 4, 5]

y = [2, 4, 6, 8, 10]

# Create a line plot

plt.plot(x, y)

# Add labels and title

plt.xlabel('X-axis')

plt.ylabel('Y-axis')

plt.title('Simple Line Plot')

# Show the plot

plt.show()

* Matplotlib is a comprehensive 2D plotting library for Python. It is widely used for creating static, animated, and interactive visualizations in Python. Matplotlib is particularly popular in the fields of data science, machine learning, and scientific research, where visualizing data is a crucial aspect of the analysis process

#### Seaborn:

* Seaborn is a statistical data visualization library in Python built on top of Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn comes with several built-in themes and color palettes to make it easy to create aesthetically pleasing visualizations. It's particularly well-suited for visualizing complex datasets with multiple variables.
* import seaborn as sns

import matplotlib.pyplot as plt

# Sample data

x = [1, 2, 3, 4, 5]

y = [2, 4, 5, 4, 5]

# Create a scatter plot with a linear regression line

sns.regplot(x=x, y=y)

# Show the plot

plt.show()

#### Overall Workflow Of Credit Card Fraud Detection:

#### Data Source:

* A good data source for credit card fraud detection using data science should be accurate,complete,covering the geographic area of interest,accessible.
* Dataset Link:(“https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud”)

#### Program:

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score, confusion\_matrix

# Load the dataset

data = pd.read\_csv('credit\_card\_data.csv')

# Preprocess the data using Pandas

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(features, labels, test\_size=0.2, random\_state=42)

# Create and train a Random Forest classifier

clf = RandomForestClassifier()

clf.fit(X\_train, y\_train)

# Make predictions on the test set

predictions = clf.predict(X\_test)

# Evaluate the model

accuracy = accuracy\_score(y\_test, predictions)

conf\_matrix = confusion\_matrix(y\_test, predictions)

print("Accuracy:", accuracy)

print("Confusion Matrix:\n", conf\_matrix)

#### Conclusion:

* In the phase 2 concusion, we will summarize the key findings and insights from the advanced techniques. We will reitrate the impact of these technique on improving the accuracy and robustness of credit card fraud detection.