Traffic Congestion Prediction Report

Project Title: Traffic Congestion Prediction

Submitted by:
Name: Krishna Singh
Roll No: 2024011008000140
Department: Computer Science and Engineering (AI) SECTION B
Institute: KIET Group of Institutions, Ghaziabad
Date: 22 April 2025

Introduction

Traffic congestion is a significant challenge in urban planning and transportation systems. It leads to longer travel times, increased fuel consumption, and elevated stress levels for commuters. With the advent of sensor-based data collection from roads, it is possible to use machine learning techniques to predict congestion levels based on factors like time of day, traffic volume, vehicle speed, etc.

This project aims to develop a model that classifies road sections into three categories of congestion: High, Medium, and Low. The data used is collected from traffic sensors and includes multiple features influencing traffic flow.

Methodology

The project follows the following steps:

1. Data Upload: The CSV file is uploaded using Google Colab's `files.upload()` function.
2. Data Cleaning: Column names are stripped of whitespace, and missing values are dropped.
3. Feature Encoding: All categorical columns, including time-related fields like 'Day' or 'Time', are encoded using LabelEncoder.
4. Feature and Target Selection: The target variable is automatically detected if it includes terms like "congestion" and "level".
5. Train/Test Split: The data is split into training and testing sets (80/20 ratio).
6. Model Training: A Random Forest Classifier is used for training.
7. Evaluation: The model is evaluated using a classification report and a confusion matrix.
8. Feature Importance: We visualize which features most influence the model's predictions.

CODE

The Python code was implemented and executed in Google Colab. Below is the complete code used for the traffic congestion prediction model.

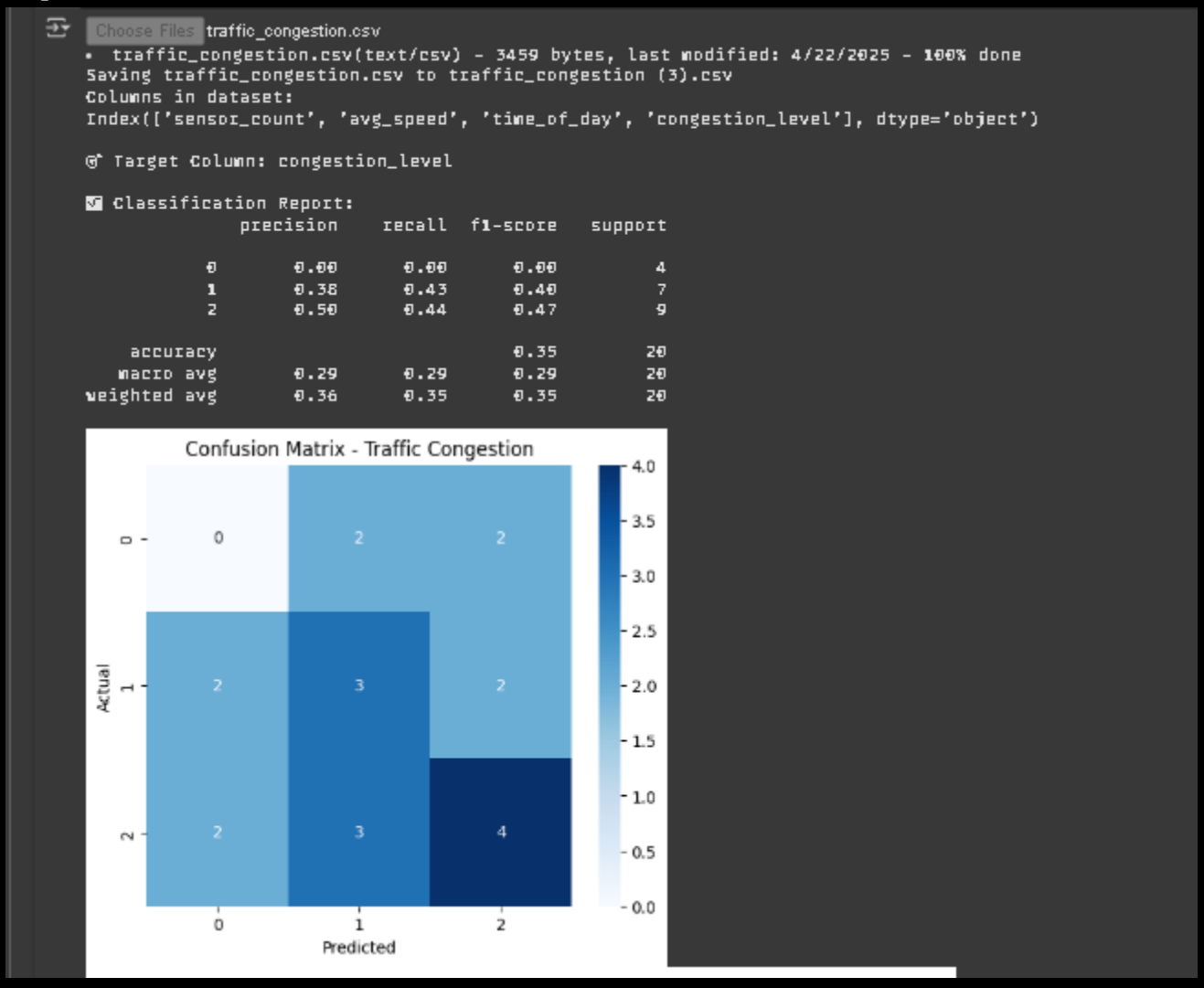
```
# Step-by-step ML code from uploading data to evaluating performance
from google.colab import files
uploaded = files.upload()

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix

data = pd.read_csv('traffic_congestion.csv')
data.columns = data.columns.str.strip()

target_column = None
for col in data.columns:
    if "congestion" in col.lower() and "level" in col.lower():
        target_column = col
        break

data.dropna(inplace=True)

label_encoder = LabelEncoder()
```

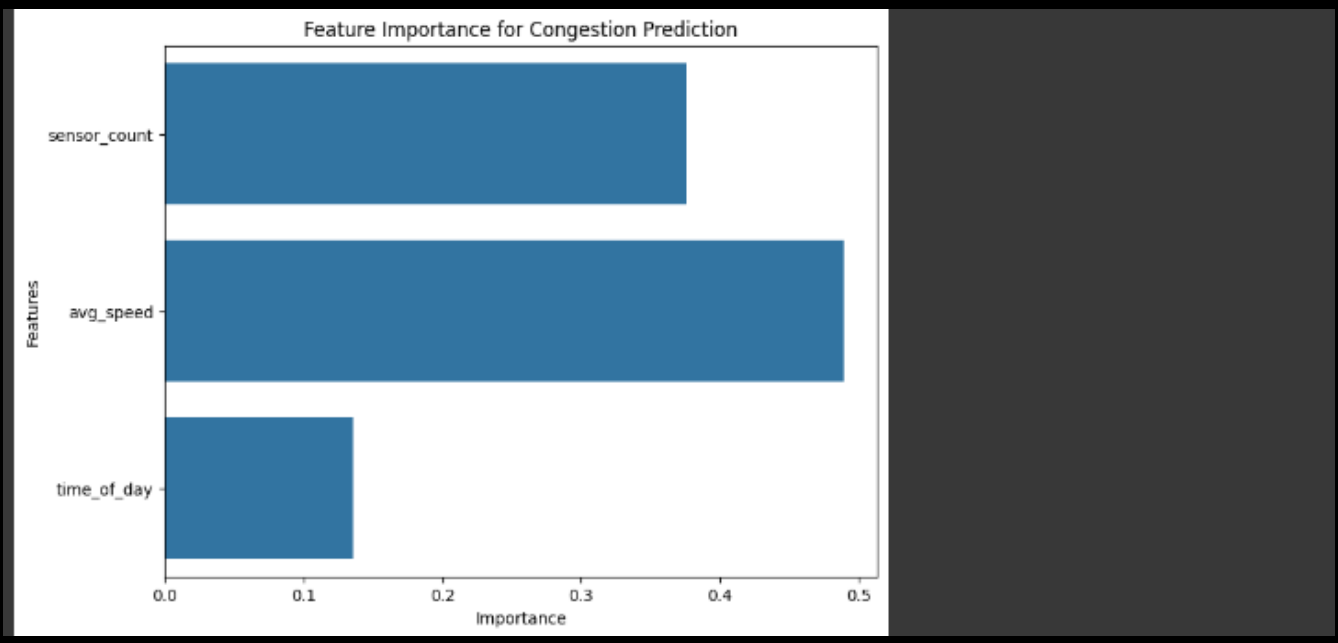
```
for col in data.columns:
    if data[col].dtype == 'object' and col != target_column:
        data[col] = label_encoder.fit_transform(data[col])

if data[target_column].dtype == 'object':
    data[target_column] = label_encoder.fit_transform(data[target_column])

X = data.drop(target_column, axis=1)
y = data[target_column]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

print(classification_report(y_test, y_pred))
conf_matrix = confusion_matrix(y_test, y_pred)
```

Output / Result

The model was evaluated using a classification report and confusion matrix. The classification report provided precision, recall, and F1-score for each class (Low, Medium, High congestion).

Additionally, feature importance was plotted to analyze which parameters most impacted the model's decisions.




Feature Importance for Congestion Prediction
sensor_count
avg_speed
time_of_day
Features
0.0
0.1
0.2
0.3
0.4
0.5
Importance

References / Credits

1. Dataset: traffic_congestion.csv (provided by instructor)
2. Tools Used: Google Colab, Python, pandas, seaborn, matplotlib, scikit-learn