Traffic Management System Using IOT, Data Analytics And Machine Learning

OBJECTIVES:

The objective of the project is to provide real-time traffic data to users, assisting them in selecting the most efficient route to their destination. Additionally, the project aims to utilize machine learning to predict congestion, offering suggestions that incorporate parking availability when it is needed.

PROJECT DESCRIPTION:

- The Traffic Management project aims to leverage IoT devices and data analytics to monitor traffic flow and congestion in real-time.
- By providing commuters with access to this information through a public platform or mobile apps, the project intends to assist them in making informed decisions about their routes, ultimately alleviating traffic congestion.
- The project involves defining clear objectives, designing the IoT traffic monitoring system, developing the traffic information platform, and integrating them using IoT technology and Python.

CONGESTION PREDICTION:

Congestion prediction in traffic management involves using data and technology to anticipate and manage traffic congestion effectively. Here are some key methods and techniques used for congestion prediction:

- **Data Collection:** Gathering real-time data from traffic cameras, Google map API for Traffic data and from the location of mobile apps, to monitor traffic conditions and collect relevant information.
- **Machine Learning:** Utilizing machine learning algorithms to analyze historical and real-time traffic data, weather conditions, and special events to predict congestion patterns.
- **Predictive Analytics:** Using statistical techniques and historical data to forecast congestion during peak hours or specific events.
- **Dynamic Routing:** Providing drivers with real-time route recommendations to avoid congestion-prone areas.

There are several Machine learning algorithm that makes prediction much easier such as

- Linear regression
- Logic regression
- Support vector machines
- Decision trees etc.

LINEAR REGRESSION ALGORITHM:

Linear regression is a supervised machine learning algorithm used for predicting a continuous target variable based on one or more input features.
It assumes a linear relationship between the inputs and the target.

KEY COMPONENTS OF LINEAR REGRESSION:

- **Linear Equation:** The linear regression model is represented as: **`y = mx + b`**, where **`y`** is the target variable, **`x`** is the input feature, **`m`** is the slope (weight), and `b` is the intercept (bias).
- **Least Squares Method:** Linear regression typically uses the least squares method to find the values of **`m`** and **`b`** that minimize the sum of squared errors between the predicted and actual values.
- **Simple Vs Multiple Linear Regression:** In simple linear regression, there's one input variable, while in multiple linear regression, there are also some multiple input variables.
- **Assumptions:** Linear regression assumes that the relationship between the input features and the target variable is linear, that the errors are normally distributed and have constant variance, and that the input features are not highly correlated.
- **Evaluation:** Common metrics for evaluating linear regression models include Mean Squared Error (MSE), R-squared (R^2), and others to assess the model's accuracy.

DATASET :

The data set is automatically generated with the help of Raspberry Pi which deducts the vehicle and changes the traffic lights accordingly. The following is just a sample of data for one month January with recorded data on vehicle count and Festivals of that month to make prediction easier.

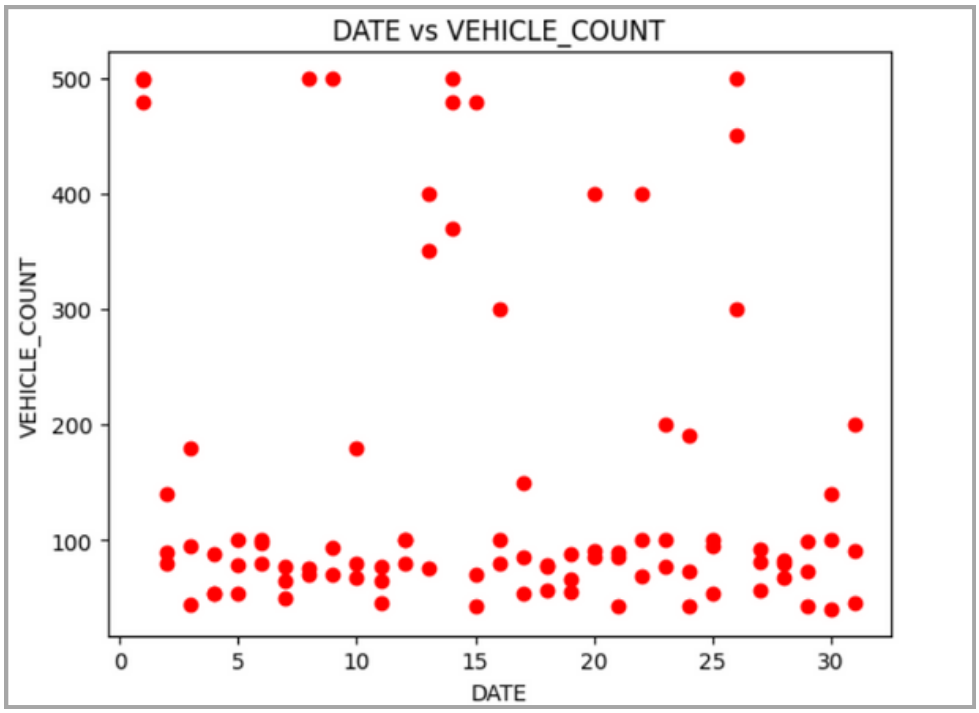
	YEAR	MONTH	DATE	FESTIVAL	VEHICLE_COUNT
0	2021	January	1	new_year_day	500
1	2021	January	2	_	80
2	2021	January	3	sunday	180
3	2021	January	4	_	88
4	2021	January	5	_	100
...	...	...	...	...	...
88	2023	January	27	_	57
89	2023	January	28	_	67
90	2023	January	29	sunday	99
91	2023	January	30	_	100
92	2023	January	31	_	45

93 rows × 5 columns

THE PROGRAM AND DOCUMENT FOR THIS VEHICLE PREDICTION ARE ATTACHED ON THE PHASE 4 REPOSITORY.

(CONGESTION_PREDICTION.IPYNB / CONGESTION_PREDICTION.PDF)

Linear regression :- Data Virtualization. (original data)



Linear regression :- Data Virtualization. (Trained Data)

Moreover, enhancing the training data would yield more precise outcomes. During festive seasons, the likelihood of congestion increases. To mitigate this issue, consider training the model with specific dates, festival names, and the total count of occurrences

Consider pongal as the festival and the trained data set is as follows.

	YEAR	MONTH	DATE	FESTIVAL	VEHICLE_COUNT
13	2021	January	14	pongal	480
44	2022	January	14	pongal	500
76	2023	January	15	pongal	480

To predict the congestion on January 2024.

```
def vehicle_cnt(a):
    result = regressor.predict(np.array(a).reshape(1, -1))
    return(result[0,0])
print("It is possible to travel",int(vehicle_cnt(4)), end="")
print(" Vehicles During Pongal 2024")
```

```
It is possible to travel 490 Vehicles During Pongal 2024
```

Please refer above content in the PDF (Congestion_Prediction.pdf) for more details.

The predicted data is send through the API from the **HERO KU** platform and received on the user app.

APPLICATION DEVELOPMENT:

The App is developed by using Kotlin as a primary language and interfacing heroku and firebase to get real time data in traffic deduction and prediction.

Kotlin: a statically typed language, is extensively utilized in application development, especially for Android, offering enhanced null safety and versatility for various application types.

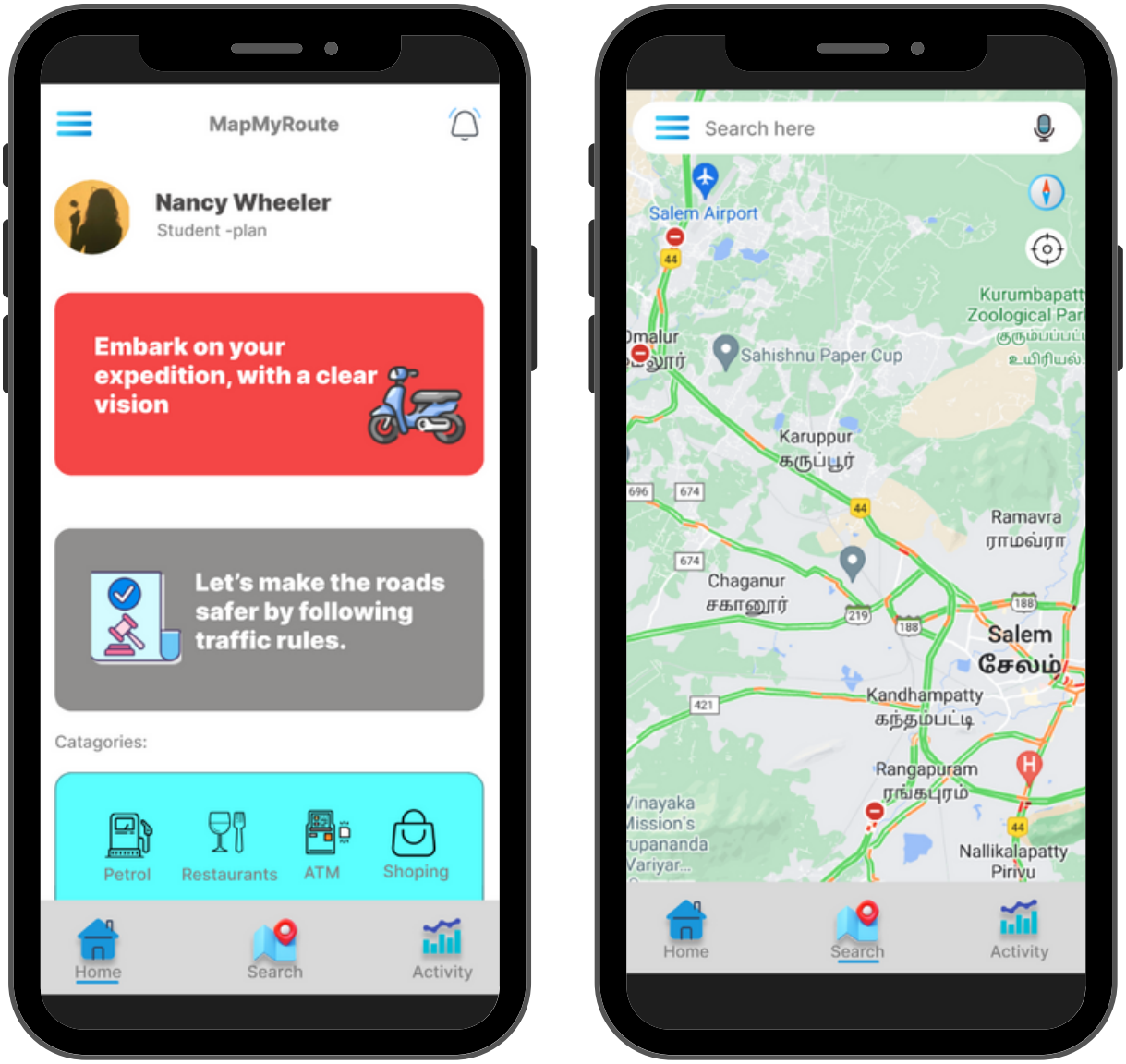
Firebase : The current and Historical Traffic data are stored in Firebase to carry the traffic information to the central servers like GCP , Heroku etc.

Heroku : The deployed machine learning model in the Heroku server retrieves the traffic data from Firebase to perform machine learning operations and route prediction activity and send required data to the API

The Following Screenshot represents the User interface of our app Home screen and Traffic map screen

THE APP AND RELEVANT INFORMATION ABOUT THE RUNNING PROCESSES ARE SHARES IN THE PHSAE_4 REPOSITORY.

APP NAME : **Map My Route** Fig : Home Screen and Traffic Map Screen as follows.



- The app is made by using Android Studio with Kotlin as its primary functioning language And used XML Layouting for its user interface.
- Using Google Map API, and Prediction algorithm the app can provide traffic congestion on the map.
- In addition to that user can able learn traffic rules and guide lines.

CONCLUSION:

Offering real-time data through a user-friendly interface facilitates swift customer engagement, alleviating congestion, lowering road accidents, and mitigating zone freezing.