Overcrowding Module User Guide

Introduction

The Overcrowding Module is designed to track user movement in real-time and predict potential overcrowding in specific areas. This module calculates various features like distance, speed, direction, acceleration, frequent areas, and more. It uses machine learning techniques to predict future locations and can be applied to analyze traffic patterns or predict busy zones.

Prerequisites

* Python 3.5 or higher
* Required Libraries: pandas, scikit-learn, geopy, numpy, glob, os, TensorFlow (2.0 or higher)
* Trajectory Data in **.plt** format

Components

**1. RealTimeTracking Class**

This class contains methods to process and analyze GPS data.

**Key Methods**

* **get\_trajectory(gps\_data)**: Returns the trajectory for the specified user.
* **get\_distance(point1, point2)**, **get\_speed(...)**, **get\_acceleration(...)**: Calculate distance, speed, and acceleration between points.
* **get\_frequent\_areas(trajectory)**: Returns clusters representing frequent areas the user has visited.
* **train\_personalised\_model(trajectory\_data)**: Trains a personalized predictive model for the user.
* **test\_prediction(trajectory)**: Tests the predictive model on a given trajectory.
* **predict\_traffic(gps\_data)**: Predicts traffic in frequent areas (requires further development).

How to Use

1. **Read Data**: Use the **read\_plt(file\_path, user\_id)** function to read trajectory data from **.plt** files.
2. **Initialize RealTimeTracking**: Create an instance of the **RealTimeTracking** class with the user ID you want to track.

real\_time\_tracking = RealTimeTracking(user\_id)

1. **Train the Model**: Loop through the trajectory files and train the model.

for file\_path in glob.glob(directory\_path): trajectory\_data = read\_plt(file\_path, user\_id) real\_time\_tracking.train\_personalised\_model(trajectory\_data, retrain=False)

1. **Test the Model**: You can test the model using the **test\_prediction(trajectory)** method.

test\_data = read\_plt(test\_file\_path, user\_id) trajectory\_to\_predict = real\_time\_tracking.get\_trajectory(test\_data) real\_time\_tracking.test\_prediction(trajectory\_to\_predict)

1. **Predict Traffic**: You can extend and use the **predict\_traffic** method to predict traffic patterns in common areas.

**2. PredictiveTracking Class**

This class contains the methods to load data, load and save the model. It also contains the model architecture and all the model parameters. If you want to make model specific adjustments

Conclusion

The Overcrowding Module provides a powerful toolkit for tracking user movements and predicting future locations. It can be used for various applications like traffic analysis, urban planning, and overcrowding prediction. By customizing the model and utilizing various features, users can adapt the module to specific requirements and datasets.