# Smart City: Traffic prediction using Deep Learning

Leena AlQasem Randa Almohammadi Banan Alhethlool

Leenabdulh@gmail.com randa1414@gmail.com banan.alhethlool@gmail.com

## Abstract:

The purpose of this project is to use a traffic dataset and apply Deep Learning algorithms to it. This project can be used to help urban cities, especially smart cities, have a better understanding of traffic patterns. This leads to better traffic management, which is getting more and more popular nowadays. It can also be used to ensure the smart vehicle drivers get their route without further delay or traffic wait time, solving many problems. Predicting traffic can be used for infrastructure and safety improvements. The prediction of traffic can be implemented in many cities, including smart cities, in many ways that will improve human life.

## Design:

The deep learning project's progress will follow this way. First, the dataset that is used in this project is the Traffic Prediction Dataset. It was downloaded from Kaggle.com. For better exploration, we did some exploratory data analysis (EDA). Then, we did some feature engineering to improve the features, such as dropping the unnecessary feature and formatting the datetime, and scaling for data normalization, such as differencing and normalization. Then we use a deep learning model called Gated Recurrent Unit (GRU), which is a developed model of RNN. Moreover, the model runs 35 epochs, models 4 junctions, and finds a Root Mean Square Error (RMSE) for each junction.

## Dataset:

The dataset for this project consists of datetime, vehicles, and junction information. The data is collected from sensors at every junction. It has over 48, 000 observations and was obtained as an open source from Kaggle, Here.

## Methodology:

First, since the dataset has four junctions, we treated the junctions separately for modeling. Then, we split the data into training with 85% and testing with 15%, because we want to train the dataset as much as possible. Then, the chosen model for this project is the Gated Recurrent Unit (GRU). Below, we will highlight the important features:

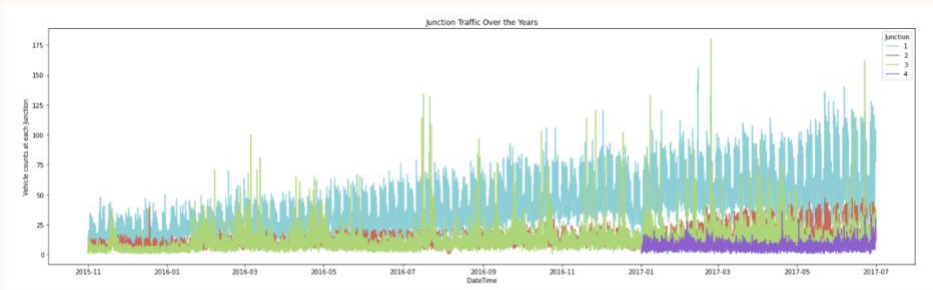
### Gated Recurrent Unit (GRU):

- Construct six layers.
- Use tanh as the activation function.
- For performance optimization, we used stochastic gradient descent.

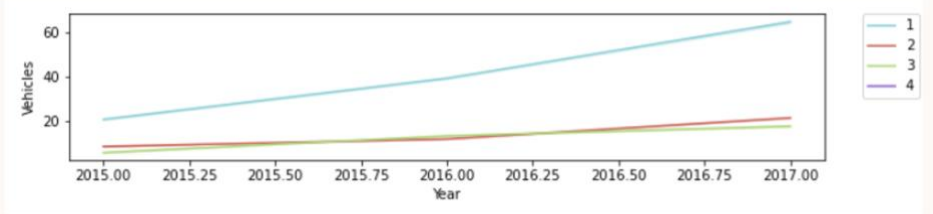
## Tools:

- **Technologies:** Python, Jupyter Notebook.
- **Libraries:** Pandas, NumPy, matplotlib, Seaborn, scikit-learn, Keras, and TensorFlow.

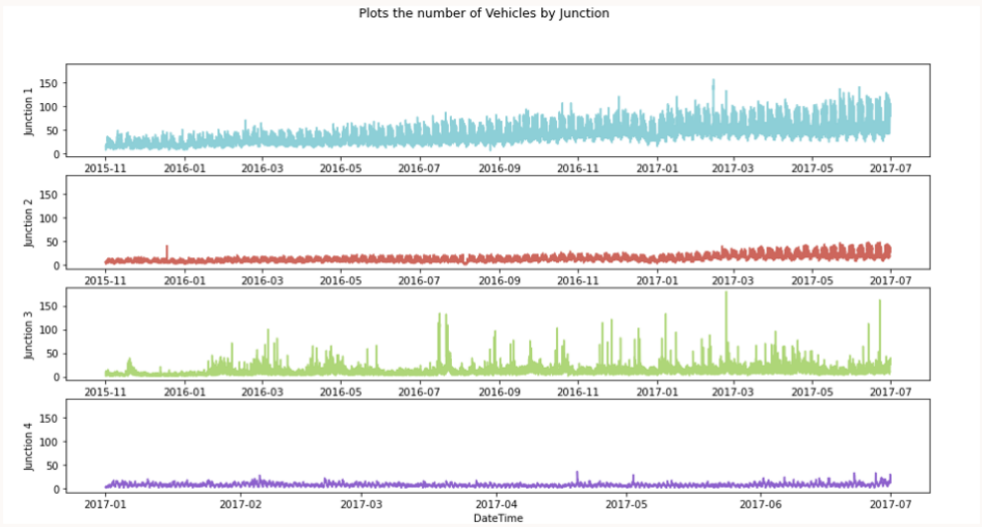
## Communication:




Traffic increases over the year.



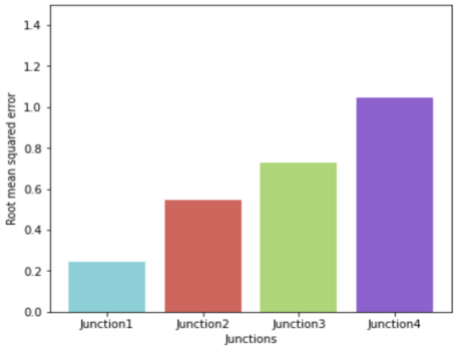

All three junctions have an upward trend except the fourth.




Timeseries of traffic for each junction over the year.

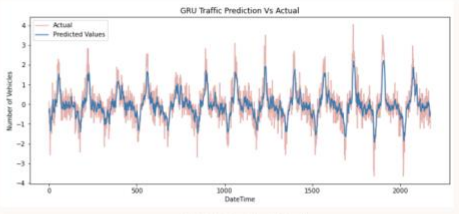
**Epochs** **35**

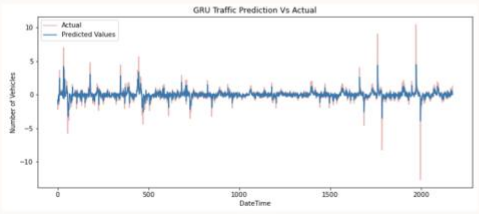
| JUNCTIONS | RMSE |
|---|---|
| Junction 1 | 0.2443 |
| Junction 2 | 0.5470 |
| Junction 3 | 0.7337 |
| Junction 4 | 1.0391 |

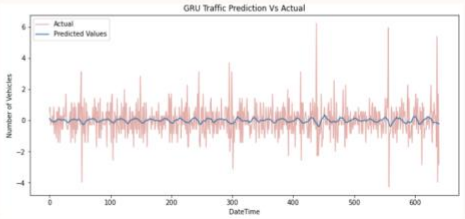



Junction 1


Junction 2




Junction 3

Junction 4