Traffic Management

Team member

920321205303 - OJAS.M.S

Phase 4 – Development part 2

Project title: Traffic management

**Trafic management**

**Introduction**:

Traffic management using IoT (Internet of Things) is a modern approach to enhance transportation systems and improve the overall efficiency and safety of traffic flow. IoT technology involves connecting various devices and sensors to the internet, allowing them to collect and exchange data in real-time. In the context of traffic management, IoT plays a crucial role by enabling the monitoring, analysis, and control of traffic conditions. Here's an overall introduction to traffic management using IoT:

Sensors and Data Collection: IoT devices, such as traffic cameras, smart traffic lights, and vehicle sensors, are strategically placed throughout road networks. These devices collect data on traffic volume, vehicle speed, and road conditions, which is then transmitted over the internet.

• **Data Analytics**: Collected data is processed and analyzed in real-time. Advanced analytics tools can identify traffic patterns, congestion points, and potential safety hazards. This information is valuable for traffic management authorities to make informed decisions.

• **Traffic Optimization**: IoT-enabled traffic lights and dynamic signage can adjust in response to real-time traffic data. This helps optimize traffic flow by reducing congestion and minimizing wait times.

• **Predictive Maintenance**: IoT can monitor the health of infrastructure components like bridges, roads, and traffic signals. Predictive maintenance can be scheduled based on data from sensors, ensuring the longevity and safety of these assets.

• **Emergency Response**: IoT technology can also improve emergency response times by providing real-time updates on accidents or road incidents. This helps authorities deploy resources more efficiently.

• **Smart Parking**: IoT-based solutions enable drivers to find available parking spaces through mobile apps, reducing the time spent searching for parking and lowering traffic congestion.

• **Environmental Impact**: By managing traffic more efficiently, IoT can reduce fuel consumption, emissions, and overall environmental impact.

•**Public Transportation Enhancement:** IoT can provide real-time information on public transportation schedules and routes, making it easier for commuters to plan their journeys.

• **Data Accessibility**: Many cities make traffic data available to the public and developers, which can lead to the creation of innovative apps and services that further enhance the transportation experience.

**1 - Problem Definition and Design Thinking**

Traffic management refers to the coordinated planning, control, and optimization of traffic flow on roads, streets, and highways to ensure safe and efficient transportation of people and goods. It involves a complex system of policies, strategies, and technologies aimed at reducing congestion, enhancing safety, and improving overall mobility. Here's a detailed explanation of traffic management:

1. \*\*Traffic Flow Analysis:\*\*

- Traffic management begins with the collection of data about traffic patterns, volumes, and congestion levels. This data is obtained through various sources like traffic cameras, sensors embedded in roadways, and surveys.

2. \*\*Traffic Engineering and Design:\*\*

- Traffic engineers use collected data to design and optimize road networks. This includes determining road widths, lane configurations, signal placements, and road geometries to accommodate current and future traffic needs.

3. \*\*Traffic Signals and Signage:\*\*

- Traffic signals and road signs are strategically placed to guide drivers, regulate intersections, and provide essential information. They play a crucial role in managing traffic by controlling the right of way and informing drivers about speed limits, turns, and hazards.

4. \*\*Public Transit Integration:\*\*

- Efficient traffic management often involves integrating public transit systems like buses and trains into the overall transportation network. This encourages the use of public transport, reducing the number of private vehicles on the road.

5. \*\*Traffic Control Devices:\*\*

- These include devices like traffic lights, stop signs, speed bumps, and roundabouts that are placed at key locations to regulate and calm traffic, improving safety and reducing congestion.

6. \*\*Intelligent Transportation Systems (ITS):\*\*

- ITS uses technology, such as traffic cameras, variable message signs, and GPS, to monitor and manage traffic in real-time. These systems provide data to traffic control centers, allowing for dynamic adjustments to traffic flow and the rapid response to incidents.

7. \*\*Traffic Signal Synchronization:\*\*

- Traffic lights are synchronized to create “green waves” along major routes, reducing stops and starts, and improving traffic flow. This reduces fuel consumption and emissions.

8. \*\*Traffic Management during Special Events or Incidents:\*\*

- Traffic management plans are developed for special events like parades, sports games, and emergencies. These plans may involve road closures, rerouting, or deploying additional personnel to handle increased traffic.

9. \*\*Congestion Pricing:\*\*

- Some cities implement congestion pricing, where drivers are charged a fee for using certain roads or areas during peak traffic times. This discourages unnecessary travel during congested periods and generates revenue for transportation improvements.

10. \*\*Traffic Education and Enforcement:\*\*

- Public awareness campaigns and law enforcement efforts help educate drivers about safe and responsible behavior on the road. Enforcement of traffic rules, such as speed limits and seatbelt use, contributes to overall safety.

11. \*\*Data Analytics and Prediction:\*\*

- Advanced data analytics and machine learning are used to predict traffic patterns, congestion hotspots, and potential incidents. This allows for proactive traffic management strategies.

12. \*\*Environmental Considerations:\*\*

- Traffic management also considers environmental impact. Strategies may include promoting alternative transportation methods like cycling or walking and encouraging the use of electric vehicles.

In summary, traffic management is a multifaceted approach that combines engineering, technology, policy, and public awareness to ensure safe and efficient transportation in urban and suburban areas. It aims to alleviate congestion, reduce travel times, enhance safety, and minimize the environmental impact of transportation systems.

**2 - Innovation**

Designing a traffic management system involves various components, such as road layout, signals, signage, and technology. Here's a high-level overview:

1. \*\*Road Layout\*\*: Plan the road network to accommodate traffic flow efficiently. Consider factors like lane width, intersections, and pedestrian crossings.

2. \*\*Traffic Signals\*\*: Install traffic lights at intersections to regulate the flow of vehicles. Use sensors or timers to optimize signal timings.

3. \*\*Signage\*\*: Place clear and standardized road signs to guide drivers, including speed limits, stop signs, and directional signs.

4. \*\*Markings\*\*: Use road markings, like lane dividers, crosswalks, and arrows, to direct traffic and enhance safety.

5. \*\*Technology\*\*: Implement traffic management technology, such as CCTV cameras and sensors, to monitor traffic conditions in real-time.

6. \*\*Traffic Management Center\*\*: Establish a control center to collect data and manage traffic. This center can adjust signal timings, respond to accidents, and coordinate emergency services.

7. \*\*Public Transportation\*\*: Promote public transportation options like buses and trains to reduce the number of private vehicles on the road.

8. \*\*Pedestrian and Cyclist Facilities\*\*: Design sidewalks, bike lanes, and pedestrian crossings to prioritize safety for non-motorized road users.

9. \*\*Smart Traffic Solutions\*\*: Explore smart solutions like adaptive traffic signals that adjust based on real-time traffic conditions.

10. \*\*Traffic Education and Enforcement\*\*: Educate drivers about traffic rules and enforce them through law enforcement agencies.

11. \*\*Environmental Considerations\*\*: Promote eco-friendly transportation options and reduce emissions through measures like carpool lanes or electric vehicle charging stations.

12. \*\*Emergency Response Plans\*\*: Develop plans for handling accidents and emergencies to minimize disruptions and ensure safety.

13. \*\*Data Analysis\*\*: Continuously collect and analyze traffic data to identify congestion points and areas that need improvement.

**3 - Development Part 1**

Introduction

Nowadays, the accelerated growth of the population and, consequently, the number of vehicles in the cities added to the technological limitations of traffic control signs have made vehicular traffic one of the problems of modern life. This problem has negative consequences for the environment, health and the economy.

Here is where the concept of Intelligent Transportation System (ITS) comes in, as a critical component of smart city infrastructure [[**1**](https://www.mdpi.com/2227-7080/10/1/5#B1-technologies-10-00005)]. Using big data information and communication technology [[**2**](https://www.mdpi.com/2227-7080/10/1/5#B2-technologies-10-00005)], ITS can provide real-time road infrastructure analysis and more efficient traffic control [[**3**](https://www.mdpi.com/2227-7080/10/1/5#B3-technologies-10-00005),[**4**](https://www.mdpi.com/2227-7080/10/1/5#B4-technologies-10-00005)]. This system relies on traffic predictions as a critical component [[**5**](https://www.mdpi.com/2227-7080/10/1/5#B5-technologies-10-00005),[**6**](https://www.mdpi.com/2227-7080/10/1/5#B6-technologies-10-00005)]. The purpose of traffic forecasting is to predict future traffic conditions on a transportation network based on historical observations [[**7**](https://www.mdpi.com/2227-7080/10/1/5#B7-technologies-10-00005)]. This data can be helpful in ITS applications such as traffic congestion control and traffic light control [[**8**](https://www.mdpi.com/2227-7080/10/1/5#B8-technologies-10-00005)]. For example, it can calculate the likelihood of congestion on the corresponding road segment and prepare for it in advance [[**9**](https://www.mdpi.com/2227-7080/10/1/5#B9-technologies-10-00005)].

Traffic prediction can be divided into two types of techniques: parametric, including stochastic and temporal methods, and non-parametric, such as machine-learning (ML) models [[**10**](https://www.mdpi.com/2227-7080/10/1/5#B10-technologies-10-00005)], recently used to solve complex traffic problems. The review made in [[**11**](https://www.mdpi.com/2227-7080/10/1/5#B11-technologies-10-00005)] found that non-parametric algorithms outperform parametric algorithms due to their ability to deal with a large number of parameters in massive data. In [[**12**](https://www.mdpi.com/2227-7080/10/1/5#B12-technologies-10-00005)] five ML algorithms are evaluated to forecast the total volume of the traffic flow in Porto city; the algorithms are: Linear Regression, Sequential Minimal Optimization (SMO) Regression, Multilayer Perceptron (MLP), M5P model tree and Random Forest (RF). The experimental results show that the M5P regression tree outperforms the other regression models. The authors in [[**13**](https://www.mdpi.com/2227-7080/10/1/5#B13-technologies-10-00005)] reported some multi-model ML methods for traffic flow estimation from floating car data. In particular, they evaluated the capacity of Gaussian Process Regressor (GPR) to address this issue. Deep learning (DL) as a subset of ML uses multilayered neural networks exposed to many data to train themselves. This capability of DL models to extract knowledge from complex systems has made them a robust and viable solution in the field of ITS [[**14**](https://www.mdpi.com/2227-7080/10/1/5#B14-technologies-10-00005)].

A Multilayer Perceptron Neural Network (MLP-NN) is presented in [[**15**](https://www.mdpi.com/2227-7080/10/1/5#B15-technologies-10-00005),[**16**](https://www.mdpi.com/2227-7080/10/1/5#B16-technologies-10-00005)], this last with a mutual information technique to forecast traffic flow. The simulations showed a decrease in forecast error in comparison to the results of the mean and Autoregressive Integrated Moving Average (ARIMA) models that used traffic data from previous periods. Back-Propagation Neural Network (BPNN) is one of the most typical architectures of Neural Networks and is widely used in many prediction and classification tasks. In [[**17**](https://www.mdpi.com/2227-7080/10/1/5#B17-technologies-10-00005)] an urban traffic signal control system based on traffic flow prediction using BPNN is proposed. Also [[**18**](https://www.mdpi.com/2227-7080/10/1/5#B18-technologies-10-00005)] used BPNN to predict future traffic volumes in the design of a traffic light control system along with a genetic algorithm for timing optimization. With this method the average waiting rate is reduced by almost 30 percent compared with the fixed-time traffic light control system. Following this method, the combination of a genetic algorithm and neural network in [[**19**](https://www.mdpi.com/2227-7080/10/1/5#B19-technologies-10-00005)] leads to a named Genetic Neural Network. Ref. [[**20**](https://www.mdpi.com/2227-7080/10/1/5#B20-technologies-10-00005)] presents a deep learning neural network method for optimizing traffic flow and reducing congestion at key intersections by using historical data from all the movements of an intended intersection, with time series and environmental variables as the input features. The output is fed into a delay equation that generates the best green times to manage traffic delay. In [[**21**](https://www.mdpi.com/2227-7080/10/1/5#B21-technologies-10-00005)] a short-term traffic flow prediction based on an improved wavelet neural network (WNN) is proposed. They use an improved particle swarm optimization (IPSO) to avoid being trapped in a local extremum. The outputs of the IPSO are the corresponding wavelet neural network parameters, and experimental results show that this algorithm is more efficient than the WNN and PSO–WNN algorithms alone. The prediction results are more stable and more accurate. Compared with the traditional wavelet neural network, the error is reduced by almost 15 percent.

Recurrent neural networks (RNNs) have an internal state that can represent context information; they hold information about past inputs for a period of time and are typically used to capture dynamic sequences of data. RNNs based DL methods Long Short-Term Memory (LSTM) [[**22**](https://www.mdpi.com/2227-7080/10/1/5#B22-technologies-10-00005)], short-term flow prediction [[**23**](https://www.mdpi.com/2227-7080/10/1/5#B23-technologies-10-00005),[**24**](https://www.mdpi.com/2227-7080/10/1/5#B24-technologies-10-00005)] and Gated Recurrent Units (GRU) in [[**25**](https://www.mdpi.com/2227-7080/10/1/5#B25-technologies-10-00005)] outperform the ARIMA model; additionally, the authors report that this is the first use of GRU in traffic flow prediction. Currently, GRU models continue to be used for the development of intelligent traffic flow prediction [[**26**](https://www.mdpi.com/2227-7080/10/1/5#B26-technologies-10-00005)].

On the other hand, stacked autoencoders (SAEs) are an unsupervised learning method that extracts features from unlabeled data and uses them to train the model. SAEs presented in [[**27**](https://www.mdpi.com/2227-7080/10/1/5#B27-technologies-10-00005)] proved to be more accurate than the Back Propagation Neural Network (BP NN) model, the Random Walk (RW), the Support Vector Machine (SVM), and the Radial Basis Function (RBF) NN model for the short-term prediction of the traffic volume. Other works recently have concentrated on hybrid methods [[**28**](https://www.mdpi.com/2227-7080/10/1/5#B28-technologies-10-00005)]. Ref. [[**3**](https://www.mdpi.com/2227-7080/10/1/5#B3-technologies-10-00005)] reported a new hybrid DL model by using Graph Convolutional Network (GCN) and the deep aggregation structure of GRU; for data preprocessing Moving Average is used along with Data Normalization using MinMax Scaler. A Hybrid Least Square Support Vector Machine (LSSVM) is presented in [[**29**](https://www.mdpi.com/2227-7080/10/1/5#B29-technologies-10-00005)]. To search the optimal parameters of LSSVM, this paper proposes a hybrid optimization algorithm that combines particle swarm optimization (PSO) with a genetic algorithm.

In this paper, five ML models: MLP-NN, Gradient Boosting Regressor, Random Forest Regressor, Linear Regressor and Stochastic Gradient Regressor, and two DL models based on RNNs: GRU and LSTM; are compared in the task of traffic flow prediction of each lane of an intersection, with the purpose of applying them in the modernization of traffic light controllers, allowing a better traffic flow without the need to completely change the traffic light system, making its implementation more feasible. The experiments demonstrate that all models have good capability in predicting vehicular flow and can be used in a smart traffic light controller.

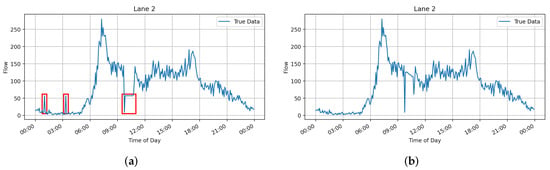
The rest of the paper is organized as follows: [**Section 2**](https://www.mdpi.com/2227-7080/10/1/5#sec2-technologies-10-00005) describes the materials and methods to train the machine-learning models. [**Section 3**](https://www.mdpi.com/2227-7080/10/1/5#sec3-technologies-10-00005) presents the results of several metrics used to evaluate the performance and compare the ML and DL models. [**Section 4**](https://www.mdpi.com/2227-7080/10/1/5#sec4-technologies-10-00005) describes the proposed usage scenario in the real-world. Conclusions and future work are described in [**Section 5**](https://www.mdpi.com/2227-7080/10/1/5#sec5-technologies-10-00005).

## 2. Materials and Methods

In this paper, a Road Traffic Prediction Dataset from Huawei Munich Research Center is used, which is a public dataset for traffic prediction derived from a variety of traffic sensors, i.e., induction loops [[**30**](https://www.mdpi.com/2227-7080/10/1/5#B30-technologies-10-00005)], it is important to note that, at present, there are a few public datasets [[**31**](https://www.mdpi.com/2227-7080/10/1/5#B31-technologies-10-00005)]. The data can be used to forecast traffic patterns and modify stop-light control parameters. The dataset contains recorded data from six crosses in the urban area for 56 days, in the form of flow time series, depicting the number of vehicles passing every five minutes for a whole day, which is recommended for short-term predictions [[**32**](https://www.mdpi.com/2227-7080/10/1/5#B32-technologies-10-00005)]. For this research, four of the six intersections are used to simulate four lanes of an intersection.

#### **2.1. Data Preprocessing**

It is common to find missing values in databases represented by zeros, probably due to sensor failures. The article [[**25**](https://www.mdpi.com/2227-7080/10/1/5#B25-technologies-10-00005)] impute the missing data points using the historical average value, and [[**20**](https://www.mdpi.com/2227-7080/10/1/5#B20-technologies-10-00005)] reports substituting these values with the mean of the entire column containing the missing value. Although these substituted data are not realistic values, the researchers determined that they are better than no data. However, when performing the predictions, this procedure generates spikes in the real values, increasing the error because, on some occasions, the trend shows that the zero values were real. These spikes are highlighted in red rectangles as shown in [**Figure 1**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f001)a; this is why a moving average is applied using the 12 previous readings. [**Figure 1**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f001)b shows how abrupt changes caused by the general average are avoided. Next, the database is split into 75% of the data (42 days) for training and 25% (14 days) for testing.



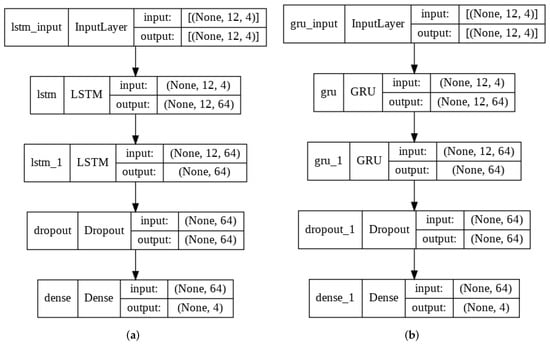
**Figure 1.** Differences between the substitution of zeros applying a general average and a moving average. (**a**) General Average; (**b**) Moving Average.

The data are scaled in a range from 0 to 1, following the standard normal distribution using MinMaxScaler from scikit-learn library [[**33**](https://www.mdpi.com/2227-7080/10/1/5#B33-technologies-10-00005)]. For this experiment, the previous hour’s traffic flow is used, which is a time sequence of 12 data points, to predict the traffic flow coming in the next five minutes. To do this, lists are created grouped into 13 readings; these lists are used for training and testing purposes. The generated lists are then converted into arrays and the training sequence is shuffled. Once this is done, the last column of the arrays is taken as the output ‘Y’, and the remaining columns as the inputs ‘X’.

#### **2.2. Recurrent Neural Networks**

#### **2.2.1. RNNs Design**

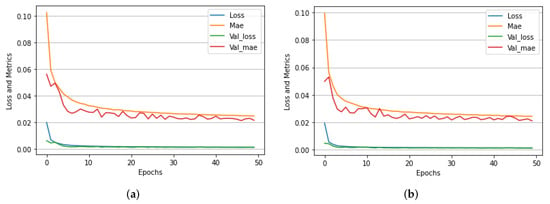
Two recurrent neural networks are designed: GRU and LSTM. Keras library [[**34**](https://www.mdpi.com/2227-7080/10/1/5#B34-technologies-10-00005)] is used to create the models. The architecture is the same for both as shown in [**Figure 2**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f002) and explained below: The input layer with shape equal to the number of time steps per the number of lanes. Then two recurrent layers with 64 neurons, 20% dropout, and finally, an output layer with neurons equal to the number of lanes and sigmoid activation function.



**Figure 2.** Architecture of the recurrent neural networks. (**a**) LSTM-NN; (**b**) GRU-NN.

#### **2.2.2. RNNs Training**

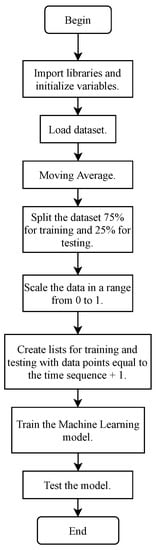
In the compilation of the model, mean squared error (MSE) is used as loss function, and the optimizer is RMSprop from Keras library [[**34**](https://www.mdpi.com/2227-7080/10/1/5#B34-technologies-10-00005)] with default parameters and mean absolute error (MAE) is the metric function. For training, a batch size of 128 and 50 epochs are used, five percent of the training data is used for validation. The experiments are performed in Google Colaboratory [[**35**](https://www.mdpi.com/2227-7080/10/1/5#B35-technologies-10-00005)] along with Weights & Biases [[**36**](https://www.mdpi.com/2227-7080/10/1/5#B36-technologies-10-00005)] for tracking them. [**Figure 3**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f003) shows the training performance of the two architectures, where the loss and evaluation metrics in training and validation can be observed.



**Figure 3.** Training performance of both neural networks. (**a**) LSTM NN; (**b**) GRU NN.

#### **2.3. Machine Learning Methods**

Five regression models from the scikit-learn library [[**33**](https://www.mdpi.com/2227-7080/10/1/5#B33-technologies-10-00005)] in Python are used: Linear Regression, Gradient Boosting Regressor, MultiLayer Perceptron Regressor, Stochastic Gradient Descendent Regressor and Random Forest Regressor, all of them with default parameters and a random state equal to zero for the reproducibility of the experiment. A reshape of ‘X’ for both training and testing is made because these models require a 2D array instead of the 3D used in the RNN’s; after that, the models are fed with the training split. The methodology summary presented in the form of a flowchart is shown in [**Figure 4**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f004).



**Figure 4.** Flowchart of proposed method.

## 3. Results

To evaluate the performance of ML and DL algorithms, first, an inverse scaler was applied to the ’y’ test. Then we relied on the metrics of the scikit-learn library [[**33**](https://www.mdpi.com/2227-7080/10/1/5#B33-technologies-10-00005)], the mean absolute error (MAE) [[**21**](https://www.mdpi.com/2227-7080/10/1/5#B21-technologies-10-00005),[**37**](https://www.mdpi.com/2227-7080/10/1/5#B37-technologies-10-00005)], root mean square error (RMSE) [[**7**](https://www.mdpi.com/2227-7080/10/1/5#B7-technologies-10-00005),[**29**](https://www.mdpi.com/2227-7080/10/1/5#B29-technologies-10-00005)], mean absolute percent error (MAPE) [[**14**](https://www.mdpi.com/2227-7080/10/1/5#B14-technologies-10-00005),[**15**](https://www.mdpi.com/2227-7080/10/1/5#B15-technologies-10-00005)], R-squared (2�2) [[**3**](https://www.mdpi.com/2227-7080/10/1/5#B3-technologies-10-00005),[**9**](https://www.mdpi.com/2227-7080/10/1/5#B9-technologies-10-00005)] and explained variance (EV) are used. They are defined as:

MAE(,̂)=1∑=0−1|−̂|MAE(�,�^)=1�∑�=0�−1��−�^�

(1)

MAPE(,̂)=100%∑=0−1|−̂|MAPE(�,�^)=100%�∑�=0�−1��−�^���

(2)

RMSE(,̂)=⎡⎣⎢1∑=0−1(−̂)2⎤⎦⎥12RMSE(�,�^)=1�∑�=0�−1��−�^�212

(3)

2(,̂)=1−∑=1(−̂)2∑=1(−̲)2�2(�,�^)=1−∑�=1���−�^�2∑�=1���−�¯2

(4)

explained\_variance(,̂)=1−[−̂][]explained\_variance(�,�^)=1−����−�^����

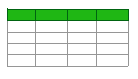
(5)

where *n* is the number of samples, *y* is the observed traffic flow, ̂�^ is the predicted traffic flow and ̲�¯ is the mean.

MAE and RMSE measure absolute prediction errors, and MAPE measures relative prediction errors. Smaller numbers indicate higher prediction performance for these three metrics [[**37**](https://www.mdpi.com/2227-7080/10/1/5#B37-technologies-10-00005)]. The values of 2�2 and EV range from zero to one, and the closer to the value of one the better the regression model fits.

[**Table 1**](https://www.mdpi.com/2227-7080/10/1/5#table_body_display_technologies-10-00005-t001) lists the performance metrics of each ML and DL models, Multilayer Perceptron and Gradient Boosting obtained R-Squared and Explained Variance above 0.93, MAE of 10.8, MAPE of 21% and RMSE of 15.4. In contrast, Random Forest had an R-squared and Explained Variance slightly below 0.93, MAE of 10.88, MAPE of 21% and RMSE of 15.5. While GRU and LSTM obtained R-Squared and Explained Variance near 0.92, MAE of 10.88, MAPE of 22% and RMSE of 15.6, Linear Regression R-Squared and Explained Variance were 0.926, MAE of 11.2, MAPE of 24% and RMSE of 15.85; finally, Stochastic Gradient had R-Squared and Explained Variance of 0.9, MAE of 12.8, MAPE of 29% and RMSE of 18.

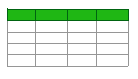
**Table 1.** Comparison of performance metrics using the first dataset [[**30**](https://www.mdpi.com/2227-7080/10/1/5#B30-technologies-10-00005)].



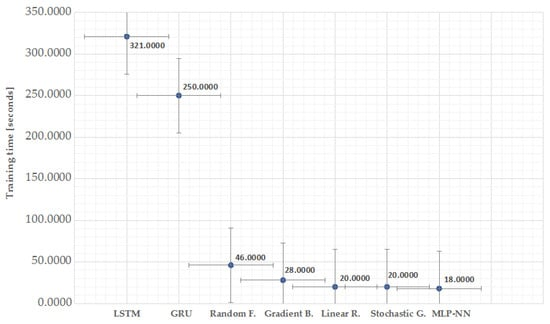
For the results shown, RNNs are iteratively trained ten times, and the average of each metric is calculated; in the case of the ML models (scikit-learn), the random state allows us to have the same results each time.

Additionally, robustness testing was performed using a different dataset [[**25**](https://www.mdpi.com/2227-7080/10/1/5#B25-technologies-10-00005)] than the one initially used for training and validation. These new data are collected from the PeMS dataset, which has over 15,000 sensors deployed throughout the state of California, specifically the fourth district, which lies in the Bay Area, Alameda, Oakland of the U.S. For robustness, 2�2 and EV score are good parameters because these metrics are dimensionless, work for different datasets of different scales and are normalized. The metrics obtained with the external dataset are listed in [**Table 2**](https://www.mdpi.com/2227-7080/10/1/5#table_body_display_technologies-10-00005-t002). Comparing the results of 2�2 and EV score listed in [**Table 1**](https://www.mdpi.com/2227-7080/10/1/5#table_body_display_technologies-10-00005-t001) and [**Table 2**](https://www.mdpi.com/2227-7080/10/1/5#table_body_display_technologies-10-00005-t002), it can be observed that in both cases 2�2 and EV score are between the values of 0.9 and 0.95. Therefore, since 2�2 and EV remain within the same range regardless of the dataset, we can confirm that the proposed models are robust for traffic flow prediction.

**Table 2.** Performance metrics using the second dataset (PeMS) [[**25**](https://www.mdpi.com/2227-7080/10/1/5#B25-technologies-10-00005)].

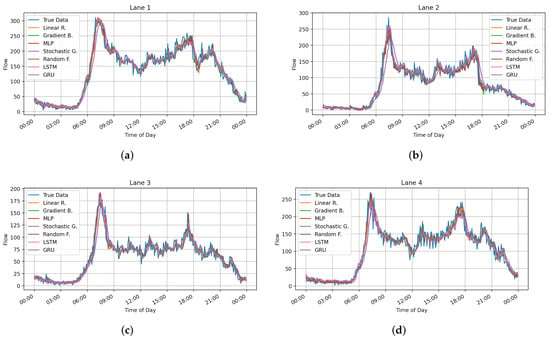


On the other hand, to analyze the cost-benefit of the implementation of the models, the average training time was obtained for each one of them, the experiments were carried out in the Google Colaboratory [[**35**](https://www.mdpi.com/2227-7080/10/1/5#B35-technologies-10-00005)] execution environment, and also [[**36**](https://www.mdpi.com/2227-7080/10/1/5#B36-technologies-10-00005)] was used to keep track of the times. [**Figure 5**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f005) depicts the training time of the seven ML models tested in this study. It is important to note that scikit learn models train takes less time than designed RNN’s. LSTM and GRU models are the ones with the longer training time 321 and 250 seconds, among the scikit learn models Random Forest, Gradient Boosting, Linear Regression, Stochastic Gradient and MLP-NN 46, 28, 20, 20 and 18 seconds being MLP-NN the fastest and it is the model that also has better performance metrics, as shown in [**Table 1**](https://www.mdpi.com/2227-7080/10/1/5#table_body_display_technologies-10-00005-t001).

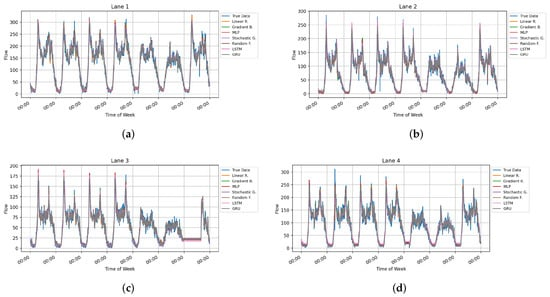


**Figure 5.** Comparison of training times.

Predictions performed over the test split for one day in the four lanes are shown in [**Figure 6**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f006) and for an entire week are plotted in [**Figure 7**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f007).



**Figure 6.** Comparison of traffic flow prediction models for one test day. (**a**) Traffic flow prediction of lane 1; (**b**) Traffic flow prediction lane 2; (**c**) Traffic flow prediction lane 3; (**d**) Traffic flow prediction lane 4.

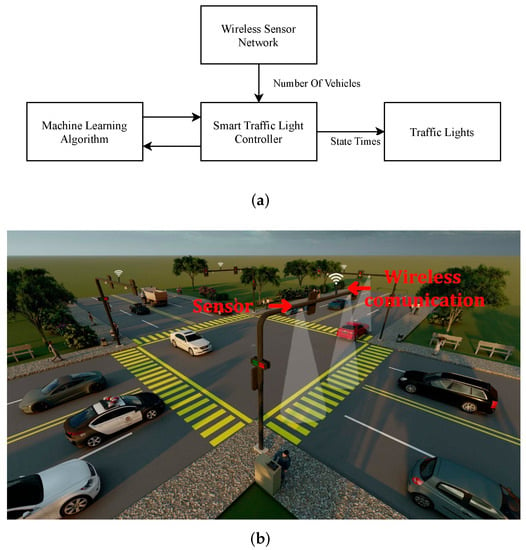


**Figure 7.** Comparison of traffic flow prediction models over an entire week. (**a**) Traffic flow prediction of lane 1; (**b**) Traffic flow prediction lane 2; (**c**) Traffic flow prediction lane 3; (**d**) Traffic flow prediction lane 4.

## 4. Proposed Usage Scenario

These models can be used in a smart traffic light controller, fed by traffic sensors that count the number of vehicles passing through a lane every certain period; with these readings, a database similar to the one used in this paper can be created. Once the database is generated, the ML model can be trained for each intersection. Then the traffic flow for the next period can be predicted by using a given number of past readings.

Once the prediction is made, it will allow better programming of the times of each state, either manually by an operator or automatically using an algorithm to calculate the optimal times of the traffic light states. The whole process can be carried out by wirelessly communicating the traffic light with a central station or at the controller itself. [**Figure 8**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f008)a shows a block diagram of the main elements of the proposed system and [**Figure 8**](https://www.mdpi.com/2227-7080/10/1/5#fig_body_display_technologies-10-00005-f008)b its representation in a real-world scenario.



**Figure 8.** Proposed usage. (**a**) Block diagram; (**b**) Representation in real-world scenario.

## 5. Conclusions

In this paper, we proposed several ML and DL models for the traffic flow prediction at an intersection of vehicular traffic, thus laying the groundwork for adaptive traffic control. Two public datasets were used to train, validate and test the proposed models. Experimental results showed that Multilayer Perceptron Regressor has better performance and takes less processing time to train (18 s). Gradient Boosting Regressor has a similar performance but takes more processing time (28 s). Both RNNs and Random Forest Regressor have a similar score. However, RNNs are slow to train (between 250 and 321 s). Finally, Linear Regression and Stochastic Gradient Regressor have good processing time (20 s) but are the worst performance between these models. All ML and DL models achieved an explained variance score (EV Score) and R-squared (2�2) greater than 0.90, MAE near to 10; the RMSE is near 15 and the MAPE is between 20 and 30 percent. Actually, the performance of these seven algorithms does not differ

**4 - Development Part 2**

ItroductioInn :

Traffic management refers to the process of controlling and regulating the flow of vehicles, pedestrians, and other modes of transportation on roadways and in urban areas. It involves various strategies and systems to ensure the safe and efficient movement of people and goods. This can include traffic signals, road signs, lane markings, speed limits, public transportation systems, and the use of technology to monitor and manage traffic congestion. Effective traffic management is essential for reducing accidents, minimizing congestion, and improving the overall quality of transportation networks in cities and regions.

Create a platform to display traffic management:

<html>

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Traffic Management Platform</title>

<link rel=”stylesheet” href=”styles.css”>

</head>

<body>

<header>

<h1>Traffic Management Dashboard</h1>

</header>

<nav>

<!—Add navigation links here (

</nav>

<section id=”map”>

<!—Add a map or traffic data visualization here (

</section>

<section id=”information”>

<!—Display traffic information, statistics, or alerts here (

</section>

<footer>

<p>&copy; 2023 Traffic Management Platform</p>

</footer>

</body>

</html>

Java script (script.js):

Const canvas = document.getElementById(‘trafficCanvas’);

Const ctx = canvas.getContext(‘2d’);

Let isTrafficLightsOn = false;

Function drawTrafficLights() {

// Draw traffic lights on the canvas

// Implement logic to draw red, green, and yellow lights

}

Function startTrafficLights() {

isTrafficLightsOn = true;

manageTrafficLights();

}

Function stopTrafficLights() {

isTrafficLightsOn = false;

}

Function manageTrafficLights() {

If (!isTrafficLightsOn) {

Return;

}

// Implement logic to control traffic lights

// Change lights in a timed manner (e.g., red for 20 seconds, green for 30 seconds, yellow for 5 seconds)

// Use setTimeout() to manage the timing of light changes

// Red light

drawTrafficLights(‘red’);

setTimeout(() => {

// Green light

drawTrafficLights(‘green’);

setTimeout(() => {

// Yellow light

drawTrafficLights(‘yellow’);

setTimeout(manageTrafficLights, 5000); // Repeat after 5 seconds

}, 30000); // Yellow light for 30 seconds

}, 20000); // Green light for 20 seconds

}

// Initial setup

drawTrafficLights(‘red’); // Initial state: red light

Traffic Management server:

A traffic management server is a centralized system used to monitor and control traffic in various applications, such as computer networks, transportation systems, or smart cities. In the context of computer networks, it can be used to optimize data flow, allocate bandwidth, and ensure Quality of Service (QoS) for network users. In transportation systems, it helps regulate traffic flow, control traffic lights, and manage congestion. In smart cities, it can integrate data from various sources to make informed decisions about traffic management. The specific functions and capabilities can vary depending on the application.

Design platform to receive and display traffic management data from IoT sensors involves several components and considerations.

Here’s a high-level overview:

Components:

Designing a platform to receive and display traffic management data from IoT sensors involves several components and considerations. Here’s a high-level overview:

\*\*IoT Sensors\*\*: You’ll need a network of IoT sensors deployed at strategic locations to collect traffic data. These sensors could include cameras, lidar, ultrasonic sensors, and more.

2. \*\*Data Collection Hub\*\*:

- Sensors transmit data to a data collection hub. This hub can be a cloud-based server, an on-premises server, or a combination of both.

- Use IoT communication protocols like MQTT, CoAP, or HTTP to transmit sensor data to the hub securely.

3. \*\*Data Ingestion\*\*:

- Implement data ingestion mechanisms to receive, validate, and store data from the sensors. This can be done through APIs or dedicated IoT platforms.

- Ensure data integrity and security during the ingestion process.

4. \*\*Data Storage\*\*:

- Store the incoming data in a scalable and secure database system. Options include SQL databases (e.g., PostgreSQL) or NoSQL databases (e.g., MongoDB).

- Consider a time-series database for handling timestamped data efficiently.

5. \*\*Data Processing\*\*:

- Implement data processing pipelines to clean, transform, and enrich the data. This may involve real-time or batch processing, depending on your requirements.

- Analyze the data to extract meaningful insights, such as traffic flow, congestion, and anomalies.

6. \*\*User Interface\*\*:

- Develop a web-based or mobile user interface for users to access and visualize the traffic data.

- Consider responsive design for various devices and browsers.

- Implement role-based access control to ensure data security.

7. \*\*Real-Time Monitoring\*\*:

- Enable real-time monitoring and visualization of traffic data using charts, maps, and dashboards.

- Use technologies like WebSocket for real-time updates.

8. \*\*Alerting System\*\*:

- Implement an alerting system that notifies stakeholders when predefined thresholds or anomalies are detected.

- This can be integrated with email, SMS, or other communication channels.

9. \*\*Data Analytics\*\*:

- Apply machine learning and data analytics to predict traffic patterns, optimize signal timings, and improve traffic management.

- Implement data analytics tools and libraries for these tasks.

10. \*\*Integration\*\*:

- Integrate with other systems and services, such as traffic signal control systems, emergency services, or city management platforms.

11. \*\*Scalability and Redundancy\*\*:

- Design the platform to be scalable, both in terms of handling more sensors and accommodating increased data volume.

- Implement redundancy and failover mechanisms for high availability.

12. \*\*Security\*\*:

- Prioritize data security by using encryption, access controls, and security best practices to protect data both in transit and at rest.

13. \*\*Compliance\*\*:

- Ensure that the platform complies with relevant data protection and privacy regulations, such as GDPR or HIPAA, depending on your location and use case.

14. \*\*Documentation and Training\*\*:

- Provide thorough documentation for users and administrators.

- Offer training for those who will use or manage the platform.

15. \*\*Maintenance and Updates\*\*:

- Establish a plan for ongoing maintenance, updates, and system monitoring to address issues and incorporate new features.

16. \*\*Feedback Mechanism\*\*:

- Create a feedback mechanism for users to report issues or suggest improvements to the platform.

Remember that this is a complex project, and you may need a team of developers, data scientists, and domain experts to implement it successfully. Additionally, the choice of specific technologies and platforms will depend on your project’s requirements and constraints.

**Advantages of Traffic Management:**

• Real-time Data:

IoT devices can collect and transmit real-time traffic data, enabling authorities to make informed decisions and adjustments.

• Reduced Congestion:

Smart traffic management can optimize traffic flow, reducing congestion and travel times.

• Safety Improvements:

IoT sensors can monitor road conditions and provide alerts for potential hazards, enhancing road safety.

• Environmental Benefits:

Traffic optimization can reduce fuel consumption and emissions, promoting environmental sustainability.

• Cost Efficiency:

Efficient traffic management can lead to cost savings in fuel and maintenance

**Disadvantages of Traffic Management:**

• Privacy Concerns:

Collecting and using data from IoT devices may raise privacy issues, as it involves tracking and monitoring individuals' movements.

• Security Risks:

IoT systems can be vulnerable to cyberattacks, potentially disrupting traffic management and safety systems.

• Implementation Costs:

Setting up a comprehensive IoT-based traffic management system can be expensive, including the cost of sensors and infrastructure.

• Maintenance and Upkeep:

IoT devices require regular maintenance and updates to remain effective, adding to ongoing operational costs.

• Data Management:

Managing and processing the vast amount of data generated by IoT devices can be challenging and requires robust infrastructure

**Benifits of traffic management using IOT :**

**1 .**Real-time Data:

IoT sensors and devices provide real-time traffic information, allowing authorities to make data-driven decisions and respond quickly to traffic issues.

**2 .**Traffic Optimization:

IoT enables the optimization of traffic signals, lane management, and routing, reducing congestion and improving traffic flow.

**3 .**Safety Improvements:

IoT sensors can monitor road conditions, detect accidents, and provide alerts, enhancing road safety for drivers and pedestrians.

**4 .**Environmental Impact:

IoT-based traffic management can reduce fuel consumption and emissions by minimizing traffic jams and congestion.

**5 .**Cost Savings:

Efficient traffic management can lead to cost savings in terms of fuel, maintenance, and infrastructure usage.

**6 .**Enhanced Public Transportation:

IoT can improve public transportation systems, making them more reliable and efficient, encouraging the use of public transit.

**7 .**Smart Parking:

IoT can assist drivers in finding available parking spaces, reducing the time spent searching for parking and lowering fuel consumption.

**8 .**Data Analytics:

IoT data can be used for long-term planning, helping authorities make informed decisions about infrastructure improvements and expansions.

**9 .**Better Commuter Experience:

Reduced travel times and improved traffic conditions lead to a more pleasant and efficient commuting experience.

**10 .**Scalability:

IoT traffic management systems can be easily scaled and adapted to accommodate changing traffic patterns and population growth.

**Conclusion**:

In conclusion, traffic management using IoT technology offers a multifaceted approach to improving urban transportation systems. By providing real-time data, optimizing traffic flow, enhancing safety, and reducing environmental impact, IoT-based solutions can significantly benefit both authorities and commuters. Despite potential challenges like privacy concerns, security risks, and implementation costs, the advantages of improved traffic management, cost savings, and enhanced user experience make IoT an invaluable tool for creating more efficient and sustainable urban transportation networks. As technology continues to advance, the potential for further innovation and improvement in traffic management through IoT remains promising.