ISSN 1870-4069

Traffic Prediction Architecture based on Machine Learning Approach for Smart Cities

Marco C. Hernández Inzunza[2], Luis Hernández Robles[3], Miriam A. Carlos Mancilla[1], Emmanuel López Neri[1]

[1] CIIDETEC-Universidad del Valle de México Jalisco, Mexico

[2] Universidad Autónoma de Sinaloa, Departamento de Ciencias en Computación, Mexico

[3] Instituto Tecnológico de Culiacán, Departamento de Ciencias en Computación, Mexico

{marcohdes94i, lu97is}@gmail.com, {miriam.carlos,emmanuel.lopezne}@uvmnet.edu

Abstract. It is common to see more and more people dealing with traffic due to excessive population growth in cities. The traffic has become one of the main topics of interest is used for smart cities approaches. Accordingly, this study presents the development and implementation of a new architecture to predict the traffic flow in a city and the strategy used in this scheme. The proposal considers the use of machine learning, computer vision, deep learning, and neuronal networks to implement the solution. The architecture is composed of four main components; (1) A Machine Algorithm System (*MASY)* that works using pattern recognition of the traffic; (2) A Neuronal Artificial System (*NASY*) helps with the traffic classification; (3) A Web user application (WeUsAP) to present the results, and process entry user data and finally, (4) A Car Counting Wizard (CCW) video capturing component based on computer vision to create a statistical analysis of vehicles. Consequently, some results and comparatives are presented in order to obtain an analysis of data accuracy.

Keywords: Artificial Intelligence, Machine Learning, Deep Learning, Computer Vision, Artificial Neural Networks (ANNs); Vehicular Traffic prediction; Smart Cities; Module architecture.

1 Introduction

In the last decade, cities have shown an accelerated growth of vehicular traffic linked with the increase in population and the subsequent road traffic demand [24] but in same way multiple traffic control strategies have continued introduced with the similar accelerated advance of technology. Although some of these strategies use a proactive approach, is required that quickly adapts to changes in use of the traffic road. Statistically, a person wastes an average of 42 hours a year driving in the traffic during peak hours [1]. The elements used to design an optimal vehicular traffic condition control are vehicular flow capacity of a road and efficient urban transportation planning. Most accidents are the result of a wrong driving decision that negatively affects the traffic condition. Some of the most common variables to consider for traffic control are the fast population growth factor, public transportation network, accessible routes, infrastructure degradation, among others.

In this proposal, machine learning, deep learning, and computer vision are used. The proposal is interested in traffic classification using an Artificial Neural Network (ANN) and the implementation of an architecture that considers four components is presented. First, A Machine Algorithm System (MASY) for pattern recognition, Second, A Neuronal Artificial System (NASY) for data prediction using a machine learning strategy; Third, A Web User Application (WeUsAP) to communicate the results to users; Finally, a Car Counting Wizard (CCW) based on computer vision to create a statistical analysis of vehicles. The proposal does not require GPS.

This paper is organized as follows. Section 2 presents the proposed architecture for traffic predictions; section 3 discusses results and finally, some conclusions are presented.

2 Related Works

Many proposals work with branches of Artificial intelligence (AI) such as machine learning, deep learning, or data analysis, in one effort to solve problems easily and predict future system characteristics. The machine learning approach helps with the design and implementation of a system that learns automatically. For instance, in [20] a comparison between regression models based on a machine learning approach to predict traffic using a virtual network is presented, in which authors describe better accuracy results using real-life data. While in [3] a traffic prediction method using machine learning is presented, the proposal works with LoRa and a traffic predetermined algorithm. The authors do not present any results or proofs.

In [7] the authors proposed five Machine Learning (ML) algorithms for traffic classification based on IP (Internet Protocol). The focus of this proposal is to demonstrate the benefits of computational performance as a significant metric for traffic classification. In [25] the authors present a traffic prediction method for an application. In this work, a statistical method is used, however, it is not intended to create a new methodology or strategy to improve the traffic, only the results of the application are presented.

Deep Learning is a way to automate a prediction analysis. For instance, a deep learning model for traffic prediction is proposed by [14]. The authors present a model that works with local data introduced manually to the system. In addition, a deep belief network is used for processing. Another work presented in [23] proposed data analytics using a machine learning algorithm to provide an adaptive model for traffic predictions. Despite the proposal presents good predictions for traffic, the model spends a lot of energy for monitoring and data processing. In [8] it is presented a binary neuronal network algorithm for traffic flow prediction. The prediction is made by short terms intervals, to visualize how the predicted value can affect the making decisions or actions. The algorithm requires to use a GPS (Global Positioning System) after they send information about places, traffic situations, objects, user preferences, among others. The information is stored and then sent to users. Some definitions for Speed and Traffic are well known in the community, however, to avoid ambiguity, in this proposal flow, and density are defined as follows. *Flow* measures how many cars pass a given point on the road at a given time t. *Density* is defined as the number of cars at one point of any road x at time t. In this proposal, the equation 1 is considered to calculate the traffic flow, presented in [2]:

$$Q=K \, x \, V, \tag{1}$$

where Q represents vehicular flow, K is the vehicular density, and V represents the speed. Other applications use the flow and density to calculate better routes, such as Waze [6] and INRIX [1], however, the accuracy of the predictions is provided by the information of users. Then, users can make decisions about alternative routes and report current traffic situations. In order to control the data and process the incoming information of the environment. According to John McCarthy [4], AI is "the science and engineering of making intelligent machines, especially intelligent computer programs".

For instance, in [9] the authors present an evaluation between flow density and motorway problems using an IA approach, the weather is considered in this proposal. A cross-validation strategy for traffic estimation is presented in [10]. In [11, 12] different strategies for traffic estimation are evaluated, comparisons are made, and the author evaluates the use of a neuronal network.

3 Architecture for Traffic Prediction

The architecture for data prediction proposed is presented in Fig. 1. The architecture does not require GPS to work. The architecture is composed of four main components, MASY, NASY, User App, and the CCW, each one is described in detail.

3.1 Machine Algorithm System (MASY)

MASY is a component designed to predict cars and to obtain an analysis. This module uses a multiple linear regression algorithm optimized with a gradient boost regressor technique for error correction as well-known as a Multiple Linear Regression (MLR) that explains the relationship between one dependent variable and multiple independent variables written in Python.

This technique describes the relationship between days, hours, time, number of cars of every day, and the traffic. MASY works with a preprocessing stage, in which the data is classified in a database made in PostgreSQL. Information from Australian Smart City [13] is used to verify the behavior of car prediction in MASY.

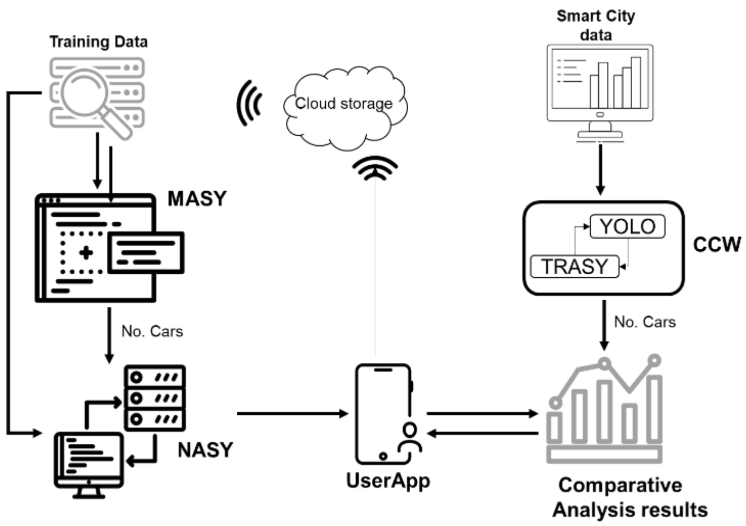

Fig. 1. Proposed architecture for traffic flow prediction.

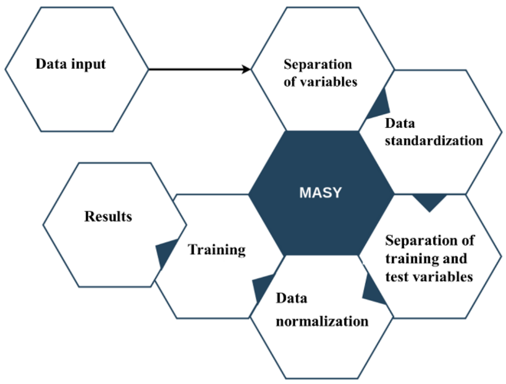

Fig. 2. MASY's module tasks.

Additional information is considered for real time estimation, such as flow (cars per time), road conditions, trails, among others. This information is difficult to obtain from people but easy to calculate from additional devices such as cameras, sensors, or monitoring devices.

In this proposal the data was collected through cameras installed during short periods during rush and normal hours, the data stored was used to train MASY. The main purpose of MASY module is to process the information through a machine learning strategy. An overview of MASY's process is presented in Fig. 2. Once MASY receives the data, is divided into six tasks, the first one consist of dividing independent and dependent variables; the second task is the standardization of data in which the data is transformed to a normal distribution; In the third task, the training dataset and test are made; In the fourth task, the data is normalized; Fifth task is the data training and finally results are presented.

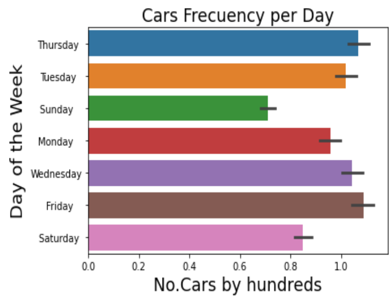

Fig. 3. Number of cars each day for a week and the days of the week.

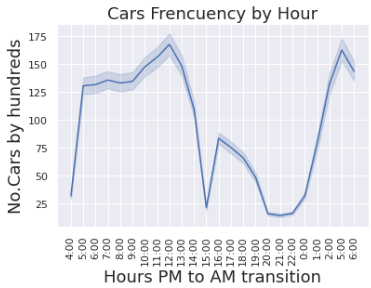

Fig. 4. Number of cars each day for a week and the days of the week.

Traffic is calculated by equation 1. A data visualization analysis of the dataset helps to understand how the city's dataset results are compared with real data of the city. Fig. 3 shows the calculation of the car behavior of each day for a week.

As we can notice, there was more car traffic on Thursdays and Fridays, followed by Wednesdays, and Tuesdays with a similar number of cars, and Sunday with a minor number of cars in the week. In Fig. 4 the amount of cars is displayed during different hours of the day. The graph represents the average number of cars measured during a week at different times. Once this information is obtained, is sent to NASY.

3.2 Neuronal Artificial System (NASY)

NASY is used for traffic classification. Its objective is to perform a traffic prediction and to inform whether there will be traffic or not based on real estimations. NASY uses traffic flow theory [2], it was trained using 60,000 samples from Australian Smart city data. NASY receives the data from MASY, processes the information using a neural network, and supervised learning to predict the traffic behavior.

The data entries for NASY are speed, flow, and density of cars. The *Speed* is defined as a parameter according to the kind of road and traffic regulations; the density (D) is calculated using equation 2; other variables are considered, such as date, day, street type, suburbs, and the location for an accurate prediction. A pre-processing procedure is done equal to MASY:

$$D=N/d, \tag{2}$$

where N is the car number, and d is the distance of the road, avenue, or trail. An overview of NASY's process is presented in Fig.5. NASY is composed of seven tasks. MASY and NASY perform the same procedure for the first four tasks. The fifth task is the simulation of the neuronal network; the sixth task performs the data training called ReLU (Rectified Linear Unit) and finally, in the seventh task results are presented.

Fig. 6 shows the ANN representation used in NASY, where green nodes represent the input data, mine while the blue ones present the hidden layers in the neuronal network. NASY uses a deep learning model built using Tensorflow[15] and Keras[16].

NASY is configure using 60 neurons in the input layer and 30 neurons in every two hidden layers. According to [17] these layers are enough to process the information.

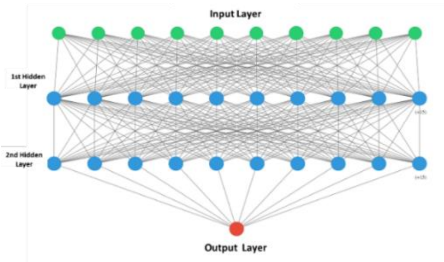

Fig. 5. NASY's process.

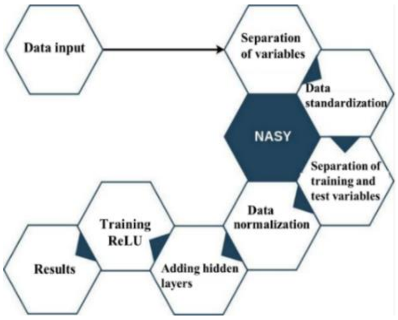

Fig. 6. ANN representation used in NASY.

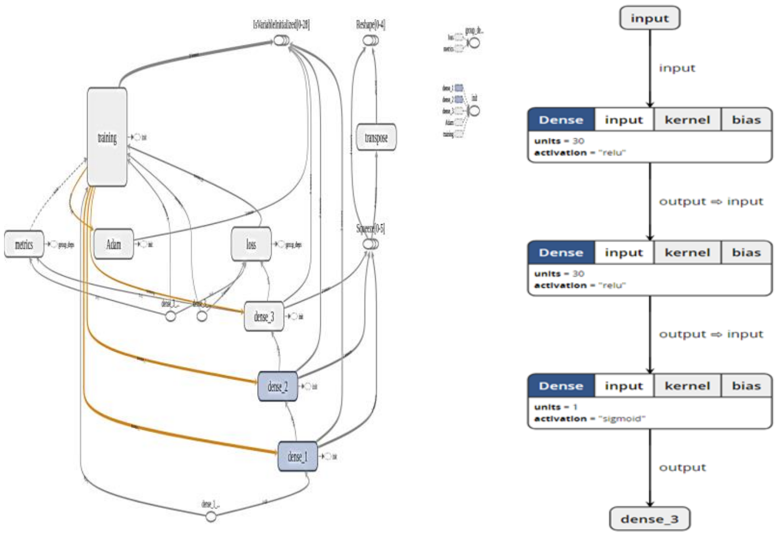

Fig. 7. Learning process of NASY based on Tensorflow and the H5 model visualization of the ANN used in NASY.

In [18, 19] the authors considered a batch size 32, epochs are defined as 100 which is the optimization parameter obtained after finish running the test. In the proposal was implemented an exhaustive grid search where the size of the batch, epochs, and optimization functions was compared. The learning algorithm proposed is based on backpropagation.

The reLU function only takes positives value and has been proved that reLU shows an engaging training behavior in experimental practice presented in [5]. The data stored by NASY are Data, Road_name, Location, Suburb, Speed_Limit, Direction, Time, Avg_speed, Max_speed, number of cars (cars), K for the vehicular flow, day of the week, Maximum vehicular flow (MaxK), Month, Year, and Type of road.

The model obtained from Tensorflow is presented in Fig. 7, shows the NASY learning process and provides an H5 (Hierarchical Data Format) file, designed to store large amounts of data to be used in the user interface explained in the next section.

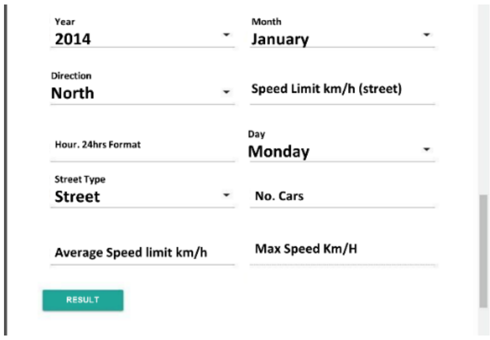

Fig. 8. Web user platform for prediction of the traffic.

The H5 model is composed of the data entries for the ANN, density (number of connected nodes), *unit* (defined as 30, these are the entries to hidden layers), and finally, the *activation function* (reLU for hidden layers and the sigmoid function for outputs).

3.3 Web User Application (WeUsAP)

The Web platform is the third component in the architecture and is useful to visualize the results obtained from NASY and MASY. The results help us to improve the models and test the models using different values. The platform uses Flask (a micro web framework for Python), a database based on PostgreSQL, and Heroku to enable the developers to build, run, and operate applications entirely in the cloud. The platform works as a simple web application; it has a friendly graphical interface measured by metrics given by Google called *Material Design*. Fig. 8 shows the user interface for the traffic prediction. The users use this application to make a traffic prediction. The required input data are the day, month, year, speed limit, type of road, among others. The information introduced pretend to be known by people who use the platform and is easy to define without having a specialized knowledge of vehicular traffic.

After making a traffic prediction, every user has to answer the next question. Do you think the car prediction was right? The information is stored in the database to make a posterior analysis about making improvements or adjustments to the algorithms for a better car prediction.

3.4 Car Counting Wizard (CCW)

CCW is the component proposed to automate the car counting to compare real data with data predictions by NASY. This component integrates two submodules Tracking System (TRASY) [21]. TRASY is an open code algorithm written in python that uses OpenCV for image manipulation. The main objective of TRASY is counting objects that pass through a green frame. 2) You Only Look Once (YOLO) [22]. YOLO is an open-source project written in C and CUDA for real-time object detection. YOLO classifies images into the types of objects identified, such as motorcycle, bus, cars, trailers, even people, and stores the information in a database.

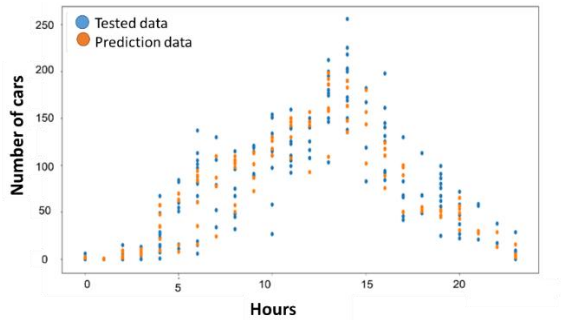

Fig. 9. MASY's prediction results.

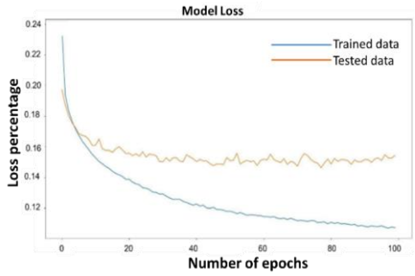

Fig. 10. NASY's prediction results.

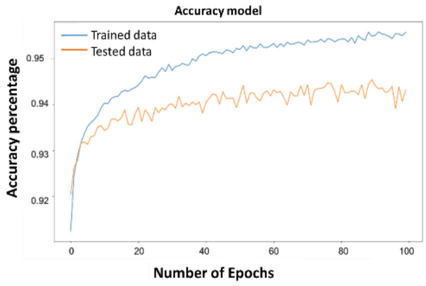

Fig. 11. NASY's model accuracy.

The result information obtained in this module is used to compare the results between the proposed architecture and the obtained by CCW.

4 Results

An original trained data set is represented by Ds_i. A subset DS_i*'* is randomly selected $DS_i' = rand(DS_i)$; The *tested dataset* DS_T is the remaining dataset classified as true or false once the outputs of training are given. Based on the Ds_i, MASY is able to predict the average number of cars that could appear in a specific hour of one day. In Fig. 9, the results achieved for traffic predictions made by MASY are presented. Blue dots represent the DS_T, and yellow dots show the predictions. The car prediction presents a 90% of accuracy of a day.

Fig. 10 shows the relationship between the input data (trained data DS_i) and the results obtained. The loss of precision in the results of the proposed model (NASY) is not significant and the model is able to adapt to the missing information to obtain good results. The number of epochs is set to 100.

The prediction is still being stable and maintains the accuracy (Fig. 11), the blue line represents the trained dataset DS_i and yellow line describes the test dataset DS_T. The performance of the algorithm is good and it is capable of adapting to obtain good predictions whether the information is not complete or if the information is loss.

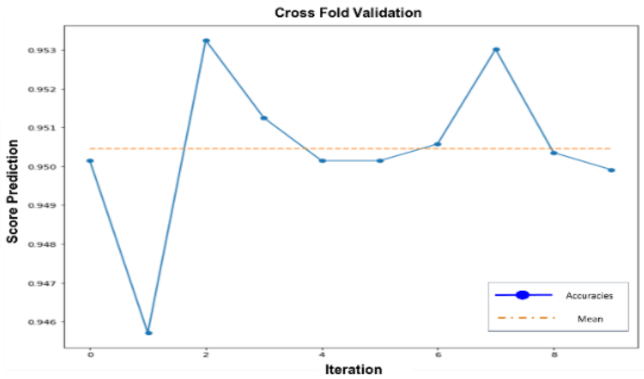

Fig. 11. Architecture cross validation for urban traffic predictions.

After training and testing the model, a cross-validation technique is made in order to observe and analyze the behavior of the architecture. The goal of cross-validation is to verify the model's ability to predict new data (Fig. 12). The testing was configured to 10 iterations. It is possible to observe that as time passes the accuracy of the predictions increase. The architecture can learn from the above information to increase its knowledge and maintain the accuracy of the data.

The stability of the architecture was reached in only 2 iterations. A Hyper-parameter optimization process was applied to the model to ensure the parameters. The parameters were defined as batch size: 25 vs 32, 100 epochs vs 500, and Adam algorithm vs Rmsprop. This test was applied in a grid-search cross-validation. The best parameters according to the test were: batch size": 32, "epochs": 100, "optimizer": "Adam".

Finally, as seen in the related works, there are multiple strategies to address such a complex and highly relevant issue today, such as traffic prediction. However, current proposals are only dedicated to covering a single specific point, that is, counting cars, verifying a certain position or testing the effectiveness of an algorithm. In this proposal, we work with 4 main components. The first is a module for the identification of the count of the cars and what type they are; the second module valid and gather the information of the hour, the day, the traffic zone and components such as the type of circulation. The third module learns the parameters and compares using real data with those obtained in the model to learn and be able to make a more accurate prediction of the amount of traffic per time and day, and kind of road. Finally, the four module is an application that allows users to interact with the application to get more real and achievable data for anyone. These modules present a whole comprehensive system that uses machine learning, a neural network, among other techniques. This is a prototype that can be easily extended to be used in many places of a city, as part of a future work.

5 Conclusion

The proposal of architecture for traffic prediction based on neural networks and strategies such as machine learning, computer vision, and deep learning was made. The proposed architecture works relatively fast and in a productive manner to obtain promising results if the information used is real.

The neural network is able to obtain a better prediction and to improve the learning during the processing stage with high accuracy. Users can test the algorithm and access the architecture using real information about their location, country, and traffic situation. The web application uses achievable information for any user, and it is easy to know. Finally, a comparison was made between the data obtained from the architecture and those obtained directly from CCW.

The results obtained present that the prediction model closely resembles the counting of cars made by CCW, the approximation was quite good, and the model improves as time passes, due to its learning process. For future implementations, traffic prediction should be improved, and the simplest model should be made. In addition, this proposal will be linked to an urban traffic platform (CiudadelaSIM) to observe and analyze the behavior of vehicles based on real data. For future implementations, it is possible to use a “Low power” device using a raspberry PI. A standalone device can be used to make this implementation useful and easy to supervise. This proposal can use a device to interact with a mobile application and the request to make predictions can be made by a http request, in addition the architecture and the trained data can be exported to any mobile device.

References

1. INRIX: Los Angeles Tops INRIX Global Congestion Ranking (2017)
2. Fernández-Aguilera, R.: Elementos de la teoría del tráfico vehicular. Fondo Editorial. Santiago, Chile, pp. 216 (2011)
3. Sagarika Verma, Sayali Badade: Traffic prediction using machine learning. In: Proceeding of National Conference on Machine Learning (2019)
4. McCarthy, J.: What is Artificial Intelligence?. Stanford University, Computer Science Department, pp. 1–15 (2007)
5. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Gordon, G.J., Dunson, D.B., Dudík, M. (eds.), AISTATS of JMLR proceeding, 15, pp. 315–323 (2011)
6. Waze: https://www.waze.com/business/#boat1(2006)
7. Williams, N., Zander, S., Armitage, G.J.: A Preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification. Computer Communication ACM SIGCOMM, 36(5), pp. 1–15 (2006)
8. Zheng, W., Lee, D.G., Asce, M., Shi, Q.: Short-term freeway traffic flow prediction: bayesian combined neural network approach. Journal of Transportation Engineering, 132(2), pp. 1–8 (2006)
9. Florio, L., Mussone, L.: Neural-network models for classification and forecasting of freeway traffic flow stability. Control Engineering Practice, 4(2), pp. 153–164 (1996)
10. Kohavi, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: International Joint Conference on Artificial Intelligence (IJCAI), 2. pp. 1137–1143 (1995)
11. Van Hinsbergen, C.P., Van Lint, J.W.C., Sanders, M.: Short term traffic prediction models. In: 14th World Congress on Intelligent Transport Systems, ITS World Congress, pp. 1–7 (2007)

12. Zheng, W., Lee, D.G., Asce, M., Shi Q.: Short-term freeway traffic flow prediction: Bayesian combined neural network approach. Journal of Transportation Engineering, 132(2), pp. 1–8 (2006)
13. Kaggle: Your home for data science. https://www.kaggle.com/ (2018)
14. Yuhan Jia, Jianping Wu, Yiman Du: Traffic speed prediction using deep learning method. In: IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 1217–1222 (2016)
15. TensorFlow: https://www.tensorflow.org/?hl=es (2018)
16. Keras Documentation: https://keras.io/ (2018)
17. Panchal, G., Ganatra, A., Kosta, Y., Panchal, D.: Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers. International Journal of Computer Theory and Engineering, 3(2), pp. 332–337 (2011)
18. Buitick, L., Louppe, G., Blondel, M., Pedregosa, P., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Glober, J., Holt, B., Varoquax, G.: API design for machine learning software: experiences from the scikit-learn project. In: Workshop: Languages for Data mining and Machine Learning, pp. 108–122 (2018)
19. Scikit-learn: Tuning the hyper-parameters of an estimator. Scikit-learn developers, documentation section 3.2. (2019)
20. Nikravesh, A.Y., Ajila, S.A., Lung, C., Ding, W.: Mobile Network Traffic Prediction Using MLP, MLPWD, and SVM. In: IEEE International Congress on Big Data (Big Data Congress), pp. 402–409 (2016)
21. Nikishaev, A.: Traffic counting example based on OpenCV object detection with background subtraction (2018)
22. Redmon, J.: YOLO: Real-Time Object Detection (2018)
23. Morales, F., Ruiz, M., Gifre, L., Contreras, L.M., Lopez, V., Velasco, L.: Virtual network topology adaptability based on data analytics for traffic prediction. In: IEEE/OSA Journal of Optical Communications and Networking, 9(1), pp. A35–A45 (2017)
24. Department for Transport: Road traffic forecast. Department for Transport's National Transport Model (NTM), Moving Britain Ahead (2018)
25. Hinsbergen, C., Lint, J.W.C., Sanders, F.: Short term traffic prediction models. In: 14th World Congress on Intelligent Transport Systems (2007)