

Applying a Multilayer Perceptron for Traffic Flow Prediction to Empower a Smart Ecosystem

Yan Mendes Ferreira[1(✉)], Lucas Rodrigues Frank[1], Eduardo Pagani Julio[1], Francisco Henrique C. Ferreira[1], Bruno José Dembogurski[2], and Edelberto Franco Silva[1]

[1] Federal University of Juiz de Fora, Juiz de Fora, Brazil
{yanmendes,lucasrodrigues,eduardo.pagani,francisco.henrique,edelberto}@ice.ufjf.br
[2] Federal Rural University of Rio de Janeiro, Rio de Janeiro, Brazil
brunodembogurski@ufrrj.br

Abstract. A direct impact of population density is more cities suffering from constant traffic jams. Thinking this way, Intelligent Transportation Systems, a key area in smart cities, uses computational intelligence techniques and analyses to aid in traffic dimensioning solutions. In this context, accurate traffic prediction models are vital to creating a more autonomous and intelligent environment. With an increase in projects for intelligent cities, research in the area of computational intelligence becomes a necessity, since its models can address complex real-world problems, which are usually difficult for conventional methods. In this work, an application is introduced applying machine learning to empower a smart ecosystem. To validate it, an extensive evaluation was performed, comparing it with the state-of-the-art and, also, verifying the impact of parameter variation and activation functions on the model of traffic flow prediction. All evaluations were done using real data traffic of two very distinct scenarios. Firstly, a free traffic flow scenario was evaluated in a benchmark dataset. Then, both models were evaluated in a complex traffic scenario where traffic flow is not continuous nor large. In both scenarios, the presented application, called SmartTraffic, outperforms the current state-of-the-art, with a performance gain of over 100% when compared in the first scenario and an improvement of approximately 31%, on average, in the second one.

Keywords: Smart cities · Machine learning · Traffic flow prediction · Intelligent Transportation Systems · Smart Campus

1 Introduction

New concepts and solutions where smart objects and smart applications play an important role have emerged in the last few years, such as smart cities [2].

© Springer Nature Switzerland AG 2019
S. Misra et al. (Eds.): ICCSA 2019, LNCS 11619, pp. 633–648, 2019.
https://doi.org/10.1007/978-3-030-24289-3_47

The concept of intelligent cities addresses the use of intelligent technologies to solve the challenges of urban practices such as socio-environmental, economic and cultural activities. In this paper, we focus on traffic management, a segment of Intelligent Transportation Systems (ITS) [6,8,12] that tackles traffic dimensioning and orientation.

ITS aim to better the use of urban space and time, enhancing also the people quality of life by helping people to move around more easily, safely and economically, in a more environmentally friendly manner [5].

As part of these initiatives, accurate traffic prediction models are vital for creating an autonomous and intelligent environment [8]. Although these models are mostly used for whole cities, the concept can be also applied to other smaller but complex environments, like an university campus [2,16].

The SmartCampus[1] project is an effort to create a smart environment in an university campus. To make this possible, many initiatives are being proposed to promote the integration between existing campus' services and the development of new ones. A better comprehension of vehicle traffic in the campus, combined with validating an accurate prediction model can provide information for future applications and supporting decision making.

The university campus interconnects two key neighborhoods of our city and allows free citizen and vehicle traffic through its entrances. Thus, traffic volume is not limited to its students, professors and staff, but also by those who use the campus as a place to practice leisure or exercising activities and those who just pass by. The heterogeneous characteristics of the drivers and their behavior profiles (academic, leisure or passerby) make the prediction task harder since the traffic volume vary throughout the day. This variance is intrinsically linked to the drivers' profile and their schedules - *e.g.* beginning of academic activities and morning commuters going to work. All data is captured by a security solution recently acquired by the University that is presented in Sect. 3.

The main contribution of this work is a study on predictive techniques in the context of traffic volume estimation, culminating in an application - Smart Traffic -.

The secondary contributions derived from SmartTraffic are threefold: (i) a comparison of the performance of two distinct approaches of traffic volume estimation, adopting the best-performing one; (ii) evaluation of the performance of two activation functions in the opted model and (iii) a review of these models' parameters and their effect in the prediction accuracy. All these are done utilizing two distinct datasets, as discussed in Sect. 4. The evaluations are done using real data traffic in two very distinct scenarios: (i) a free traffic flow scenario and (ii) a complex scenario without large nor continuous traffic flow.

The rest of this paper is organized as follows. Initially, the related work in Sect. 2 is discussed. Section 3 presents this work architectural proposal and describes its workflow. Moreover, in Sect. 4, the experimental results are presented. Finally, Sect. 5 summarizes the paper and outlines future research directions.

[1] https://www.campusinteligente.ufjf.br/.

2 Related Work

In this section, ITS related papers will be discussed, focusing on traffic prediction. For this reason, each one will be discussed separately and then a comparison will be made between this works proposal and the ones in the literature.

The work [8], mentioned from now on by Huang *et al.,* poses as one of the state-of-the-art solutions for traffic flow prediction. The authors propose a Deep Learning Architecture which is achieved by training a Deep Belief Network (DBN) - that can be seen as a stack of Restricted Boltzmann Machines (RBMs) -. The RBM is an unsupervised pattern learning Artificial Neural Network (ANN) that learns through a greedy layerwise training. To fine tune the model's parameters, a supervised regression layer is added on top of the DBN. Using the premise that traffic is a correlated network, the idea of multi-task regression was exploited in the supervised learning process.

The authors implement different traffic flow forecast approaches and compare to their own architecture, outperforming all the others. Despite utilizing a sigmoid regression layer, the authors point the possibility of replacing it by other regression algorithms such as Support Vector Regression (SVR).

In [13], the authors also applied a deep learning strategy and reinforced the use of historical data to predict traffic flow. Its model is based on Stacked Autoencoders (SAE): A deep neural network that aims to reproduce the input layer into the output layer by adjusting the synapses weights. They train the network in a greedy fashion. The unsupervised learning algorithm parameters are then fine-tuned with a sigmoid regression layer.

The model presented in [7] propose the use of the Gated Recurrent Units (GRU). That is a modification of the Long Short Term Memory (LSTM) neural network, whose main characteristic is the storage of information. Thus, allowing the consideration of previous entries when processing new data, enabling it to work with time series.

While the LSTM has been used as a predictive model for traffic flow in previous works such as [15,18], GRU has not been used for this purpose until then. These two models are evaluated, comparing the results with an AutoRegressive Integrated Moving Average (ARIMA) model, concluding that both models outperformed the ARIMA model, and that in 84% of the tests the model GRU outperformed the LSTM.

It is valid to mention that the authors do not assess the scalability of the proposed model and all tests are conducted utilizing a very small amount of data (4 weeks). Since the work proposed in this paper aims to empower a smart ecosystem, scalability is key, therefore this solution was not implemented and evaluated.

The previously mentioned papers evaluate their architectures with free-flowing highways, which is a less challenging task. Differently, [4] propose an architecture composed by a spin-off implementation of a SVR, suited for online applications called On-line SVR (Ol-SVR) [14]. The authors propose an architecture that works both in typical and atypical scenarios (e.g. vehicular crashes, work zone, holidays, etc). They compare their approach with other solutions

from the literature, showing that their proposal present a solid option for both scenarios, even though it was outperformed in some cases.

All mentioned papers utilize the dataset PeMS - introduced in the Sect. 4.1 - to evaluate their proposal. The first three utilize data collected on highways, both recent and historical - *i.e.* traffic volume of past weeks on the same weekday -, while the latter utilizes both typical and atypical traffic scenarios for their evaluation, but only recent data is inputted in the prediction process. The approaches found in the literature are summarized on Table 1 and compared to the one presented in this paper.

Table 1. Related work comparison.

Work	Architecture	Historical	Atypical traffic
[8]	DBN + SVR	**X**	
[13]	SAE + SR	**X**	
[7]	GRU	**X**	
[14]	Ol-SVR		**X**
SmartTraffic	MLP	**X**	**X**

3 Proposal

As anticipated in Sect. 1, the SmartCampus project is an effort to promote a more intelligent and autonomous university campus. To achieve that, a series of smaller initiatives are being proposed and implemented as services and applications that will be available in an integrated API.

SmartTraffic is one of these initiatives' pilot, whose main objective is to provide live traffic volume estimation, serving as a support for future applications - *e.g.* estimating parking spots availability - and supporting the administration in the decision-making process, such as estimating traffic volume in a public event on the campus.

The data source that made this initiative possible comes from the security software, Sentry[2]. It is responsible for registering all vehicles license plates that pass through its cameras. It is worth mentioning that, among other transformations, the plates are hashed to preserve anonymity. It's important to reinforce that the hardware and software used could be replaced by a non-professional camera and open-source software - *e.g. OpenCV Library*.

Figure 1 synthesizes the proposal. The SmartCampus project provides encapsulated services as endpoints of an API and the workflow adopted in the implementation of SmartTraffic application is presented. Foremost, it exposes the utilized predictor and then discuss the steps in this pipeline in the following subsections.

[2] https://www.sentry.com.br/.

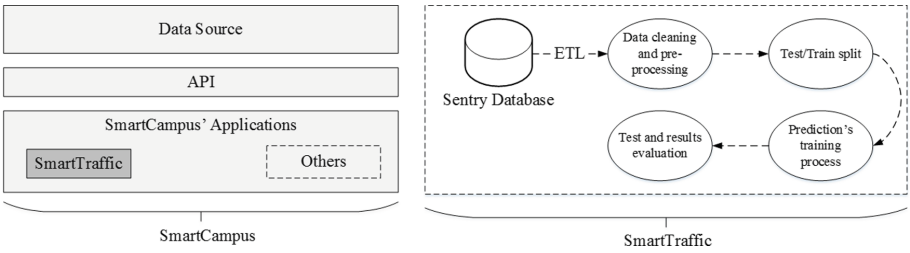

Fig. 1. SmartCampus project global view and SmartTraffic's pipeline.

3.1 Model Specification

Time series models are well-known problem solvers for traffic flow prediction. From simpler and traditional models, as the ARIMA [19], to more complex and non-linear [9], the time series approach is well-taken in the literature. Opposed to that, another noteworthy line of research utilize data-driven algorithms such as ANNs [11] or Local Weighted Learning [17] to solve traffic flow prediction problems.

This work proposal, SmartTraffic, is composed of a triple-layered Multilayer Perceptron (MLP) ANN. The usage of an MLP is justified based on the success of forecasting cyclic scenarios as seen in the work of [3].

The MLP is an ANN where all neurons in a layer are connected to all other neurons in adjacent layers. It contains an input layer, a number of hidden layers - three in this particular architecture - and an output layer. Since the result is a one dimension answer, the traffic flow forecast, the output layer contains only one neuron. The model is described in Eq. 1.

$$f(t+1) = \sum_{i=2}^{4} W_i \cdot g(W_{i-1}^T \cdot x_{i-1} + b_{i-1}) + b_i \tag{1}$$

Where: W_i and W_{i-1} is the set of weights of the ith and $i-1^{th}$ layer, respectively. b_i and b_{i-1} are the bias added to the $i+1^{th}$ and i^{th} layer, respectively. Furthermore, $g(x)$ is the activation function and, finally, x_{i-1} is the set of neurons of the $i-1^{th}$ layer.

The input layer fed to the network, defined in Eq. 2, are past observations, *i.e.* the traffic volume. Moreover, since a cyclic pattern was observed - as seen in Fig. 2 -, the model does not use only recent observations but also historic observations, *i.e.* the traffic volume of the same weekday and time of day in previous weeks. The use of historical data shows an improvement in results as seen in the work of [1,8,13].

$$\begin{aligned} x_1 = &\{\#V_t,\ \#V_{t-1},\ ...,\ \#V_{t-P}\} \cup \\ &\{\#V_{t+1-(1\ week)},\ \#V_{t+1-(2\ weeks)},\ ...,\ \#V_{t+1-(Q\ weeks)}\} \end{aligned} \tag{2}$$

Where V_t is the traffic volume at the tth time stamp, P is the number of recent observations taken into account and Q is the number of historical observations considered.

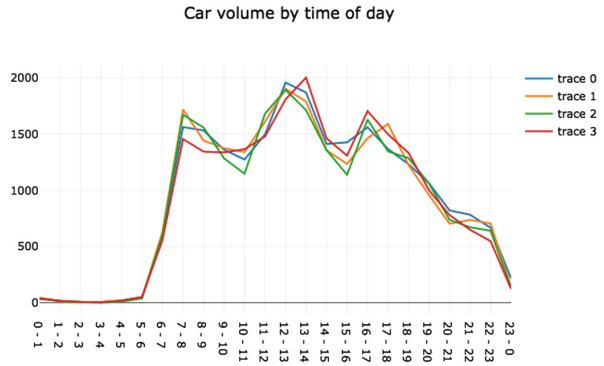

Fig. 2. Traffic volume of every wednesday in the month of september of 2017 in our university, where x-axis is the hours of a day and y-axis the car volume for four different traces.

The network learns through backpropagation [20]. In this paper, two distinct neuron activation functions are evaluated: logistic function (Logi), defined in Eq. 3, and REctified Linear Unit (ReLU), defined in Eq. 4. Those were chosen due to their popularity in the literature and their very different behaviour. The solver for the weight optimization process of choice was the stochastic gradient descent [10].

$$g(x) = \frac{1}{1 + e^{-x}} \qquad (3) \qquad\qquad g(x) = max(0, x) \qquad (4)$$

In order to guarantee reproducibility and to compare this works proposal with [8], the *scikit-learn*[3] implementations of the MLP regressor, RBM and SVR was chosen.

3.2 Data Cleaning and Pre-processing

During peaks in traffic, the software would register multiple entries for a single vehicle since it would move slow enough for the system to trigger another snapshot. All observations of the same vehicle within a 1-minute range were removed and only the first was maintained.

Some entries would not have a license plate linked to it. Since the provided dataset only had raw data and no study of the vehicle detection software was conducted, we could deal with this limitation in two different ways. It could be

[3] http://scikit-learn.org/.

a fault in the image processing algorithm but it was indeed a vehicle and should be kept as a valid information or it was a capture of something other than a vehicle - *e.g.* bike or pedestrian - and should be removed. The latter was the chosen option.

The time of day is crucial in the evaluation process. It was observed that between 20:00 and 07:00 - as seen in Fig. 2 - the volume of cars was much lower than the rest of the day. For this reason, the dataset was split in two and two predictors were trained, one for each time period. Since traffic is practically non-existent between 20:00 and 07:00 and both predictors' accuracy was very high ($\simeq$98%), only the evaluation of the dataset between 07:00 and 20:00 is shown in this work.

Another partition in the data is relative to weekdays and weekends. The traffic on weekends is mainly from people utilizing the campus as a leisure space and a very different traffic behavior was observed. The same strategy was applied here - splitting the datasets -. Due to space restrictions, we show the results of weekday's predictions only.

After splitting these sets, the traffic volume was aggregated in a time window and normalized. The normalization is applied to both datasets whereas the others processes are applied only to the university dataset.

3.3 Test/Train Split

For both scenarios a train-test split of 2/3 and 1/3, respectively, was conducted. It is noteworthy that in Huang et al. [8] original paper, the authors used 10 months of the dataset to train their model and only 2 months to test it. Even though the test set represents 17% of the dataset size, the experiments presented here utilize 33% to minimize overfitting effects and, therefore, there is a marginal difference between the results obtained in this work and [8].

3.4 Test and Evaluation Results

Along with P and Q introduced in Sect. 3.1, the number of neurons N in the hidden layer compose the parameters of the proposed architecture. The same parameters are used by Huang *et al.* [8]. However, N represents the number of components in the DBN's stack.

Even though it is possible to fine tune these parameters utilizing optimization approaches like a genetic algorithm or grid search, due to the large search spaces, random search offered a better solution since its more computationally efficient.

After performing initial tests to evaluate the model performance with different parameterizations, the following ranges were chosen: P ranging from 10 to 15 with a gap of 1, Q ranging from 0 to 5 and N ranging from 80 to 120 with 10 as a gap.

Even though there is no correlation between the number of neurons in the hidden layers and the number of components in the RBM, after the initial tests, the same range was positively evaluated for both parameters. Moreover, Huang et al. [8] achieved their best results utilizing a similar number of components.

Since the synapses weights' initial values are directly related to the prediction's performance, the best option was to randomly attribute them. We utilized cross-validation to assess our results. More precisely, we applied K-fold with $K = 3$ and five repetitions.

4 Evaluation

In this section, the model previously presented is validated and compared with the results presented in [8] in two distinct scenarios: initially, with the PeMS benchmark dataset, in Sect. 4.1 and then, in Sect. 4.2, a scenario where traffic is not continuous nor large.

It is valid to mention that this work focus on comparing the results with Huang et al. [8] approach since it was the technique that obtained the best results in the literature, considering the state-of-the-art review done during this work.

Moreover, a parametrization analysis of both models is assessed in order to further increase this monograph contribution for the PeMS scenario. The same analysis for the University scenario is not shown due to the results' similarity and, therefore, lack of relevance. Since different activation functions are also being evaluated, all parametrization analysis will be based on the results generated by best performing function.

We evaluate the models' performance in both scenarios using the Mean Absolute Percentage Error (MAPE) metric. It is defined in Eq. 5.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{\hat{y}_t - y_t}{\hat{y}_t}\right| \quad (5) \qquad R^2 = 1 - \frac{\sum_{t=1}^{n}(y(t) - \hat{y_1}(t))^2}{\sum_{t=1}^{n}(y(t) - \hat{y_2}(t))^2} \quad (6)$$

For the second scenario, an adaptation of the Coefficient of Determination metric was also used, denoted by R^2, to assess the improvement of the Smart-Traffic model over Huang *et al.* [8] proposals.

To do this, the area between the curves produced by the models' prediction - $\hat{y_1}(t)$ and $\hat{y_2}(t)$ - and the true function - $y(t)$ - is compared. It is defined in Eq. 6. The greater the value of R^2, the greater the improvement of one model over the other.

Finally, the performance the proposed model in both scenarios is evaluated utilizing two different activation functions, as described in Sect. 3.1, comparing the results obtained.

All tests were run in a machine which has the following configuration: Intel Core i7-3770 CPU @ 3.40 GHz × 8 processor, 8 GB of RAM and on a 64-bit Ubuntu 16.04 operational system. The programming language of choice was Python 3.5.2.

4.1 PeMS

Scenario Description. The PeMS[4] is a well known benchmark dataset for traffic-related problems. It consists of an open repository of data collected by a network of sensors. Its sensors are spread across the state of California that register traffic-related information such as vehicle volume in a highway.

In order to reproduce the test scenario from [8], a dataset with the same characteristics used by them was extracted, *i.e.* 12 months from the fifty most busy roads in the year of 2011.

Results. Initially, statistical analysis is conducted to assess the impact of each variable in each models' performance, *i.e.* a two-factor (NN or RBM) and (i) six treatments for P (10, 11, 12, 13, 14, 15), (ii) Q (0, 1, 2, 3, 4, 5) and (iii) five treatments N (80, 90, 100, 110, 120). All analysis were done with the aid of the IBM SPSS tool[5] and all tests were done using a significance level of 5%.

Since the experimental sample is over 30 elements (180 to be more precise), a Kolmogorov-Smirnov (KS) normality test was conducted. Since neither of the factors presented a normal distribution, a homoscedasticity test is not needed.

Next, the following hypothesis are formulated for each individual treatment and a non-parametric Kruskal-Wallis (KW) test is conducted. This is the appropriate choice considering there are three or more treatments. The results of the KW test are shown in Table 2.

- **HP0**: Increasing the parameter P value does not substantially influantiates neither models' prediction, *i.e* $\mu_{P=10} = \mu_{P=11} = \mu_{P=12} = \mu_{P=13} = \mu_{P=14} = \mu_{P=15}$
  HP1: Increasing the parameter P value substantially influences both models' prediction, *i.e* at least one of the equities does not hold
- **HQ0**: Increasing the parameter Q value does not substantially influantiates neither models' prediction, *i.e* $\mu_{Q=0} = \mu_{Q=1} = \mu_{Q=2} = \mu_{Q=3} = \mu_{Q=4} = \mu_{Q=5}$
  HQ1: Increasing the parameter Q value substantially influences both models' prediction, *i.e* at least one of the equities does not hold
- **HN0**: Increasing the parameter N value does not substantially influantiates neither models' prediction, *i.e* $\mu_{N=80} = \mu_{N=90} = \mu_{N=100} = \mu_{N=110} = \mu_{N=120}$
  HN1: Increasing the parameter N value substantially influences both models' prediction, *i.e* at least one of the equities does not hold

Given the test results in Table 2, the null hypothesis HP0 and HN0 are accepted. In other words, at a confidence level of 95%, increasing the value of the parameter P and N does not substantially influence neither models' prediction. Hence, the means for different values for P and N are statistically equivalents. A graphical interpretation of this results can be seen in the boxplots in Figs. 3 and 4.

[4] http://pems.dot.ca.gov.
[5] https://www.ibm.com/analytics/spss-statistics-software.

Table 2. KW's test results for each factor and treatment with a 5% significance level.

	Huang et al.	SmartTraffic
P p-value	0.203	0.818
Q p-value	0.000	0.000
N p-value	0.799	0.143

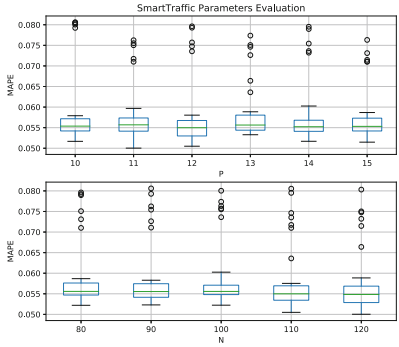

Fig. 3. Proposed model's performance in relation to the parameters P and N for the PeMS dataset.

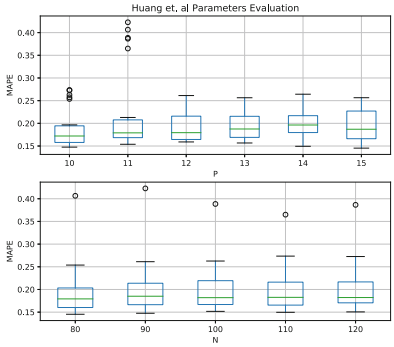

Fig. 4. Huang *et al.* [8]'s performance in relation to the parameters P and N for the PeMS dataset.

Moreover, the alternative hypothesis HQ1 is accepted. That is, at a confidence level of 95%, increasing the value of the parameter Q substantially influences both models' prediction. Therefore, at least one of the equities does not hold. To assess which, a pairwise test using the non-parametric Mann-Whitney (MW) test is performed. This is summarized in Table 3. The numbers above the main diagonal represent the p-value for the pair utilizing Huang et al. [8] means while the numbers below represent the SmartTraffic approach.

Besides two particular cases (NN $Q = 1, Q = 4$ and $Q = 2$ and $Q = 5$), the alternate hypothesis holds true in the pairwise comparison, *i.e.* increasing the parameter Q - use of historical data -, significantly changes the prediction process since means for different values for Q are statistically distinct.

Table 3. Pairwise MW test for variable Q.

Q value	0	1	2	3	4	5
0		.000	.000	.000	.000	.000
1	.000		.000	.000	.000	.000
2	.000	.000		.000	.000	.000
3	.000	.002	.000		.019	.000
4	.000	.790	.001	.009		.005
5	.000	.007	.095	.000	.028	

Considering the test results and statistical analysis, there is strong evidence that the usage of historical data strongly influences the prediction process in a positive manner as observed in both Boxplots Figs. 5 and 6.

The usage of historical data, *i.e.* $Q > 0$, improves both models' performances, ratifying the evidences pointed by [1,8,13]. The model proposed by Huang et al. [8] shows, on average, a steady improvement the more historical data it uses. As for the model proposed in this monograph, the usage of historical data is crucial in the prediction process, but it does not seem to take as much advantage of it as the previously mentioned models does. All these considerations are drawn from Figs. 5 and 6.

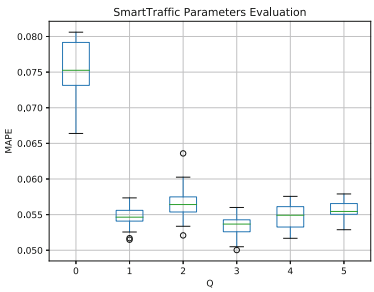

Fig. 5. Proposed model's performance in relation to the parameter Q for the PeMS dataset.

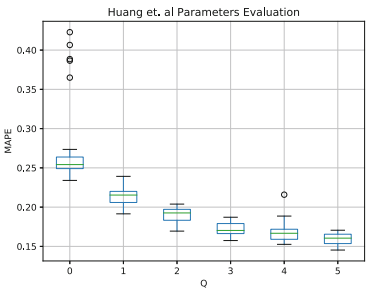

Fig. 6. Huang *et al.* [8]'s performance in relation to the parameter Q for the PeMS dataset.

The MAPE metric evaluation, presented in Fig. 7, shows that the SmartTraffic consistently outperforms Huang *et al.*'s approach. This can be verified by the best performing model, which had the following configuration: $P = 11$, $Q = 3$, $N = 120$ and it scored a 5.00% average MAPE and its best performance resulted in a MAPE of 4.68%. A plot of said model is shown in Fig. 8.

In comparison, the Huang *et al.* approach obtained an average MAPE of 14.68% and its best performance resulted in a 9.93% MAPE. Even though there are implementation and parametrization differences, the results obtained on the same dataset are comparable by, approximately, 1% for the best model.

The superiority of the ReLU (Eq. 4) over the Logistic function (Eq. 3) is incontestable for this scenario, since it outperforms the latter in every test run, as seen in Fig. 7.

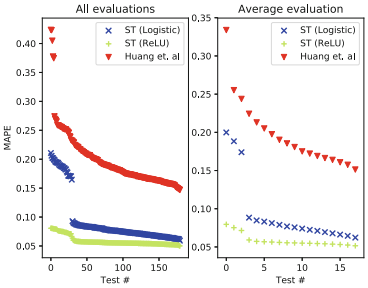

Fig. 7. All configuration evaluations for both models on the left chart. The chart on the right averages each 10 closest points for a cleaner visualization.

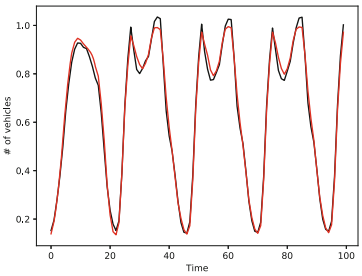

Fig. 8. Prediction curve of the SmartTraffic. The black line represents the true value and the red one the model's forecast.

4.2 SmartTraffic Project

Scenario Description. As stated throughout the paper, this specific university campus represents a peculiar scenario. Its traffic does not reflect most benchmark datasets found in the literature, such as PeMS. For that reason, an investigation is needed to assess the quality of prediction models in this particular case.

An overview image of the university's campus is presented in Fig. 9. There are three capture points, the northern and southern gate, represented by the numbers 1 and 2, respectively, and one in the road that leads to another major nucleus in the university campus, represented by the number 3. The traffic direction in the highlighted ring is one-way and flows counter-clockwise and the only entrances and exits are the cited gates. From now on the capture points will be referred by their numbers on the map.

Due to hardware problems, 2 was nonoperative during a considerable period of time that overlaps with the data this project had access to. Thus, these were discarded and all tests were performed using data from points 1 and 3.

The data captured was processed to fit the project needs and it correlates timestamps to how many vehicles passed through at that time. Even though this is a configurable parameter in the service, in the conducted experiments those values were grouped in chunks of 5 min, since it is the most used time window by related works.

Even though it is a major converging point for traffic in the city, the traffic volume is not big nor stable during the day. The raw dataset contains 249,142 entries, but it has many gaps between timestamps, mainly between midnight and 6am, where traffic usually is inexistent.

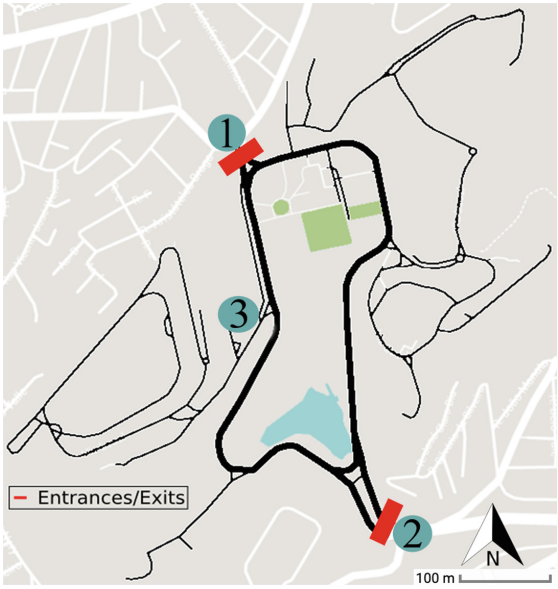

Fig. 9. Data collection points.

Results. Regarding the data, due to zeros computed, a smoothing utilizing the additive, *i.e.* add-1 estimation, also called *Laplace smoothing*, was performed. It is described in Eq. 7, where, k equals 1, $c(w_i)$ represents how many times the $i-est$ element of the array appears on the data and V is the size of the set.

$$P_{Add-k}(\omega_i|\omega i-1) = \frac{c(\omega_{i-1},\omega_i)+k}{c(\omega_{i-1})+kV} \tag{7}$$

After, a symmetric behavior is performed when forecasting, utilizing $P =$ 5-minute recent time frames to forecast the aggregate traffic in the next 5 min.

SmartTraffic outperforms Huang *et al.* by a satisfactory margin as seen in Fig. 10, even though the results were not as good as the free-flowing scenario.

The proposed application obtained an average MAPE of 14.22% and the best performance was 13.36% in its best model configuration, which was: $P =$ 11, $Q = 5$ and $N = 110$. Huang's approach obtained its best performance for the following configuration: $P = 13$, $Q = 5$ and $N = 100$, scoring an average MAPE of 18.74% and a best performance of 18.47%.

For all tests ran, the R^2 metric was computed with models that had the same parameter configuration, comparing both their performances'. A boxplot of this metric is plotted in Fig. 11, which reinforces the proposed model superiority over the state-of-the-art, specially with the ReLU (Eq. 4) activation function.

The distribution for all evaluations shows that, on average, SmartTraffic is approximately 31% better than Huang *et al.,* considering the same parameter configuration. It is also possible to see that 50% of these evaluations are between

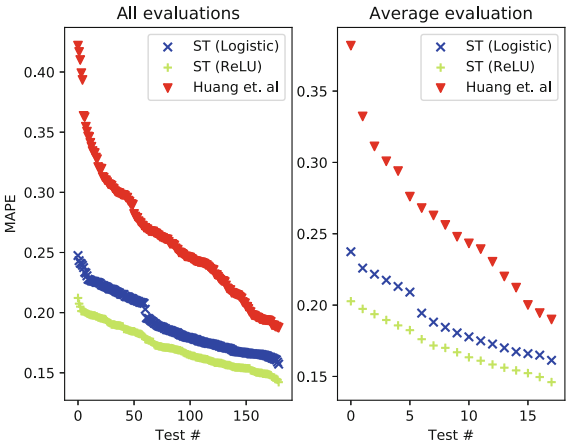

Fig. 10. All configuration evaluations for both models with a 5-minute time window on the left chart. The chart on the right averages each 10 closest points for a cleaner visualization.

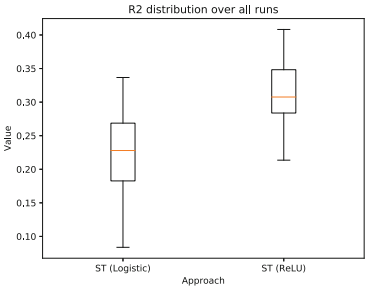

Fig. 11. R^2 metric boxplot for the university dataset over all runs with models that had the same parameter configuration.

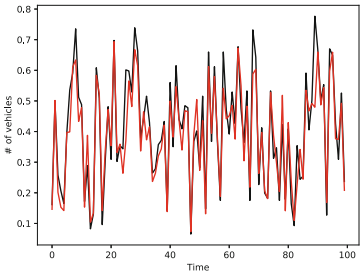

Fig. 12. Prediction curve for SmartTraffic. The black line represents the real value and the red one the model's forecast.

28% and 34%, *i.e.* half of all SmartTraffic tests are better in this percentage range. Finally, it is possible to see that 25% of these values are between 35% and 40%.

This analysis reinforces the superiority of the ReLU (Eq. 4) over the logistic (Eq. 3) activation function, specially because the second quartile of the results achieved by the ReLU network begins after the third quartile of the logistic one's ends.

However, the major performance difference between the scenarios is due to the traffic volume. The traffic volume perceived in the PeMS dataset are in the order of hundreds of thousands, when in our scenario it doesn't sum up to more than a few dozen cars during peak hours in a five minutes time window. Even

though it is attenuated by the normalization process, it is more penalizing than when dealing with large numbers.

On top of that, at both entrances, there are several traffic-controlling tools: roundabout, traffic lights, bumpers and narrower lanes. Because of that, the traffic doesn't flow continuously in the capture points, making the observations a non-stationary time series with high peaks and low valleys as seen in Fig. 12. Thus, we hypothesize that this is the main reason behind the drop in precision when compared to the other scenario.

5 Conclusion

This paper presented the SmartTraffic, one of the pilot initiatives of the SmartCampus project, an effort to support traffic-related applications in order to create a more intelligent and autonomous campus. Moreover, it presents an extensive evaluation of different state-of-the-art solutions using real data traffic in two very distinct scenarios: (i) a free traffic flow scenario and (ii) a complex scenario without large nor continuous traffic flow.

Also, the results of the statistical analysis conducted ratified the evidences pointed by [1,8,13] that the usage of historical data helps the prediction accuracy, guiding future efforts. Finally, a comparison between two activation functions is done.

As for future work, the experiments will be re-run enabling more points of data collection. Another possible investigation is the effect of splitting the raw data into even more datasets with more similar characteristics, *e.g.* months, since a significant improvement was observed when splitting the data in day/night cycles. Finally, conducting a study to cluster users with similar profiles would enlighten the visualization of the relationship between user profiles and traffic flow, allowing the prediction of traffic based on users' profile.

References

1. Abadi, A., Rajabioun, T., Ioannou, P.A.: Traffic flow prediction for road transportation networks with limited traffic data. IEEE Trans. Intell. Transp. Syst. **16**(2), 653–662 (2015)
2. Abuarqoub, A., et al.: A survey on internet of thing enabled smart campus applications. In: Proceedings of the International Conference on Future Networks and Distributed Systems, p. 38. ACM (2017)
3. Campos, L.: Modelo Estocástico Periódico baseado em Redes Neurais. Ph.D. thesis, Tese de Doutorado, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) (2010)
4. Castro-Neto, M., Jeong, Y.S., Jeong, M.K., Han, L.D.: Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions. Expert Syst. Appl. **36**(3), 6164–6173 (2009)
5. Coni, M., Garau, C., Pinna, F.: How has Cagliari changed its citizens in smart citizens? Exploring the influence of ITS technology on urban social interactions. In: Gervasi, O., et al. (eds.) ICCSA 2018. LNCS, vol. 10962, pp. 573–588. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95168-3_39

6. Dalal, K., Dahiya, P.: State-of-the-art in vanets: the core of intelligent transportation system. IUP J. Electr. Electron. Eng. **10**(1), 27 (2017)
7. Fu, R., Zhang, Z., Li, L.: Using LSTM and GRU neural network methods for traffic flow prediction. In: 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), pp. 324–328, November 2016. https://doi.org/10.1109/YAC.2016.7804912
8. Huang, W., Song, G., Hong, H., Xie, K.: Deep architecture for traffic flow prediction: deep belief networks with multitask learning. IEEE Trans. Intell. Transp. Syst. **15**(5), 2191–2201 (2014)
9. Ishak, S., Al-Deek, H.: Performance evaluation of short-term time-series traffic prediction model. J. Transp. Eng. **128**(6), 490–498 (2002)
10. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
11. Kumar, K., Parida, M., Katiyar, V.: Short term traffic flow prediction for a non urban highway using artificial neural network. Procedia-Soc. Behav. Sci. **104**, 755–764 (2013)
12. Loce, R.P., Bala, R., Trivedi, M.: Computer Vision and Imaging in Intelligent Transportation Systems. Wiley, Hoboken (2017)
13. Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.Y.: Traffic flow prediction with big data: a deep learning approach. IEEE Trans. Intell. Transp. Syst. **16**(2), 865–873 (2015)
14. Ma, J., Theiler, J., Perkins, S.: Accurate on-line support vector regression. Neural Comput. **15**(11), 2683–2703 (2003)
15. Ma, X., Tao, Z., Wang, Y., Yu, H., Wang, Y.: Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. Transp. Res. Part C Emerg. Technol. **54**, 187–197 (2015)
16. Nati, M., Gluhak, A., Abangar, H., Headley, W.: Smartcampus: a user-centric testbed for internet of things experimentation. In: 2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC), pp. 1–6, June 2013
17. Shuai, M., Xie, K., Pu, W., Song, G., Ma, X.: An online approach based on locally weighted learning for short-term traffic flow prediction. In: Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, p. 45. ACM (2008)
18. Tian, Y., Pan, L.: Predicting short-term traffic flow by long short-term memory recurrent neural network. In: 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity), pp. 153–158. IEEE (2015)
19. Van Der Voort, M., Dougherty, M., Watson, S.: Combining Kohonen maps with arima time series models to forecast traffic flow. Transp. Res. Part C Emerg. Technol. **4**(5), 307–318 (1996)
20. Werbos, P.J.: Backpropagation through time: what it does and how to do it. Proc. IEEE **78**(10), 1550–1560 (1990)