Universidade Federal de Juiz de Fora
Universidade Federal de Juiz de Fora
Bacharelado em Ciência da Computação

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

Yan Mendes Ferreira

JUIZ DE FORA
NOVEMBRO, 2018

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

Yan Mendes Ferreira

Universidade Federal de Juiz de Fora

Universidade Federal de Juiz de Fora

Departamento de Ciência da Computação

Bacharelado em Ciência da Computação

Orientador: Edelberto Franco Silva

Coorientador: Eduardo Pagani Julio

JUIZ DE FORA

NOVEMBRO, 2018

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

Yan Mendes Ferreira

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO UNIVERSIDADE FEDERAL DE JUIZ DE FORA DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Edelberto Franco Silva
Professor Doutor

Eduardo Pagani Julio
Professor Doutor

Leonardo Goliatt da Fonseca
Professor Doutor

Luciana Conceição Dias Campos
Professora Doutora

JUIZ DE FORA
29 DE NOVEMBRO, 2018

Aos meus amigos

Aos pais, pelo apoio e sustento.

Resumo

Um impacto direto da densidade populacional é um número cada vez maior de cidades que sofrem com engarrafamentos constantemente. Pensando neste problema, Sistemas de Transporte Inteligente, uma área chave de cidades inteligentes[1], usa técnicas e análises de inteligência computacional para encontrar soluções para o dimensionamento de tráfego. Neste contexto, modelos acurados de predição de tráfego são vitais para a criação de um ambiente mais autônomo e inteligente. Com a popularização de cidades inteligentes, a pesquisa na área de inteligência computacional se torna uma necessidade, já que seus modelos conseguem mitigar problemas complexos do mundo real de forma mais eficaz que abordagens tradicionais. Neste trabalho, uma aplicação que utiliza aprendizado de máquina - uma das técnicas da área - para empoderar um ecossistema inteligente é apresentada. Para validá-la, comparamo-na com o estado da arte e, também, verificamos o impacto dos parâmetros e da aplicação de diferentes funções de ativação no modelo preditivo. Todas as avaliações foram feitas utilizando dados reais de dois cenários distintos. Primeiramente, é avaliado um cenário em que o tráfego flui sem impedimentos com um *dataset* referência na literatura. Em um segundo momento, ambos os modelos são avaliados em um cenário complexo onde o tráfego não é contínuo e nem volumoso. Em ambos os cenários, a aplicação apresentada, chamada SmartTraffic, obtém melhores resultados que o estado da arte, com um ganho de performance de mais de 100% no primeiro cenário e uma melhora média de 31% no segundo.

Palavras-chave: Sistemas de Transportes Inteligentes, ecossistemas inteligentes, predição de tráfego

[1]Um ambiente composto por objetos e aplicações, usando-os para resolver desafios de práticas urbanas

Abstract

A direct impact of population density is more cities suffering from constant traffic jams. Thinking is this way, Intelligent Transportation Systems, a key area in smart cities[2], can use computational intelligence with its techniques and analyses to help in the search of solutions for traffic dimensioning. In this context, accurate traffic prediction models are vital to creating an autonomous and intelligent environment. With the popularization of smart 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 - a computational intelligence technique - to empower a smart ecosystem. To validate it, we compared compared it with the state-of-the-art and, also, verified 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: Intelligent transportation systems, smart ecosystems, traffic flow prediction.

[2]An environment composed by smart objects and applications to solve the challenges of urban practices

Agradecimentos

Aos meus pais, pelo sustento, encorajamento e apoio.

Aos professores Edelberto e Eduardo pela orientação, amizade e principalmente, pela paciência, sem a qual este trabalho não se realizaria.

Aos professores do Departamento de Ciência da Computação pelos seus ensinamentos e aos funcionários do curso, que durante esses anos, contribuíram de algum modo para o nosso enriquecimento pessoal e profissional.

"Wings begin to emerge, breaking the cycle of feeling stagnant
Finally free, the butterfly sheds light on situations that the caterpillar never considered, ending the internal struggle
Although the butterfly and caterpillar are completely different they are one and the same".

Kendrick Lamar (Mortal man)

Contents

List of Figures

List of Tables

Lista de Abreviações

ANN	Artificial Neural Network
API	Application Programming Interface
ARIMA	AutoRegressive Integrated Moving Average
DBN	Deep Belief Network
ETL	Extract-Transform-Load
GRU	Gated Recurrent Units
ITS	Intelligent Transportation Systems
KS	Kolmogorov-Smirnov
KW	Kruskal-Wallis
LSTM	Long Short Term Memory
MAPE	Mean Absolute Percentage Error
MLP	Multilayer Perceptron
MW	Mann-Whitney
Ol-SVR	On-line Support Vector Regression
PEMS	Caltrans Performance Measurement System
RBM	Restricted Boltzmann Machine
ReLU	Rectified Linear Unit
SAE	Stacked AutoEncoders
Stochastic Gradient Descent	SGD
SVR	Support Vector Regression

1 Introduction

New concepts and solutions which smart objects and smart applications play an important role have emerged in the last few years. As discussed by Abbasi *et al.* (ABBASI; SARKER; CHIANG, 2016) and Jagadish *et al.* (JAGADISH et al., 2014), there has been an increasing number of efforts by the industry and the academy towards creating more intelligent and efficient applications (HOU; WANG, 2013; ZHANG et al., 2011).

Smart environments aim to provide relevant responses with little to no user interaction, making decisions by analyzing and integrating information captured by various systems (SU; LI; FU, 2011). Its applications can range from the construction of smart homes (*e.g.* controlling lighting, electrical appliances and alarms (HAN; LIM, 2010)) to restructuring whole cities' infrastructural solutions (*i.e.* more autonomous and less bureaucratic public services (KORTUEM et al., 2010), more eco-friendly and smart power distribution (RUIZ-ROMERO et al., 2014)).

Intelligent cities (ABUARQOUB et al., 2017) addresses the use of intelligent technologies to solve the challenges of urban practices such as socio-environmental, economic and cultural activities. In this work we focus on traffic management, a segment of Intelligent Transportation Systems (ITS) (HUANG et al., 2014; DALAL; DAHIYA, 2017; LOCE; BALA; TRIVEDI, 2017) that tackles traffic dimensioning and orientation.

As part of these initiatives, accurate traffic prediction models are vital for creating an autonomous and intelligent environment (HUANG et al., 2014). 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 (ABUARQOUB et al., 2017; AL-GHAMDI; SHETTY, 2016; NATI et al., 2013).

The SmartCampus[3] 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 ac-

[3] ⟨https://www.campusinteligente.ufjf.br/⟩

curate prediction model can provide information for future applications and supporting decision making.

The studied university campus interconnects two key neighborhoods of our city and allows free citizen and vehicle traffic through its entrances. Therefore, 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 Section 5.2.

The main contribution of this work is a study on predictive techniques in the context of traffic volume estimation, culminating in an application - SmartTraffic -, considering the heterogeneous environment of the SmartCampus project - introduced in Chapter 4.

The secondary contributions derived from SmartTraffic are threefold:

- a comparison of the performance of two distinct approaches of traffic volume estimation, adopting the best-performing one;

- a review of these models' parameters and their effect in the prediction accuracy.

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 monograph is organized as follows. Initially, we introduce concepts used throughout the monograph in Chapter 2. Chapter 3 is discusses related work. Chapter 4 presents this work architectural proposal and describes its workflow. Moreover, in Chapter 5, the experimental results are presented. Finally, Chapter 6 summarizes this work and outlines future research directions.

2 Theoretical foundation

The purpose of this chapter is to review the essential background material necessary to understanding the approaches and results of this monograph. Firstly, a broad overview of Artificial Neural Networks (ANNs) is presented, since it is the machine learning model used in this work's regressor. Furthermore, we discuss the evaluation metrics used to evaluate the models.

2.1 Artificial Neural Networks

ANNs are a computational intelligence model inspired in animal brains' structure (GERVEN; BOHTE, 2018), more specifically, a set of neurons and synapses that connect them. ANNs are used to solve many different problems, specially in the artificial intelligence area (YEGNANARAYANA, 2009).

The ANN's structure is illustrated in Figure 2.1. In this illustration, the net has l layers and n neurons in each of those. Note that the latter may vary from layer to layer, however, we assume that all layers has the same number of neurons in the explanation for simplicity's sake.

The nodes (or neurons), by being stimulated by inputs (the incoming signals), change their internal state, producing an output that depends on the neuron's input and its activation.

The signals are processed by a neuron activation function $g(x)$ - such as Eqs. 2.1 and 2.2 -, producing an output that is fed to the next layer of neurons.

$$g(x) = \frac{1}{1 + e^{-x}} \tag{2.1}$$

$$g(x) = max(0, x) \tag{2.2}$$

By using this model in the supervised learning paradigm (as in the case of this

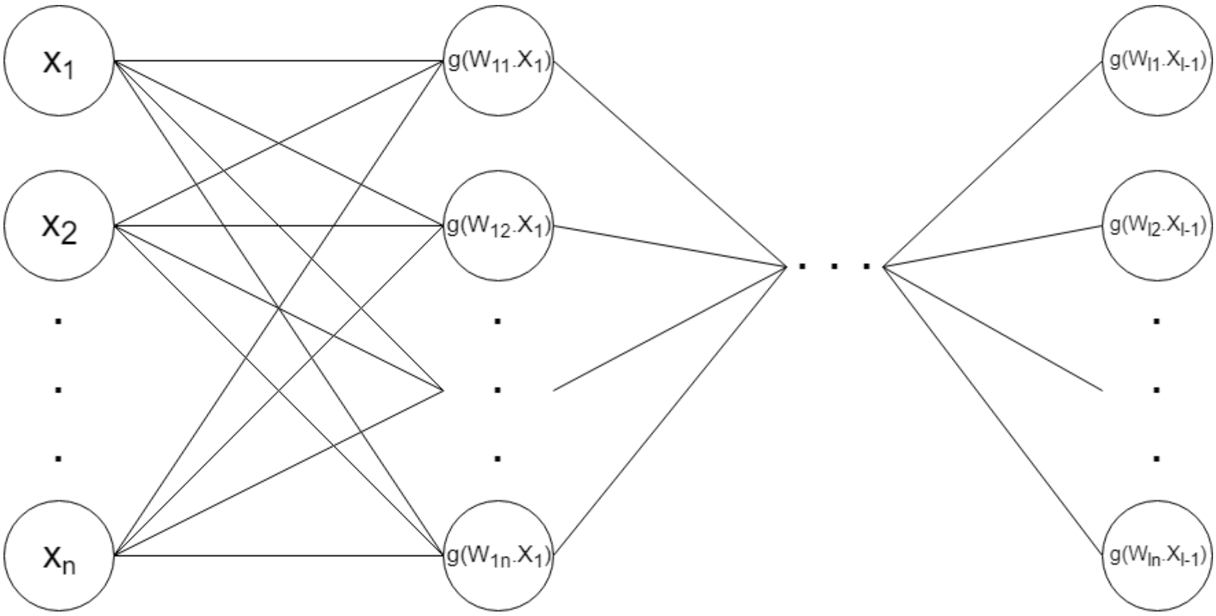

Figure 2.1: ANN's structure.

work), the ANN is a function $F : X \to Y$ that maps a set of inputs X, to an output Y. Since the learning process is supervised, there are instances which the correlation is already known (*i.e.* our training set), being the ANN's job to "learn" the function that provides this mapping.

In other words, during the learning process the ANN calibrates the synapses' weights (consequently affecting the neurons' activation), trying to find the best mapping function given a cost function that assess its effectiveness.

There are several approaches on how to execute the training process. In this work we opt for backpropagation (WERBOS, 1990) and the solver for the weight optimization process of choice was the Stochastic Gradient Descent (SGD) (KINGMA; BA, 2014).

Backpropagation aims to minimize the cost (*i.e.* the difference between the target value in our training set and the value predicted by the net). In order to do so, it uses the SGD - explained below - to calibrate the weights of the synapses. This process begins from the last layer and, as the name implies, propagates backwards the results, applying the same process for the previous layer and so forth.

SGD is a more computationally efficient version of Newton's original gradient descent algorithm, trading off efficiency for precision. Instead of evaluating the best weight calibration for a single example, it evaluates for a randomly-selected batch of training data and averages the result.

In this section we are merely trying to provide an overview of an ANN. The model is more formally discussed and mathematically defined in Chapter 4.

2.2 Evaluation metrics

The first metric of evaluation used in this work is the Mean Absolute Percentage Error (MAPE). It provides a very intuitive interpretation in terms of relative error (MYTTENAERE et al., 2016) and is commonly applied in regression problems (HUANG et al., 2014; FU; ZHANG; LI, 2016). MAPE's definition can be seen in Equation 2.3.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y(t)-\hat{y}(t)}{y(t)}\right| \tag{2.3}$$

Where $y(t)$ is the true value and $\hat{y}(t)$ is the predicted one for the time t. Since this an error metric, the smaller its value the better. Furthermore, the precision of the model can be defined by $P = 1 - MAPE$.

We also use an adaptation of the Coefficient of Determination metric, denoted by R^2, to assess the improvement of one model's prediction over the other.

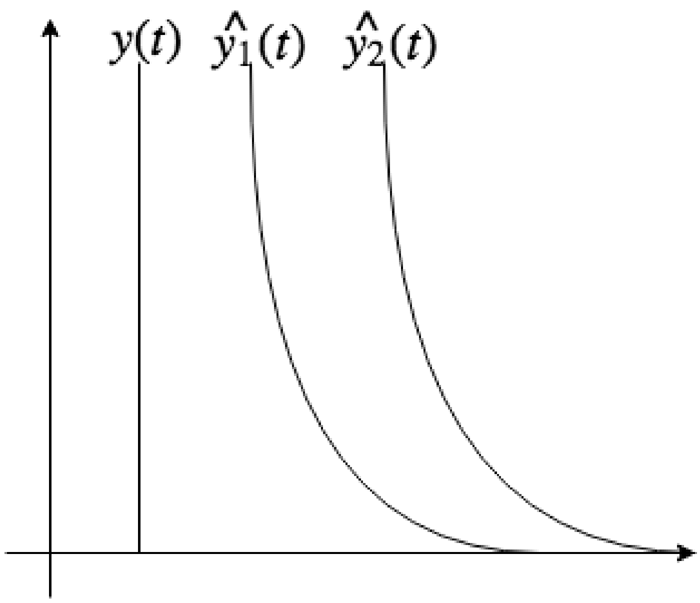

Figure 2.2: Graphical representation of the R^2 metric

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. A graphical representation of this metric is illustrated in Figure 2.2 and it is defined in Equation 2.4. The greater the value of R^2, the greater the improvement of one model over the other.

$$R^2 = 1 - \frac{\sum_{t=1}^{n}(y(t) - \hat{y}_1(t))^2}{\sum_{t=1}^{n}(y(t) - \hat{y}_2(t))^2} \tag{2.4}$$

3 Related Work

In this chapter, 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 work's proposal and the ones in the literature.

The work (HUANG et al., 2014), poses as one of the state-of-the-art solutions for traffic flow prediction. The authors propose a Deep Learning Architecture (LECUN; BENGIO; HINTON, 2015) which is achieved by training a Deep Belief Network (DBN) - is a stack of Restricted Boltzmann Machines (RBMs) (NAIR; HINTON, 2010) -. The RBM is an unsupervised pattern learning 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) (BASAK; PAL; PATRANABIS, 2007).

In (LV et al., 2015), 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 (FU; ZHANG; LI, 2016) propose the use of the Gated Recurrent Units (GRU). That is a modification of the Long Short Term Memory (LSTM) (HOCHREITER; SCHMIDHUBER, 1997) 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 (TIAN; PAN, 2015; MA et al., 2015), 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 monograph 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, (CASTRO-NETO et al., 2009) propose an architecture composed by a spin-off implementation of a SVR, suited for online applications called On-line SVR (Ol-SVR) (MA; THEILER; PERKINS, 2003). In this investigation, the authors propose an architecture that works both in typical and atypical scenarios (e.g. vehicular crashes, work zone, holidays, etc). In their work, other solutions from the literature were implemented and compared, showing that their proposal present a solid option for both scenarios, even though it was outperformed in some evaluations.

All mentioned papers utilize Caltrans Performance Measurement System (PeMS) dataset - introduced in the Section 5.1 - 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.

Table 3.1: Related work comparison.

Work	**Architecture**	**Historical**	**Atypical traffic**
(HUANG et al., 2014)	DBN + SVR	**X**	
(LV et al., 2015)	SAE + SR	**X**	
(FU; ZHANG; LI, 2016)	GRU	**X**	
(MA; THEILER; PERKINS, 2003)	Ol-SVR		**X**
SmartTraffic	MLP	**X**	**X**

This chapter presented related works. The approaches found in the literature are summarized in Table 3.1 and compared them with the architecture proposed in this work, defined in the following chapter. The advances seen in this work are the evaluation in atypical conditions and the usage of historical data in the prediction process. Furthermore, we discuss the importance of the latter in Chapter 5.

4 Proposal

As anticipated in Chapter 1, the SmartCampus[4] 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 Application Programming Interface (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.

Figure 4.1 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 sections.

4.1 Model Specification

Time series (HAMILTON, 1994) models are well-known problem solvers for traffic flow prediction. From simpler and traditional models, as the ARIMA (VOORT; DOUGHERTY; WATSON, 1996), to more complex and non-linear approaches (ISHAK; AL-DEEK, 2002), the time series approach is well-taken in the literature. As opposed to that, another noteworthy line of research utilize data-driven algorithms such as ANNs (KUMAR; PARIDA; KATIYAR, 2013) or Local Weighted Learning (SHUAI et al., 2008) to solve traffic flow prediction problems.

This work's 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 (CAMPOS, 2010).

The MLP is an ANN where all neurons in a layer are connected to all other

[4] ⟨https://www.campusinteligente.ufjf.br/⟩

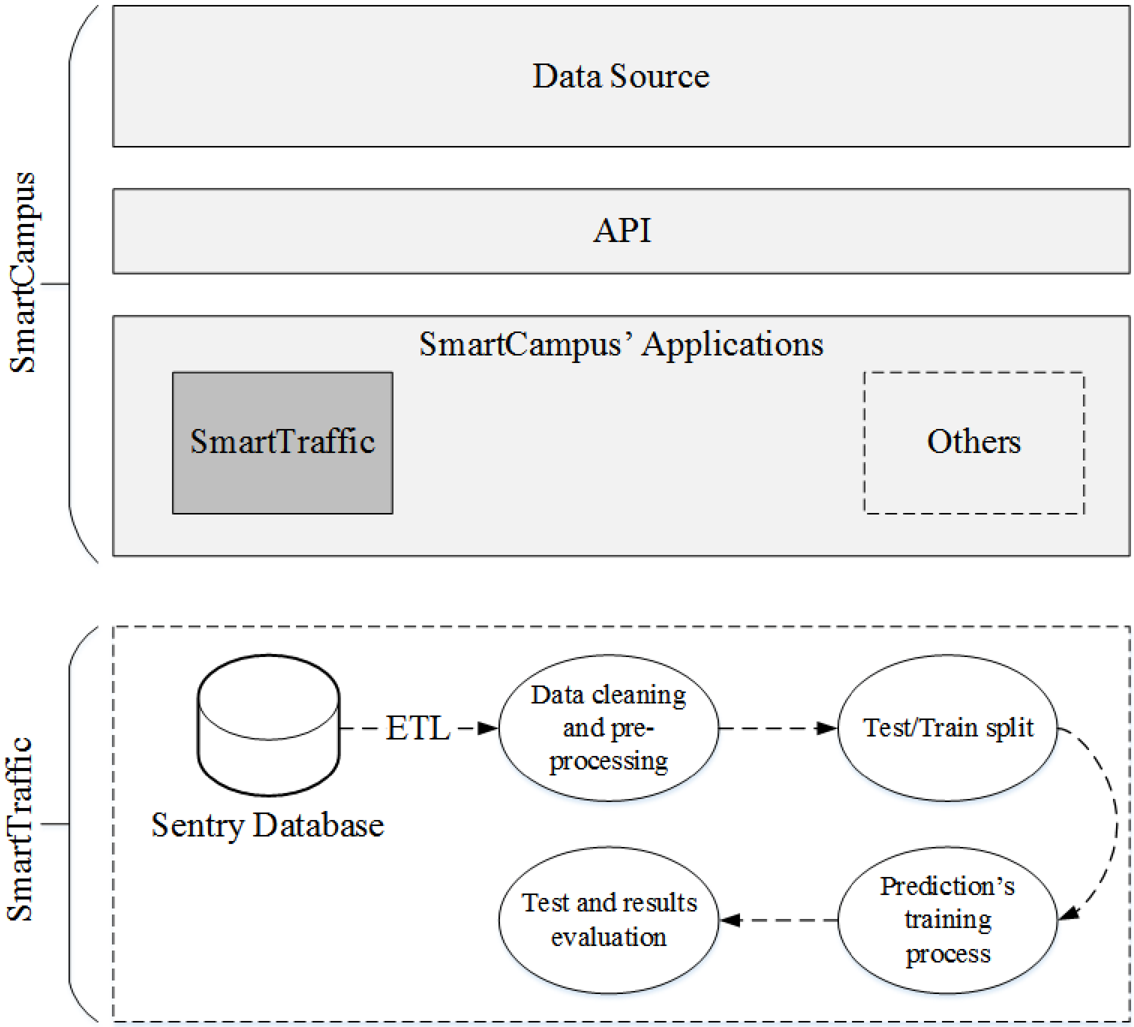

Figure 4.1: SmartCampus project global view and SmartTraffic's pipeline.

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, *i.e.* the traffic flow forecast, the output layer contains only one neuron. The model is described in Equation 4.1.

$$f(t+1) = \sum_{i=2}^{4} W_i \cdot g(W_{i-1}^T \cdot x_{i-1} + b_{i-1}) + b_i \tag{4.1}$$

Where: W_i and W_{i-1} is the set of weights of the i^{th} 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 Equation 4.2, are past observations, *i.e.* the traffic volume. Moreover, since a cyclic pattern was observed - as seen in Figure

4.2 -, 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 (HUANG et al., 2014; LV et al., 2015; ABADI; RAJABIOUN; IOANNOU, 2015).

$$\begin{aligned} x_1 = &\{\#V_t,\ \#V_{t-1},\ \ldots,\ \#V_{t-P}\} \cup \\ &\{\#V_{t+1-(1\ week)},\ \#V_{t+1-(2\ weeks)},\ \ldots,\ \#V_{t+1-(Q\ weeks)}\} \end{aligned} \tag{4.2}$$

Where V_t is the traffic volume at the t^{th} time stamp, P is the number of recent observations taken into account and Q is the number of historical observations considered.

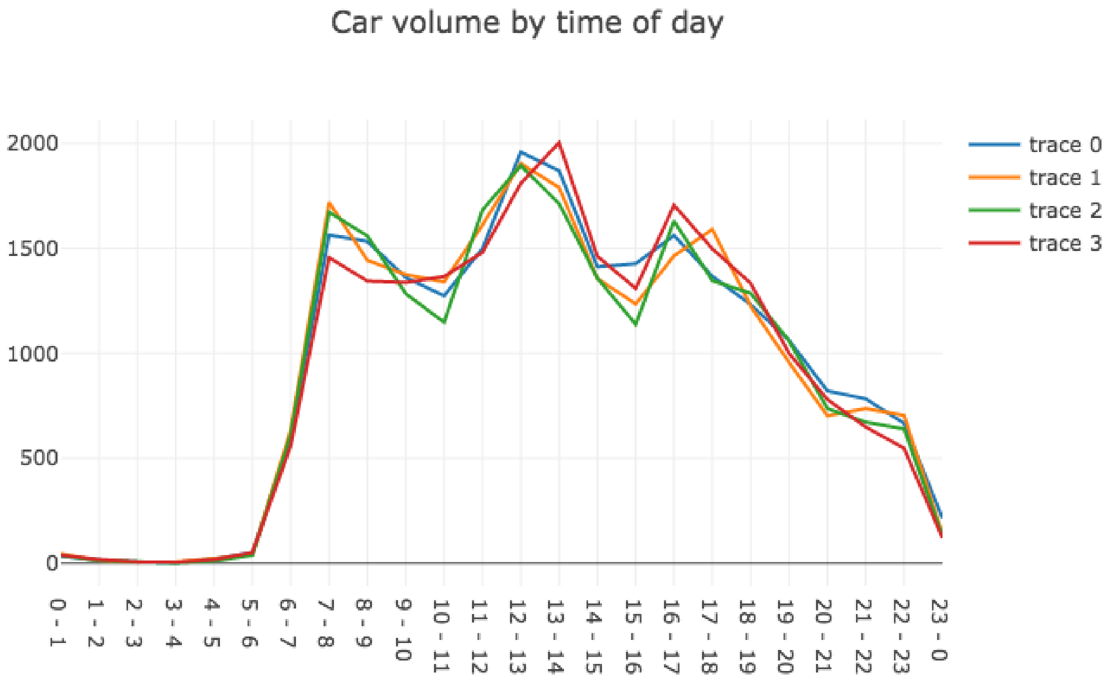

Figure 4.2: Traffic volume of every Wednesday in the month of September of 2017 in our University.

The network learns through backpropagation (WERBOS, 1990). In this work, two distinct neuron activation functions are evaluated: logistic function (Logi), defined in Equation 2.1, and Rectified Linear Unit (ReLU), defined in Equation 2.2. 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 (KINGMA; BA, 2014).

In order to guarantee reproducibility and to compare this work's proposal with (HUANG et al., 2014), the *scikit-learn*[5] implementations of the MLP regressor, RBM and SVR was chosen.

4.2 Data cleaning and pre-processing

During peaks in traffic, the software would register multiple entries for a single vehicle. Specifically, 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 entry was maintained.

Some entries would not have a license plate tied to it. Since the provided dataset only had raw data and no study of the vehicle detection software was conducted, two options were available on how to deal with this limitation. It could be a fault in the image processing algorithm, where it would not read the plate, 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 Figure 4.2 - 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 -. The results for both scenarios are presented in the following Chapter.

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.

[5] ⟨http://scikit-learn.org/⟩

4.3 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. (2014) 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 (HUANG et al., 2014).

4.4 Test and evaluation results

Along with P and Q introduced in 4.1, 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.* (HUANG et al., 2014). 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. (2014) 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. In other words, the dataset is split in three sections, two being used to train the network and one for testing. This process is repeated five times to minimize bias.

5 Evaluation

In this chapter, the model previously presented is validated and compared with the results presented in (HUANG et al., 2014) in two distinct scenarios: initially, with the PeMS benchmark dataset, in Section 5.1 and then, in Section 5.2, 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. (2014) 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 MAPE. For the second scenario, we also use the R^2, to assess the improvement of the SmartTraffic model over (HUANG et al., 2014) proposal. Both of these metrics are defined in Chapter 2

Finally, the performance the proposed model in both scenarios is evaluated utilizing two different activation functions, as described in section 4.1, comparing the results obtained.

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

5.1 PeMS

5.1.1 Scenario description

The PeMS[6] is a well known benchmark dataset for traffic-related problems (CHEN et al., 2001; RICE; ZWET, 2004). 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 highways, average speed, among others.

In the scope of this work, we utilize only free-flowing highways to assess the predictors' performance. We recreate the same test environment from Huang et al. (2014) to compare them under the same circumstances, *i.e.* 12 months from the fifty most busy roads in the year of 2011.

5.1.2 Experimental results

Initially, a 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[7] and all the following tests were done utilizing 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 5.1.

- **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' pre-

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

diction, $i.e$ 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

	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

Table 5.1: KW's test results for each factor and treatment with a 5% significance level.

Given the test results in Table 5.1, the null hypothesis HP0 and HN0 are accepted. In other words, at a confidence level of 95%, increasing the value of parameters P and N do 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 5.1 and 5.2.

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 5.2. The numbers above the main diagonal represent the p-value for the pair utilizing Huang et al. (2014) means while the numbers below represent the SmartTraffic approach.

Besides two particular cases (NN $Q = 1, Q = 4$ and $Q = 2andQ = 5$), 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.

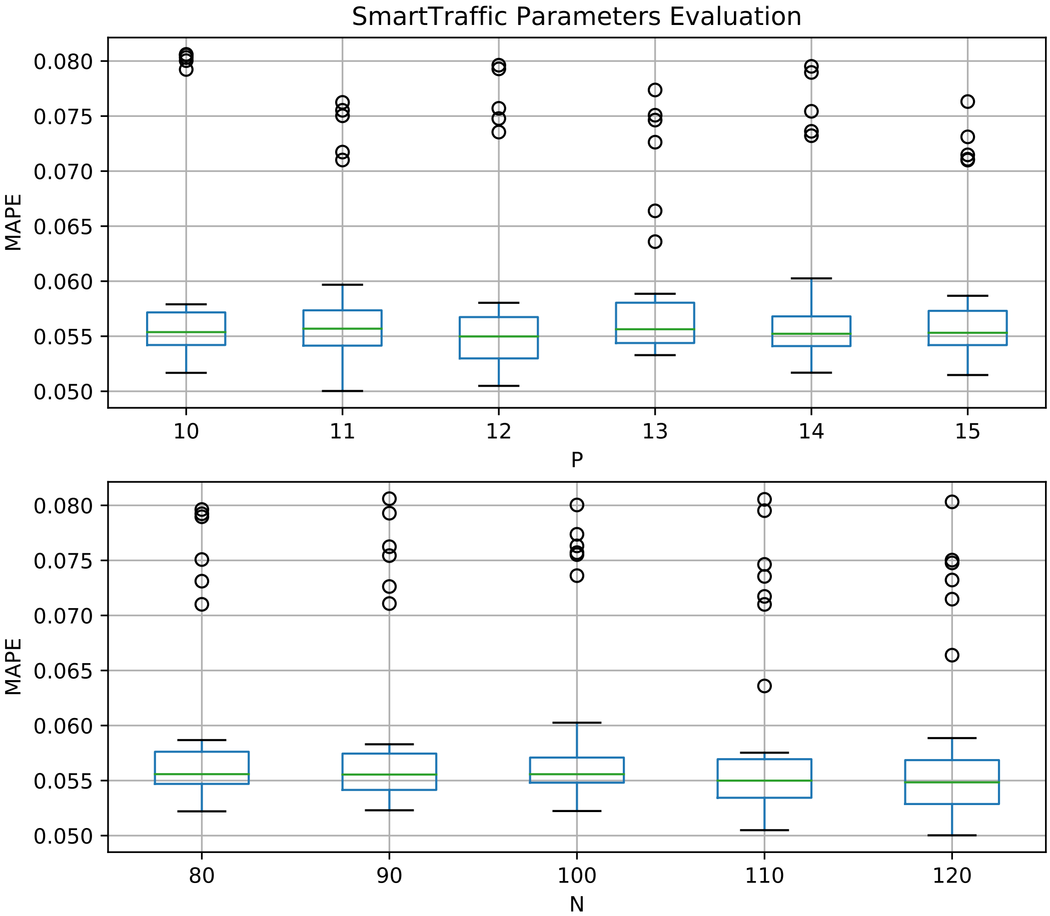

Figure 5.1: Proposed model's performance in relation to the parameters P and N for the PeMS dataset.

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 5.3 and 5.4.

The usage of historical data, *i.e.* $Q > 0$, improves both models' performances, ratifying the evidences pointed by (HUANG et al., 2014; LV et al., 2015; ABADI; RAJABIOUN; IOANNOU, 2015). The model proposed by Huang et al. (2014) 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 Figures 5.3 and 5.4.

The MAPE metric evaluation, presented in Figure 5.5, shows that the Smart-

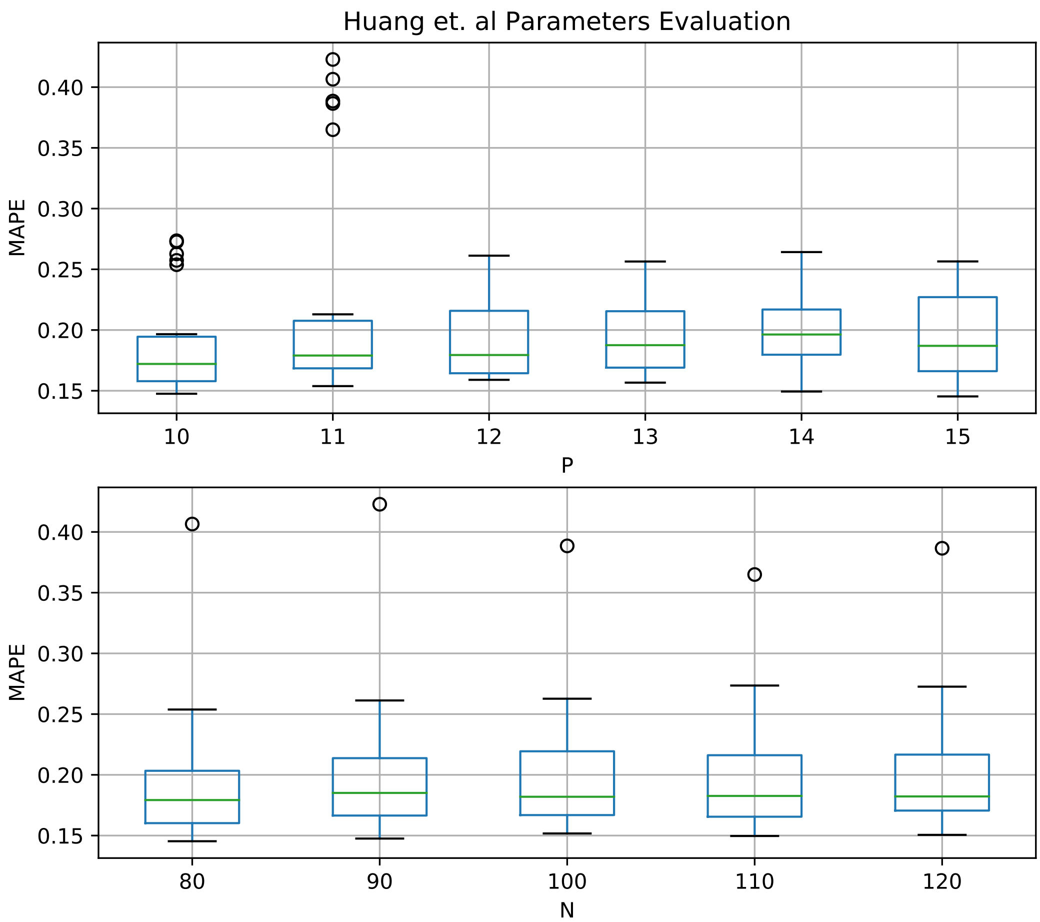

Figure 5.2: Huang et al. (2014)'s performance in relation to the parameters P and N for the PeMS dataset.

Traffic consistently outperforms Huang et al. (2014)'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 Figure 5.6.

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. 2.2 over the Logistic function (Eq. 2.1 is incontestable for this scenario, since it outperforms the latter in every test run, as seen in Figure 5.5.

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	

Table 5.2: Pairwise MW test for variable Q.

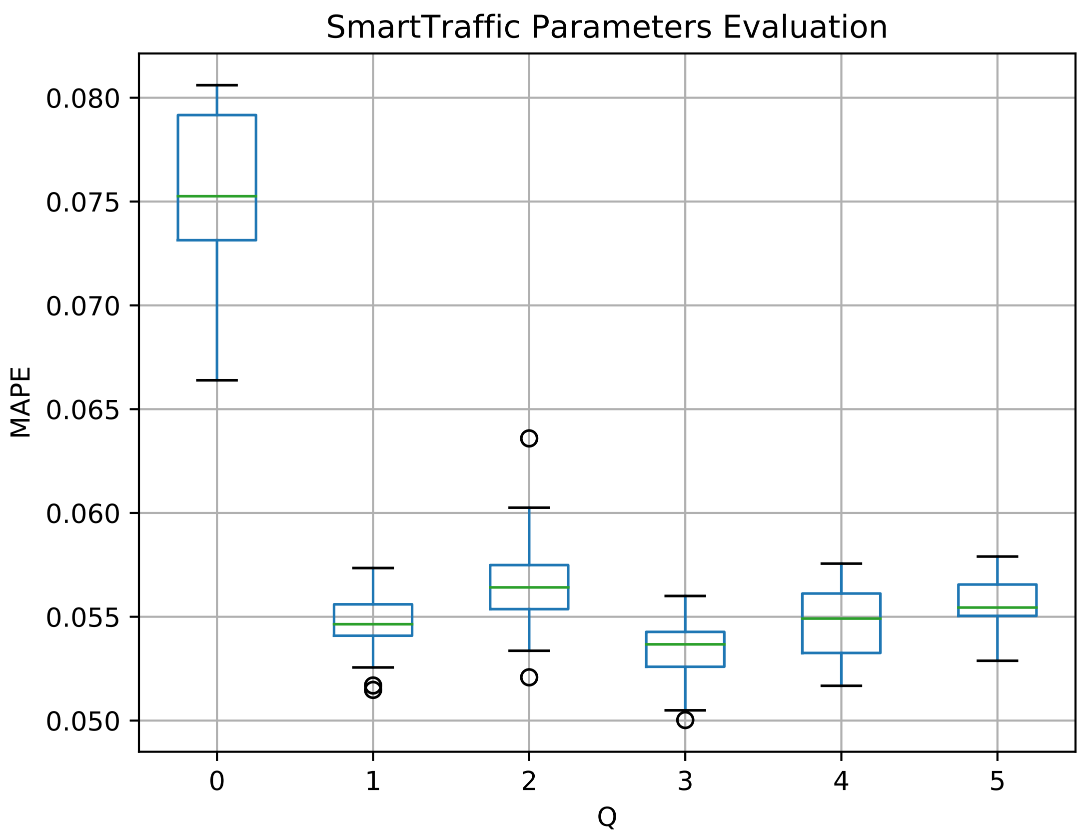

Figure 5.3: Proposed model's performance in relation to the parameter Q for the PeMS dataset.

5.2 SmartTraffic project

5.2.1 Scenario description

As already mentioned in Chapter 1, the university campus's scenario is a peculiar one. 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.

The data source that made this initiative possible comes from the security soft-

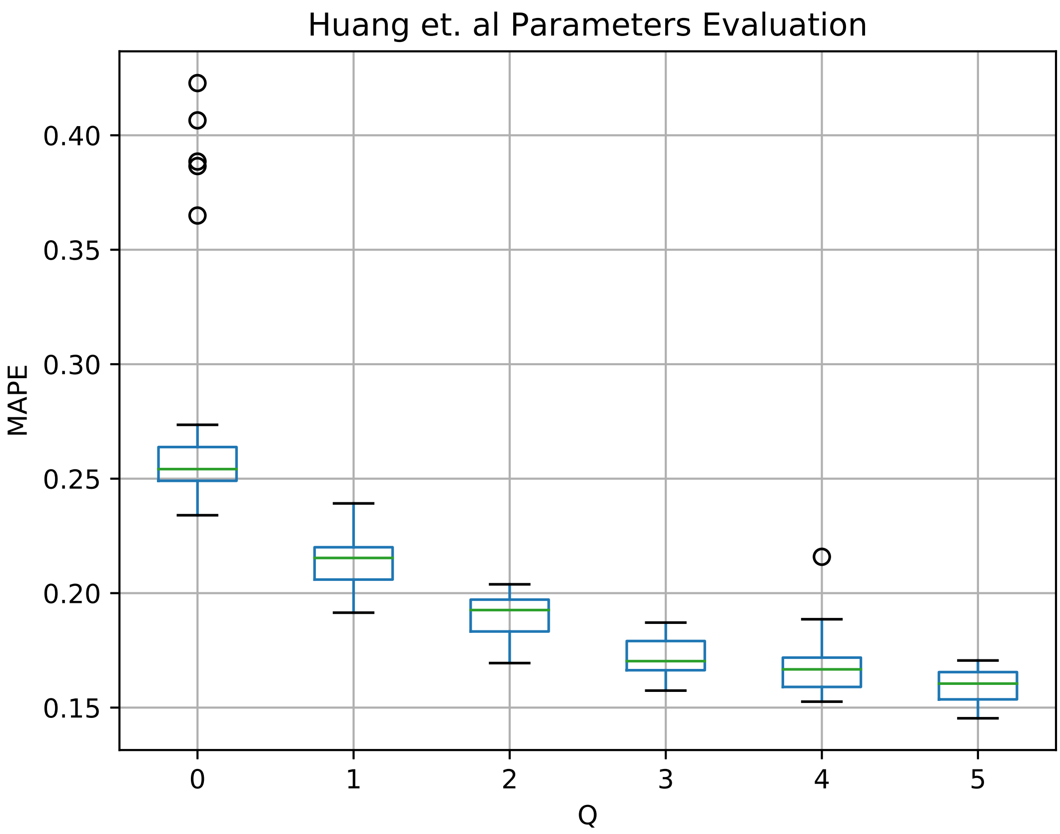

Figure 5.4: Huang et al. (2014)'s performance in relation to the parameter Q for the PeMS dataset.

ware, Sentry[8]. 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 in the Extract-Transform-Load (ETL) process to the database and only quantitative analysis are conducted. Even though it is not the scope of this work, it is necessary to make clear that the software adopted in this scenario can be replaced by a non-professional camera and open-source software - *e.g. OpenCV Library.*

An overview image of the university's campus is presented in Figure 5.7. 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

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

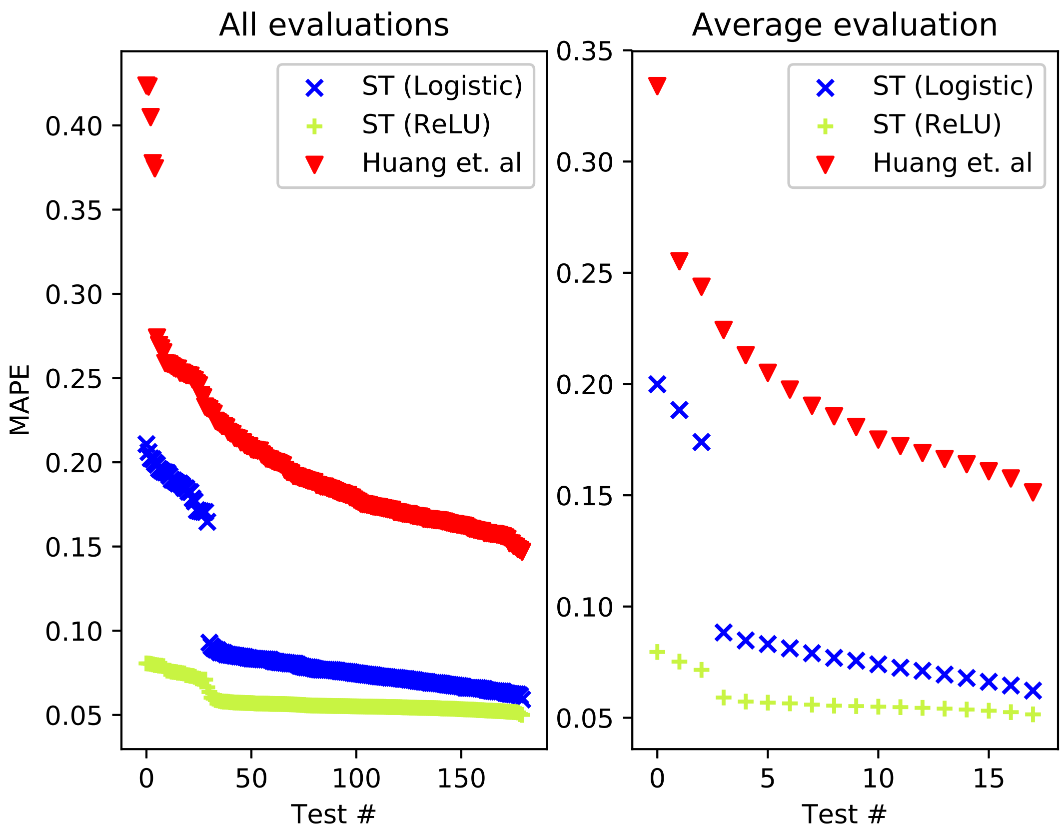

Figure 5.5: All configuration evaluations for both models on the left chart. The chart on the right averages each 10 closest points for a cleaner visualization.

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 the post-processed data consists of all transactions captured by the software at a certain point in time. Specifically, 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 minutes, 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.

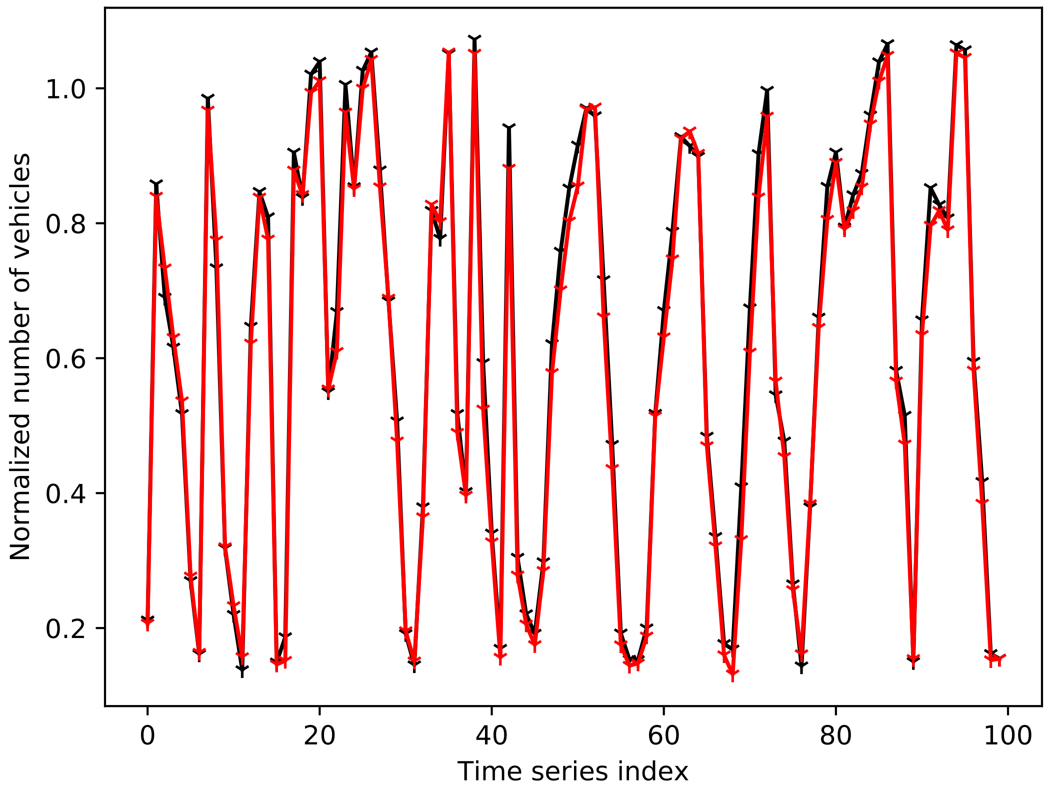

Figure 5.6: Prediction curve of the SmartTraffic. The black line represents the true value and the red one the model's forecast.

5.2.2 Experimental 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 Equation 5.1, 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.

$$P_{Add-k}(\omega_i|\omega i - 1) = \frac{c(\omega_{i-1}, \omega_i) + k}{c(\omega_{i-1}) + kV} \tag{5.1}$$

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

SmartTraffic outperforms Huang et al. (2014) by a satisfactory margin as seen in Figure 5.8, even though the results were not as good as the free-flowing scenario.

For the weekdays, the proposed application obtained an average MAPE of 14.46% and the best performance was 13.97% in its best model configuration, which was: $P =$

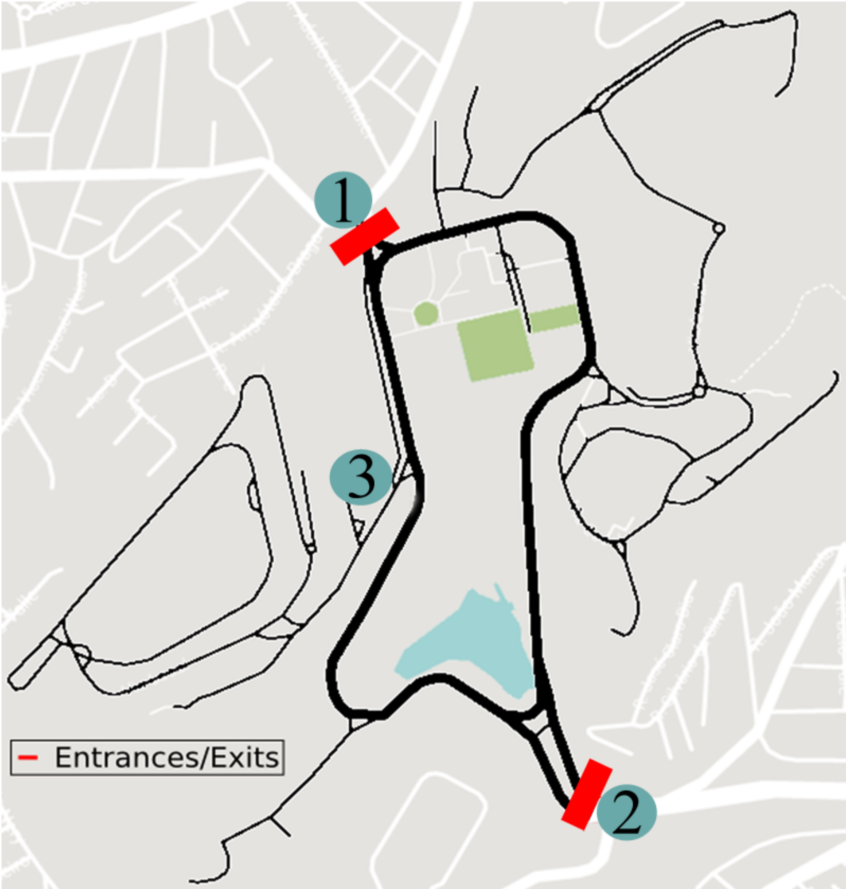

Figure 5.7: Data collection points.

13, $Q = 5$ and $N = 100$. Huang et al. (2014)'s approach obtained its best performance for the following configuration: $P = 13$, $Q = 5$ and $N = 80$, scoring an average MAPE of 18.86% and a best performance of 18.19%.

As for the weekend, the SmartTraffic approach got an average MAPE of 23.44% and the best performance was 20.86% in its best model configuration, which was: $P = 10$, $Q = 5$ and $N = 90$. Huang et al. (2014)'s approach obtained its best performance for the following configuration: $P = 13$, $Q = 5$ and $N = 80$, scoring an average MAPE of 34.11% and a best performance of 28.18%.

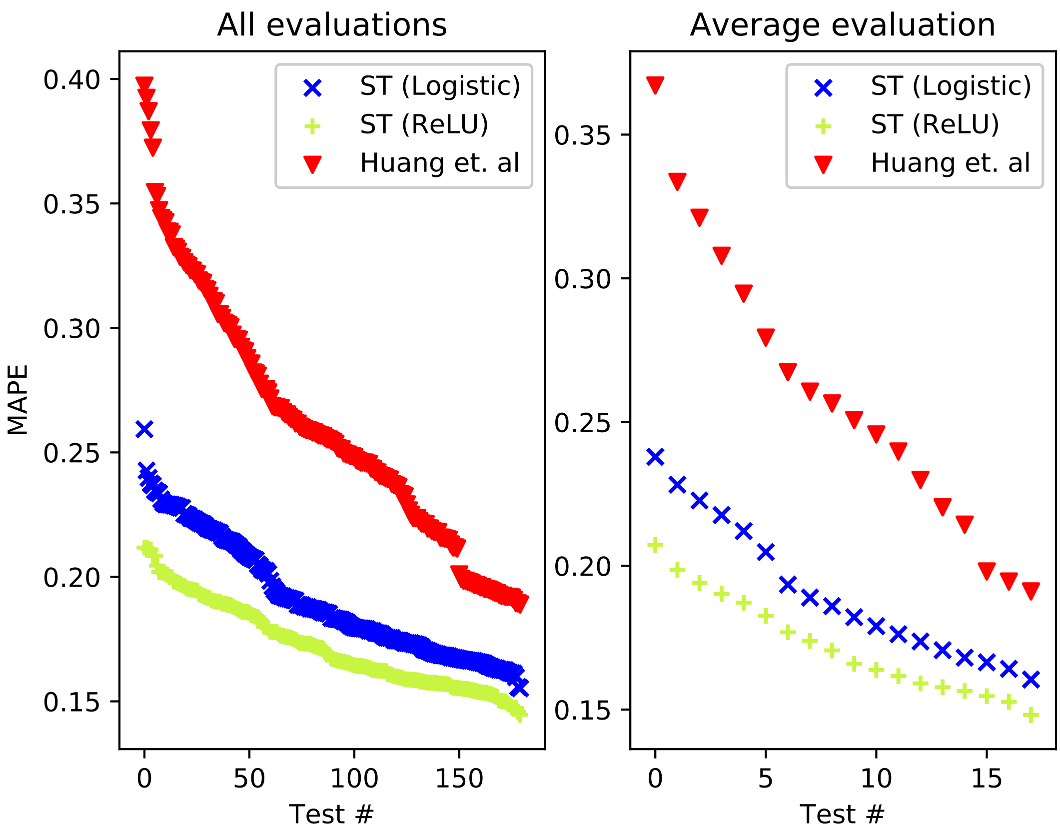

Figure 5.8: All configuration evaluations for weekdays for both models with a 5-minute time window on the left chart. The chart on the right averages each 10 closest points for a cleaner visualization.

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 Figure 5.9, which reinforces the proposed model superiority over the state-of-the-art, specially with the ReLU (Eq. 2.2 activation function.

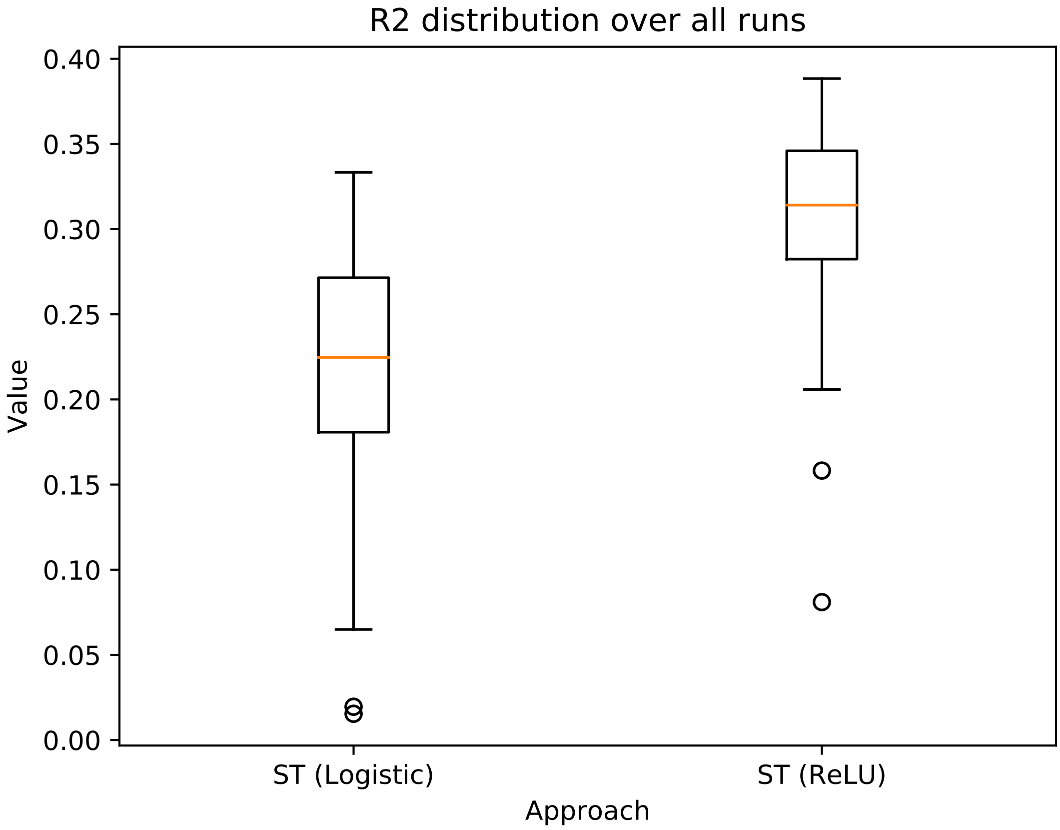

Figure 5.9: R^2 metric boxplot for the university dataset during weekdays over all runs.

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, approximately, 29% and 35%, *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 38%.

This analysis reinforces the superiority of the ReLU (Eq. 2.2 over the logistic (Eq. 2.1) activation function in this scenario, 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.

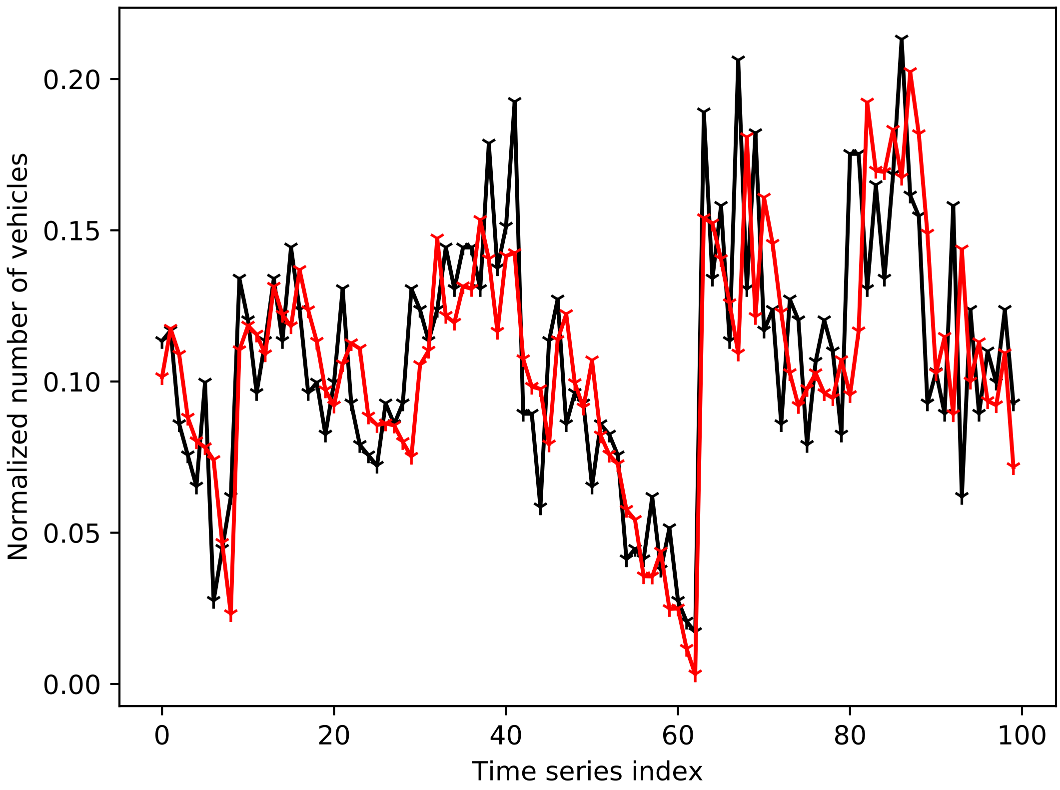

Figure 5.10: Prediction curve for SmartTraffic. The black line represents the real value and the red one the model's forecast for the university scenario.

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 Figure 5.10. Thus, there is a possibility that this is the main reason behind the precision drop between this scenario and the other previously evaluated.

5.3 Discussion

This chapter presented the experimental results of both models, along with a statistical assessment of the parameters influence in the prediction process. Furthermore, different activation functions are also evaluated and compared.

The MLP utilized in this work consistently outperforms Huang et al. (2014)'s approach in both scenarios. Moreover, the superiority of the ReLu activation function over the Logistic is also evident in both test scenarios.

Even though the university results are not as good as in the free flowing scenario, the MLP still poses as a viable option as a forecasting model to support the emerging smart ecosystem in the university campus.

6 Conclusion

This work presented a thorough study on predictive techniques in the context of traffic volume estimation, culminating in an application - namely, 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 -. Furthermore, it presents an extensive evaluation of different state-of-the-art solutions and activation functions 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. Finally, a statistical parameterization analysis was carried out to assess the influence of parameter variance in the models' accuracy.

Also, the results of the statistical analysis conducted ratified the evidences pointed by (HUANG et al., 2014; LV et al., 2015; ABADI; RAJABIOUN; IOANNOU, 2015) 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.

The last one interesting investigation to conduct is a study by clustering users with similar profiles. This study would enlighten a visualization of the relationship between user profiles and traffic flow, allowing the prediction of traffic based on users' profile.

Bibliography

ABADI, A.; RAJABIOUN, T.; IOANNOU, P. A. Traffic flow prediction for road transportation networks with limited traffic data. *IEEE transactions on intelligent transportation systems*, IEEE, v. 16, n. 2, p. 653–662, 2015.

ABBASI, A.; SARKER, S.; CHIANG, R. H. Big data research in information systems: Toward an inclusive research agenda. *Journal of the Association for Information Systems*, v. 17, n. 2, 2016.

ABUARQOUB, A. et al. A survey on internet of thing enabled smart campus applications. In: ACM. *Proceedings of the International Conference on Future Networks and Distributed Systems*. [S.l.], 2017. p. 38.

ALGHAMDI, A.; SHETTY, S. Survey toward a smart campus using the internet of things. In: IEEE. *Future Internet of Things and Cloud (FiCloud), 2016 IEEE 4th International Conference on*. [S.l.], 2016. p. 235–239.

BASAK, D.; PAL, S.; PATRANABIS, D. C. Support vector regression. *Neural Information Processing-Letters and Reviews*, v. 11, n. 10, p. 203–224, 2007.

CAMPOS, L. *Modelo Estocástico Periódico baseado em Redes Neurais*. Tese (Doutorado) — Tese de Doutorado. Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), 2010.

CASTRO-NETO, M. et al. Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert systems with applications*, Elsevier, v. 36, n. 3, p. 6164–6173, 2009.

CHEN, C. et al. Freeway performance measurement system: mining loop detector data. *Transportation Research Record: Journal of the Transportation Research Board*, Transportation Research Board of the National Academies, n. 1748, p. 96–102, 2001.

DALAL, K.; DAHIYA, P. State-of-the-art in vanets: The core of intelligent transportation system. *IUP Journal of Electrical and Electronics Engineering*, IUP Publications, v. 10, n. 1, p. 27, 2017.

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)*. [S.l.: s.n.], 2016. p. 324–328.

GERVEN, M. van; BOHTE, S. *Artificial neural networks as models of neural information processing*. [S.l.]: Frontiers Media SA, 2018.

HAMILTON, J. D. *Time series analysis*. [S.l.]: Princeton university press Princeton, NJ, 1994. v. 2.

HAN, D.-M.; LIM, J.-H. Design and implementation of smart home energy management systems based on zigbee. *IEEE Transactions on Consumer Electronics*, IEEE, v. 56, n. 3, 2010.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural computation*, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.

HOU, Z.-S.; WANG, Z. From model-based control to data-driven control: Survey, classification and perspective. *Information Sciences*, Elsevier, v. 235, p. 3–35, 2013.

HUANG, W. et al. Deep architecture for traffic flow prediction: deep belief networks with multitask learning. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 15, n. 5, p. 2191–2201, 2014.

ISHAK, S.; AL-DEEK, H. Performance evaluation of short-term time-series traffic prediction model. *Journal of Transportation Engineering*, American Society of Civil Engineers, v. 128, n. 6, p. 490–498, 2002.

JAGADISH, H. V. et al. Big data and its technical challenges. *Commun. ACM*, v. 57, n. 7, p. 86–94, 2014. Disponível em: ⟨https://doi.org/10.1145/2611567⟩.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

KORTUEM, G. et al. Smart objects as building blocks for the internet of things. *IEEE Internet Computing*, IEEE, v. 14, n. 1, p. 44–51, 2010.

KUMAR, K.; PARIDA, M.; KATIYAR, V. Short term traffic flow prediction for a non urban highway using artificial neural network. *Procedia-Social and Behavioral Sciences*, Elsevier, v. 104, p. 755–764, 2013.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436, 2015.

LOCE, R. P.; BALA, R.; TRIVEDI, M. *Computer Vision and Imaging in Intelligent Transportation Systems*. [S.l.]: John Wiley & Sons, 2017.

LV, Y. et al. Traffic flow prediction with big data: a deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 16, n. 2, p. 865–873, 2015.

MA, J.; THEILER, J.; PERKINS, S. Accurate on-line support vector regression. *Neural computation*, MIT Press, v. 15, n. 11, p. 2683–2703, 2003.

MA, X. et al. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C: Emerging Technologies*, Elsevier, v. 54, p. 187–197, 2015.

MYTTENAERE, A. D. et al. Mean absolute percentage error for regression models. *Neurocomputing*, Elsevier, v. 192, p. 38–48, 2016.

NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: *Proceedings of the 27th international conference on machine learning (ICML-10)*. [S.l.: s.n.], 2010. p. 807–814.

NATI, M. et al. Smartcampus: A user-centric testbed for internet of things experimentation. In: *2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC)*. [S.l.: s.n.], 2013. p. 1–6. ISSN 1882-5621.

RICE, J.; ZWET, E. V. A simple and effective method for predicting travel times on freeways. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 5, n. 3, p. 200–207, 2004.

RUIZ-ROMERO, S. et al. Integration of distributed generation in the power distribution network: The need for smart grid control systems, communication and equipment for a smart city—use cases. *Renewable and sustainable energy reviews*, Elsevier, v. 38, p. 223–234, 2014.

SHUAI, M. et al. An online approach based on locally weighted learning for short-term traffic flow prediction. In: ACM. *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. [S.l.], 2008. p. 45.

SU, K.; LI, J.; FU, H. Smart city and the applications. In: IEEE. *Electronics, Communications and Control (ICECC), 2011 International Conference on*. [S.l.], 2011. p. 1028–1031.

TIAN, Y.; PAN, L. Predicting short-term traffic flow by long short-term memory recurrent neural network. In: IEEE. *Smart City/SocialCom/SustainCom (SmartCity), 2015 IEEE International Conference on*. [S.l.], 2015. p. 153–158.

VOORT, M. V. D.; DOUGHERTY, M.; WATSON, S. Combining kohonen maps with arima time series models to forecast traffic flow. *Transportation Research Part C: Emerging Technologies*, Elsevier, v. 4, n. 5, p. 307–318, 1996.

WERBOS, P. J. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, IEEE, v. 78, n. 10, p. 1550–1560, 1990.

YEGNANARAYANA, B. *Artificial neural networks*. [S.l.]: PHI Learning Pvt. Ltd., 2009.

ZHANG, J. et al. Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, Institute of Electrical and Electronics Engineers, Inc., 3 Park Avenue, 17 th Fl New York NY 10016-5997 United States, v. 12, n. 4, p. 1624–1639, 2011.