Towards a new intelligent traffic system based on deep learning and data integration

Nadia Slimani[1], Siham Yousfi[2], Mustapha Amghar[1], Nawal Sbiti[1]
[1]Computer Systems and Productivity Team, Computer Science Department in Mohammadia School of Engineering, Mohammed V University, Rabat, Morocco
[2]Meridian Team, LYRICA Laboratory, School of Information Sciences, Mohammed V University, Rabat, Morocco

Article Info

Article history:

Received Sep 27, 2022
Revised Dec 15, 2022
Accepted Dec 17, 2022

Keywords:

Data integration
Deep learning
Image recognition
Intelligent traffic systems
Long short-term memory
Multivariate time series
Traffic forecasting

ABSTRACT

Time series forecasting is an important technique to study the behavior of temporal data in order to forecast the future values, which is widely applied in intelligent traffic systems (ITS). In this paper, several deep learning models were designed to deal with the multivariate time series forecasting problem for the purpose of long-term predicting traffic volume. Simulation results showed that the best forecasts are obtained with the use of two hidden long short-term memory (LSTM) layers: the first with 64 neurons and the second with 32 neurons. Over 93% of the forecasts were made with less than ±2.0% error. The analysis of variances is mainly due to peaks in some extreme conditions. For this purpose, the data was then merged between two different sources: electromagnetic loops and cameras. Data fusion is based on a calibration of the reliability of the sources according to the visibility conditions and time of the day. The integration results were then compared with the real data to prove the improvement of the prediction results in peak periods after the data fusion step.

Corresponding Author:

Nadia Slimani
Computer Systems and Productivity Team, EMI, Computer Science Department in Mohammadia School of Engineering, Mohammed V University
Rabat, Morocco
Email: slimani.nadia@gmail.com

1. INTRODUCTION

Accurate and timely traffic flow information is currently critical to individual travelers, industries and government departments. It has the potential to help road users to make better travel decisions for avoiding accident risks, reducing traffic congestion and carbon emissions, and improving the efficiency of traffic operations. This is an essential step in traffic management to address the dilemma of rising transportation demand and limited land availability [1]. According to the study conducted by Duchenaux *et al.* [2], increasing road capacity does not systematically lead to optimal congestion reduction, and may even causes serious traffic conditions. To achieve better traffic flow prediction performance, many prediction methods have been proposed covering a wide spectrum, such as parametric methods, non-parametric methods and hybrid methods [3]–[8].

Like other industries and fields, transportation is impacted by the emergence of vehicular computing hardware, vehicular and infrastructure sensors and the abundance of information sources. Thus, those advanced technologies have entered the transportation topic in era of intelligent transport system (ITS) and big data. Traffic flow forecasting relies strongly on historical and real-time traffic data of time series collected from various sources. The recent years have seen a significant amount of transportation data collected from multiple sources including electromagnetic loops, radar, cameras, mobile devices. Although there are several techniques and models for predicting traffic flow, most of them are still somewhat unsatisfactory especially for traffic forecasting during peak periods. This prompts us to rethink the traffic flow prediction problem based on deep architecture models coupled with time series analysis and the integration of those systems.

The present work will focus on the prediction problem, through the analysis of multivariate time series (MTS). Unlike univariate time series that summarize only one variable at a time, MTS arises when multiple interconnected streams of data are recorded over time, those are typically produced by devices with multiple (heterogeneous) sensors and are characterized by interactions in time but also between its different dimensions. are among the recent models of MTS [9]. The first one is based on the Fourier transform. The second model is based on random forests allowing the transformation of an MTS into a symbolic representation in the form of a string of characters. This method can suffer from scalability (complexity) problems especially for long time sequences. As a result, the classification of MTS is still an open problem.

Our project's main goal is to forecast the number of vehicles passing in highway through each toll plaza per hour and per payment mode (electronic toll and manual toll) by designing and implementing of a machine learning solution for data integration that allows, from the fusion between two data sources available, to have a more accurate and precise prediction in peak periods. The goal is to enable managers, through those predictions, to manage and control traffic through these projections in order to decrease congestion and thus promote road safety, as well as to anticipate the size of the infrastructures and the resources to be given (reduce or increase the number of lanes and toll collectors).

This article presents a new approach to solve the problem of traffic flow prediction mentioned above, it is articulated as: as an introduction, the importance of an accurate traffic forecasting for financial and security purposes are highlighted. The second part will be devoted to the theoretical concepts of the state of the art of multivariate time series forecasting and the existing machine learning models applied in traffic prediction. Then, several useful add-ons and models will be reviewed in the third part according to their contribution in the process of building up a traffic prediction model. The predictive performance will be compared based on different metrics. The fourth part will review an existing open challenge in the traffic prediction task, which is the need to enhance predictions in peak periods by data integration from multiple sources.

2. THE PROPOSED METHOD

2.1. Dataset overview

Our study is based on real databases provided by an infrastructure manager of highways. The dataset contains traffic flow data coming from two sources:

– Traffic sensors in each toll lane offering the electromagnetic loops data: The dataset contains traffic stream data recorded inside every hour during 4 successive years: 2016, 2017, 2018 and 2019. Through the different sensors installed, the vehicles are classified into three classes based on their length and axles. The following physical and quantifiable factors are used to distinguish classes: i) the length of the vehicle, ii) the number of axles, and iii) the overall height of the vehicle. The meta-data of the files includes the date (day and hour), station name and code. Also, a decomposition of the traffic by the class of the vehicle (lightweight vehicle: class 1, truck: class 2, vehicle with at least three axles: class 3, examples of recorded traffic are presented in Table 1.

Table 1. Dataset example

Station	222	234	237
Day	14/12/2019	03/11/2019	31/12/2019
Hour	3	17	21
ELE_CL1	1	3	18
SUB_CL1	0	1	6
CAD_CL1	0	1	1
CSH_CL1	11	47	49
ELE_CL2	2	8	0
SUB_CL2	4	6	2
CAD_CL2	1	1	4
CSH_CL2	12	25	57
ELE_CL3	3	1	7
SUB_CL3	5	10	2
CAD_CL3	1	7	5
CSH_CL3	2	9	15

– Closed-circuit television (CCTV): Video recordings obtained from two types of cameras. One placed at 20 Km from the toll plaza recording the vehicles passing through this area and the other near each plaza recording the incoming and outgoing vehicles (of which a sample is illustrated in Figure 1).

Figure 1. Overview of a video recording made at the Fez West highway station-Morocco

2.2. Descriptive analysis of the dataset

The boxplot, also called Tukey box presented in Figure 2, is a graphical representation that illustrates the shape and spread of the dataset distribution: the median, quartiles and extreme values. The dataset contains 26,257 observations of traffic flow information about a specific axle in Morocco's highway recorded in four years and segmented by vehicle's classification and payment method. We observe that the total traffic in the studied station varies between 0 and 1,733 vehicles per hour, with a value of median equal to 674 that shares the central box. The Figure 2 represents the boxplot of the sub-series "Cash", "Electronic toll", and "Subscription card" for class 2 vehicles.

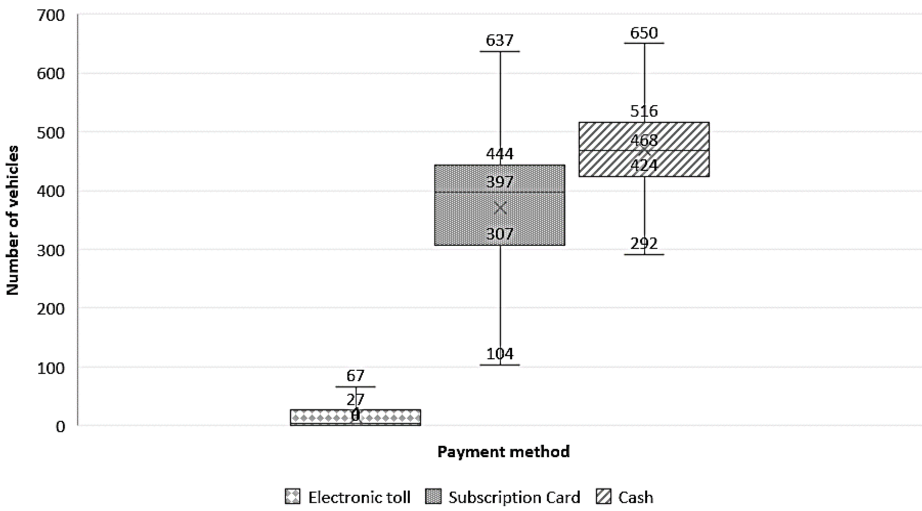

Figure 2. Dataset statistics recorded between 2016 and 2018 for class 2 vehicles

2.3. Background and methodology

2.3.1. Univariate and multivariate time series

This study deals with a time series problem, which are sets of observations that are distinguished by the important role played by the order in which they were collected. The object is the study of variables over time. There are two types of time series [10]:

– Univariate time series: a time series made up entirely of single observations (also known as scalars) that are progressively recorded across equal time intervals. Time is an implicit variable in the time series, despite the fact that a univariate time series data set is typically presented as a single column. The time variable, or index, need not be specified explicitly if the data are equally spaced apart. The series can occasionally be directly plotted using the time variable. However, the time series model itself does not make use of it.

– Multivariate time series: is used to describe a time series with multiple time dependent variables. Each variable has some relationship with other variables in addition to being dependent on its prior values. It is possible to anticipate future values using this reliance. This sort of time series can be split into two categories: those that predict one value from several factors and those that predict two or more values from multiple variables.

2.3.2. Recurrent neural networks

A Recurrent neural networks (RNN) is a class of artificial neural networks (ANN) whose associations between nodes can form a cycle, allowing the output from some nodes to influence the input to other nodes in the same network [11]. The idea of RNN was designed by Jordan in 1980s [12] to exhibit temporal dynamic behavior [13]. The result of traffic prediction tasks depends on both the current situation as well as the previous state. To solve this problem, RNN was developed to represent traffic characteristics without previous information and to explain the impacts from successive traffic records [14]. As a result, it is better able to capture the temporal correlations between the traffic data [15]. Figure 3 illustrates how an RNN is divided into several repetition modules based on time. There is typically only one gate in an RNN repeat module.

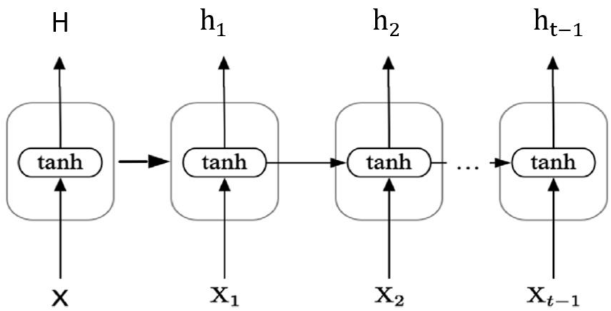

Figure 3. The functioning of RNN

If the input time sequence is presented $X = [x_1, x_2, \ldots, x_T]$, where x_i is the data observed at time i, and the output vector is $H = [h_1, h_2, \ldots, h_T]$, and h_i is the output based on the $x_i th$ input and output in the prior timestamp, which is represented as:

$$h_i = F(W.[h_{i-1}, x_i] + b)$$

where F is the activate function, W is the weight and b is the bias.

We may infer that the outputs of the previous repeat modules as well as the current timestamp input data make up the input set of the current repeat module. Based on the output given by the gate structure contained inside the module, the module evaluates how much of the information in the previous timestamps could have an influence on the current prediction result and how much information should be transited to the following timestamp. The training of RNNs also makes use of back-propagation techniques. There might be a gradient explosion or gradient disappearance issue during the optimization stage if the input series is long enough. To address this problem, the gate control was developed. Using a gating mechanism to govern how information is distributed is the idea behind gate control.

Long-short term memory (LSTM) module was presented in 1997 by Hochreiter and Schmidhuber as a solution to the long-term dependency issue that existed in fundamental RNN [16]. In the recurrent hidden layer of the LSTM are unique components referred as memory blocks. The memory blocks include unique multiplicative units called gates to regulate the information flow as well as memory cells with self-connections that store the network’s temporal state. In the original architecture, each memory block had an input gate and an output gate. The flow of input activations into the memory cell is controlled by the input gate. According to Figure 4, the output gate regulates how cell activations exit into the remainder of the network. The memory block afterwards received the forget gate [17]. Before adding it as input to the cell through the self-recurrent connection, the forget gate scales the internal state of the cell [18], [19]. To simulate the internal calculating process of the LSTM, Zhou *et al.* described four phases in [20]. With, X_t is the input value of the LSTM cell at time t, C_t is the state value memory cell at time t, h_t is the output value at time t, σ denotes the $sigmoid$ activation function and $tanh$ means the $tanh$ activation function.

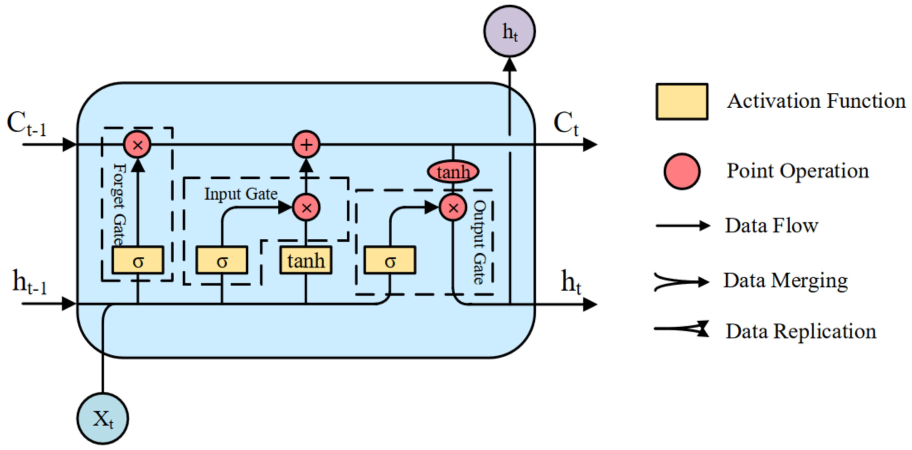

Figure 4. Structure of LSTM

3. RESEARCH MEHOD FOR PREDICTION MODEL

3.1. Feature extraction

The feature extraction step is essential for the realization of deep learning models. For this purpose, we present in Figure 5 a visualization of the major factors that can impact the highway traffic. By analyzing the transactional data offered by electromagnetic loops, we first notice that we are facing a time series problem that changes according to the following logic: i) a decrease in traffic during Ramadan and the day of "*Eid al Adha*" (Muslim celebration of sacrifice); ii) an increase in traffic during Sundays, school vacations-especially the first and last days-, summer vacations, "*Eid al mawlid Nabawi*" (Muslim celebration of the birth of the Prophet), "*Eid al Fitr*" (Muslim celebration of the breaking of the fast) and a period of two days after "*Eid al Adha*".

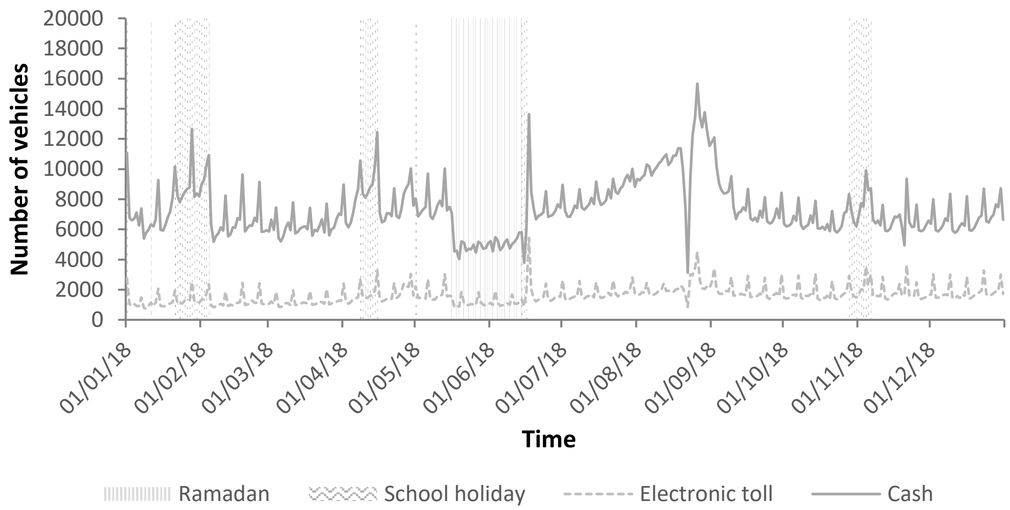

Figure 5. Visualization of some features impacting the behavior of studied time series

3.2. Tools and work environment

For the current study, Python is used through its different libraries for data extraction, creation of regression algorithms and data visualization. This language is very popular and covers a range of usage areas: web development, artificial intelligence (AI), machine learning, operating systems, mobile application development, video games and many others. Besides, the prediction model deployment study is based on TensorFlow which is a computational framework for creating machine learning models. It offers access to community resources that enable developers to quickly create and deploy machine learning-based applications.

3.3. Data processing

The data processing refers to the whole process of collecting, transforming and classifying data. In order to exploit the python libraries, the data is collected, then normalized between 0 and 1 by using the MinMaxScaler class of the Scikit-learn library [21]. Then, the data are sorted in ascending order of day and time and completed by the above-mentioned features as shown in the Table 2.

The implementation of the RNNs models is done according to the process described in Figure 6: after downloading the dataset, we perform a pre-processing that will prepare our data for the following steps. We will establish traffic-impacting features based on a visual analysis of the traffic time function, and then proceed to data normalization which reduces the complexity of the models. The next step defines the inputs and outputs, then divide the dataset into two parts: 80% of the first observations as training data, and the remaining 20% as model test data and transform the time series problem into a supervised learning problem in order to train the created deep learning models.

Table 2. Number of vehicles and some features for the day of April 17th, 2017

		Hours				
		14:00	15:00	16:00	17:00	18:00
Number of vehicles	Electronic	3	4	15	34	28
	Cash	468	447	446	545	483
Features	Sunday	0	0	0	0	0
	Weekday	1	1	1	1	1
	School vacation	0	0	0	0	0

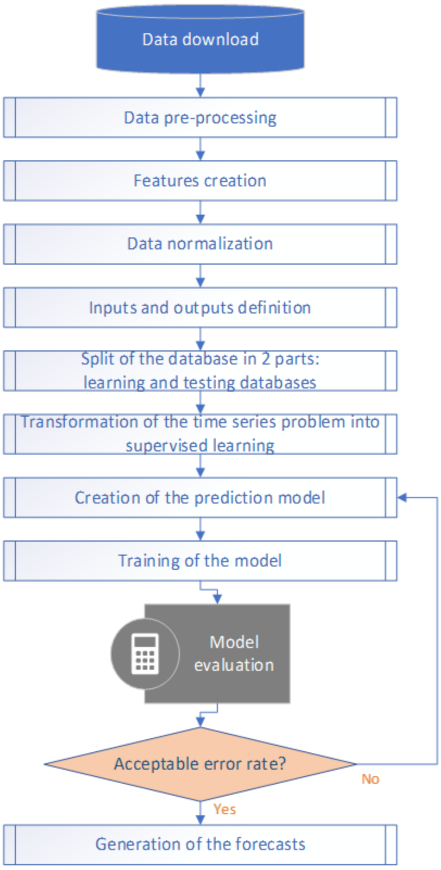

Figure 6. Steps for RNN model design

4. RESULTS AND DISCUSSION

After having tested several combinations of architectures applied to the LSTM model, the Table 3 summarizes a comparative of the obtained results. Finding the model that best depicts our data and predicts how the model will perform in the future is an unavoidable step. The architecture which recorded the best

performances consists of 14 inputs, two hidden LSTM layers and two outputs. The hidden layers are composed of 64 and 32 neurons respectively as shown in Figure 7. This finding was established after the use of three commonly used error measures to evaluate the performance of a regression model:
- Mean square error (MSE): the average of the squared differences between the predicted and expected target values in a data set. It can be calculated using the mean squared error function in the scikit-learn library.
- The root mean square error (RMSE): an extension of the MSE. The link between these two quantities is explained:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 \quad \text{and} \quad RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2}$$

with $\hat{Y}_i$ is the predicted value, Y_i is the recorded value, and N is number of observations.

Table 3. Comparative of some different architectures applied to the LSTM model with 2 hidden layers

Number of neurons	Activation function	Dropout layer	RMSE Electronic	RMSE Manual
128, 64	ReLU	0.1	21.17	22.79
		0.2	18.19	20.61
	Softmax	0.1	23.23	26.24
		0.2	25.28	20.11
64, 32	**ReLU**	**0.1**	**14.05**	**18.58**
		0.2	18.83	20.13
	Softmax	0.1	18.87	22.35
		0.2	24.18	20.71

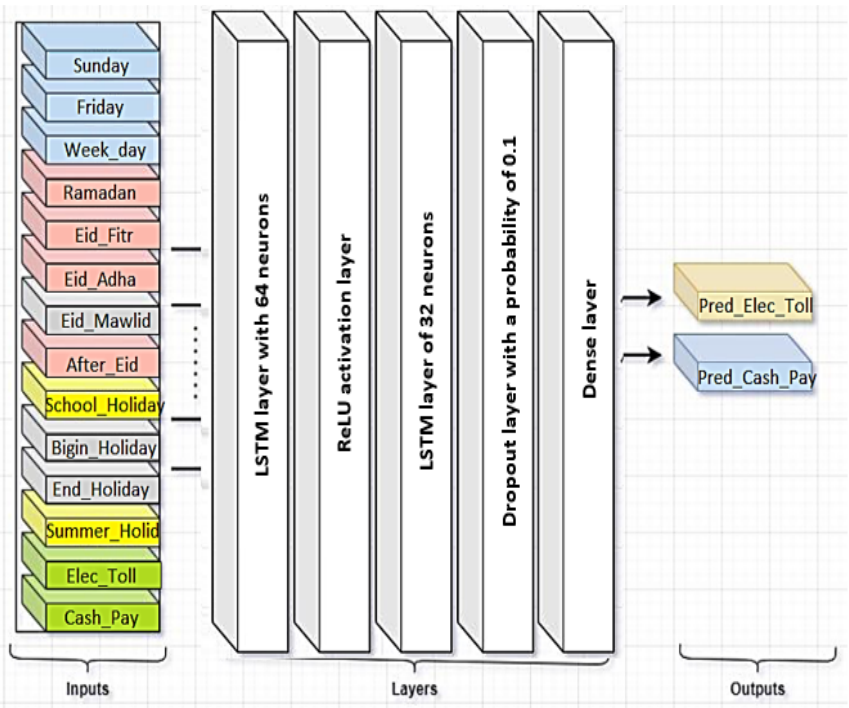

Figure 7. Architecture of RNN model

Figures 8 and 9 show a temporal representation of predicted and actual traffic, the predicted values are plotted in orange, and the actually recorded values are represented by the blue color. We notice that the two series overlap in the majority of the points, except during the peak traffic periods. Thus, over 93% of the forecasts were made with less than ±2.0% error. This discrepancy is mainly due to the quality of the model's predictions during peak periods, which we propose to improve in the following section by integrating traffic data from different data sources.

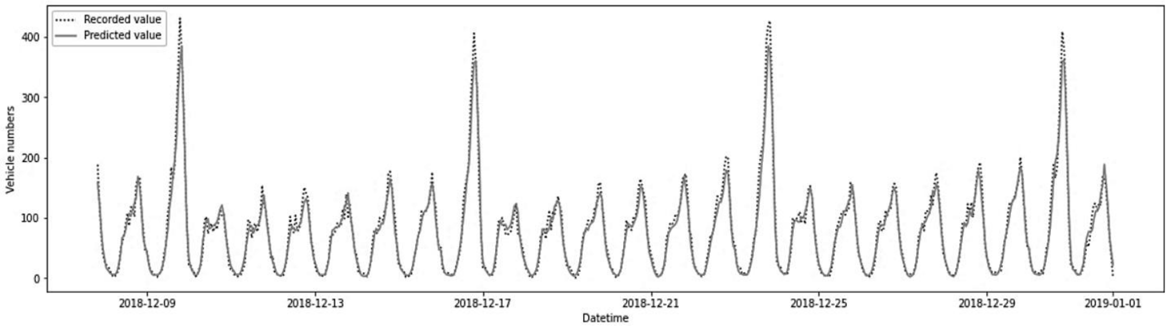

Figure 8. Plot of predicted and actual values for electronic toll collection, LSTM

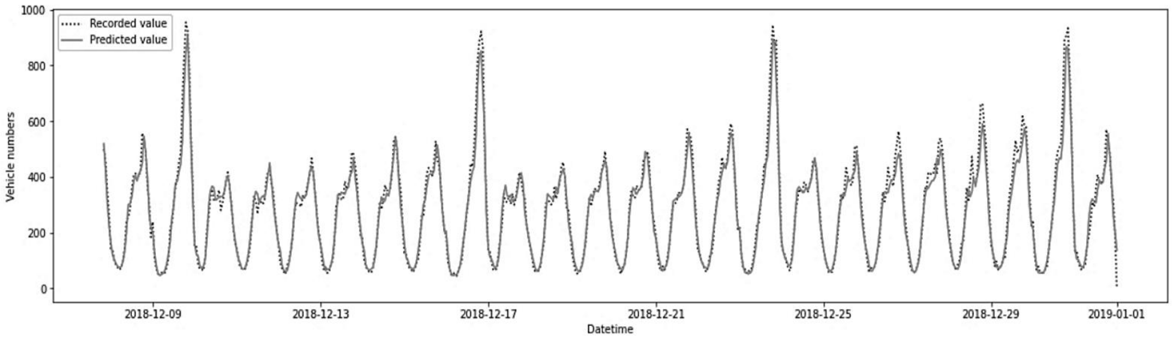

Figure 9. Plot of predicted and actual values for manual toll collection, LSTM

4.1. Improving results in peak periods by multiform collection and data fusion in highway context

Data integration aims to provide users with a uniform picture of the data and unified access to data that needs information from many sources [22]. Ziegler explains the reasons behind the value of integrating data from different sources [23], which are twofold: i) reliability of information by comparing and merging it with a collection of current information systems and ii) a deeper comprehension of the issue through the synthesis of facts from many sources into pertinent and useful knowledge.

A complicated framework with a variety of approaches is frequently needed for effective data integration from heterogeneous large volumes of data [24]. Such a framework's construction would be challenging. Factors like redundancy, reliability and severity [25] have to be considered and validated for combining data from disparate sources [26] into meaningful and more accurate predictions of traffic flow [27]. Furthermore, few studies concerning data integration in a real highway context exist prior to this present work, revealing a gap in the scientific field, especially the management of traffic peaks remains an open challenge [28]. Figure 10 shows the scheme pursued for data integration.

Two processing engines are necessary for the execution of our framework: the first is the video handling motor, which examines the CCTV recordings, and the second is the log handling motor, which checks electromagnetic loop data. Three times per day, the video streams were recorded (morning, afternoon and night). Utilizing you only look once (YOLOv4) software for real-time object detection system [29], as shown in Figure 11, we can detect cars with the best speed and accuracy. Utilizing deep SORT [30], the vehicles can be tracked. A sample of the result from the video processing is shown in Figure 12.

According to the previous sections, it was noted that the data collected from electromagnetic loops can provide imperfect or erroneous predictions during potential peak traffic days (Sundays, vacations…). In the other side, the performance of video data varies depending on the time of day, visibility and camera position. In order to improve prediction results in peak periods, we thought of integrating these two complementary data sources. For this purpose, we create three features: i) visibility: the creation of this characteristic starts with the collection of the weather data, as showed in Figure 13; ii) time of day (day=1 or night=0); and iii) probable peak days: Sunday and first and last day of vacations and celebrations.

Depending on visibility and traffic peak conditions, we rate the dependability of each data source. For instance, the video has a high "Weight Vid" index on sunny days and a lower value on nights when it is raining. Similar to electromagnetic loops data, electromagnetic loops data has a high "Weight Tran" index during off-peak times and a low weight at peak times. The function designed to distribute weights across those data sources is depicted in Figure 14.

The established model offers generally accurate predictions based on loop data, with the exception of traffic peaks brought on by particular unique events, such as religious holidays, cultural or sporting events, and travel restrictions following a health emergency. While the video processing engine offers lower error rates during peak hours, these are the only times when visibility is reduced due to opacity at night, rain, or fog. The instances where both sources simultaneously provided inaccurate information are rare, though. Consequently, there is interest in combining these complementary data sources to enhance traffic flow predictions as shown in Table 4.

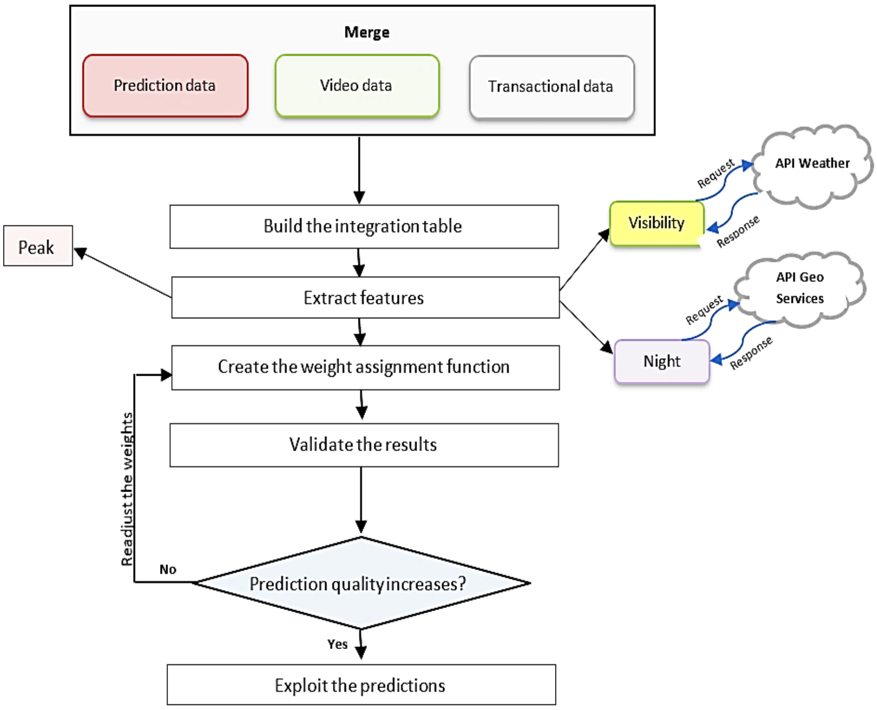

Figure 10. Data integration process

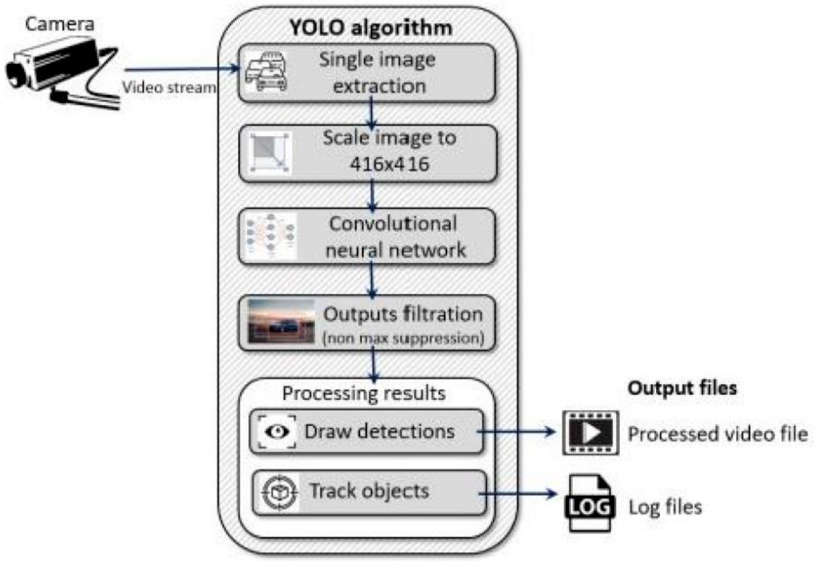

Figure 11. Illustration of YOLO algorithm

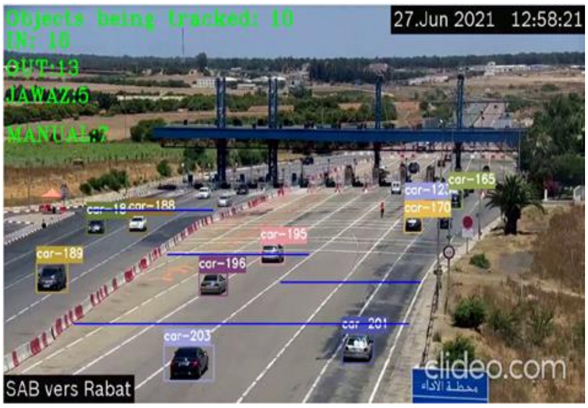

Figure 12. Utilizing deep sort and YOLOv4 to track and detect vehicles


```
{...
"weather": [{
        "id": 520,
        "main": "Rain",
        "description": "light intensity
shower rain",
        "icon": "09n"
        }],
...
"clouds": {
        "all": 75
        },
...}}
```


Figure 13. Data sent by OpenWeatherMap API

```python
def affect_weights(df):
  for i in range(len(df)):
    if df['Nuit'][i] == 1 and df['Visibilite'][i] == 0
      df['Poids_pred'][i] = 0.8
      df['Poids_vid'][i] = 0.2
    if df['Nuit'][i] == 1 and df['Visibilite'][i] == 1
      df['Poids_pred'][i] = 0.6
      df['Poids_vid'][i] = 0.4
```

Figure 14. Assigning weights between data sources

Table 4. A sample of feature fusion for the day of August 28th, 2021 for station 222

Time	TELEP_Tran	TELEP_vid	TELEP_real	Visibility	Peak	Night	Weight_Tran	Weight_Vid
06:00	402	410	410	1	0	0	0,8	0,2
12:00	325	340	346	0	1	0	0,6	0,4
21:00	841	895	897	1	1	1	0,4	0,6

Finally, we assess the performance forecasts using data from electromagnetic loops and videos. The peaks shown in loop data and the low visibility in videos are to blame for the differences seen between the predicted values and the actual values. Integration makes it possible to select a dependable data source, which enhances the accuracy of predictions as shown in Table 5.

Table 5. RMSE of electromagnetic loops and video data integration forecasts

	RMSE Manual toll	RMSE Electronic toll
Loops data	34.43	94.26
Video data	93.96	70.25
Integration	27.95	10.27

5. CONCLUSION

This paper presents several long-term prediction models based on recurrent neural networks with a real traffic data set. The architecture which recorded the best performances consists of 14 inputs, two hidden LSTM layers with 64 and 32 neurons respectively and two outputs. Thus, over 93% of the forecasts were made with less than ±2.0% error. This discrepancy is mainly due to the quality of the model's predictions during peak periods. In order to improve the prediction results especially under abnormal traffic conditions, we perform a smart system for data integration from heterogeneous data sources (loops and camera) that aims to provide the most correct and complete as possible traffic predictions. This paper verifies the robustness of the prediction model built, in various weather conditions. This work paves the way for several lines of research such as adaptive model learning and the progressive incorporation of models based on large databases which are present in a small proportion of the reviewed works and which require real-time updating of traffic data, which implies updating of model parameters and better consideration of exogenous factors.

REFERENCES

[1] A. D. Joseph *et al.*, "Intelligent transportation systems," *IEEE Pervasive Computing*, vol. 5, no. 4, pp. 63-67, Oct.-Dec. 2006, doi: 10.1109/MPRV.2006.77.

[2] E. Dechenaux, S. D. Mago, and L. Razzolini, "Traffic congestion: an experimental study of the Downs-Thomson paradox," *Experimental Economics*, vol. 17, no. 3, pp. 461–487, Sep. 2014, doi: 10.1007/s10683-013-9378-4.

[3] N. Slimani, I. Slimani, N. Sbiti, and M. Amghar, "Traffic forecasting in Morocco using artificial neural networks," *Procedia Computer Science*, vol. 151, pp. 471–476, 2019, doi: 10.1016/j.procs.2019.04.064.

[4] N. Slimani, I. Slimani, M. Amghar, and N. Sbiti, "Road traffic forecasting using a real data set in Morocco," *Procedia Computer Science*, vol. 177, pp. 128–135, 2020, doi: 10.1016/j.procs.2020.10.020.

[5] I. Slimani *et al.*, "Automated machine learning: the new data science challenge," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 4, p. 4243, Aug. 2022, doi: 10.11591/ijece.v12i4.pp4243-4252.

[6] N. Slimani, I. Slimani, N. Sbiti, and M. Amghar, "Machine learning and statistic predictive modeling for road traffic flow," *International Journal of Traffic and Transportation Management*, vol. 03, no. 01, pp. 17–24, Mar. 2021, doi: 10.5383/JTTM.03.01.003.

[7] T. Alghamdi, S. Mostafi, G. Abdelkader, and K. Elgazzar, "A comparative study on traffic modeling techniques for predicting and simulating traffic behavior," *Future Internet*, vol. 14, no. 10, Oct. 2022, doi: 10.3390/fi14100294.

[8] N. Slimani, M. Amghar, and N. Sbiti, "Deep learning and time series analysis application on traffic flow forecasting," *Journal of Theoretical and Applied Information Technology*, no. 5, p. 18, 2022.

[9] M. A. Castán-Lascorz, P. Jiménez-Herrera, A. Troncoso, and G. Asencio-Cortés, "A new hybrid method for predicting univariate and multivariate time series based on pattern forecasting," *Information Sciences*, vol. 586, pp. 611–627, Mar. 2022, doi: 10.1016/j.ins.2021.12.001.

[10] L. Chen, D. Chen, Z. Shang, Y. Zhang, B. Wen, and C. Yang, "Multi-scale adaptive graph neural network for multivariate time series forecasting," *arXiv preprint arXiv:2201.04828*, Jan. 2022.

[11] H. Eskandari, M. Imani, and M. P. Moghaddam, "Convolutional and recurrent neural network based model for short-term load forecasting," *Electric Power Systems Research*, vol. 195, Jun. 2021, doi: 10.1016/j.epsr.2021.107173.

[12] M. I. Jordan, "Serial order: a parallel distributed processing approach," in *Neural-Network Models of Cognition-Biobehavioral Foundations*, Elsevier, 1997, pp. 471–495.

[13] M. Bod´en, "A guide to recurrent neural networks and backpropagation," *The Dallas project*, vol. 2, no. 2, pp. 1–10, 2002.

[14] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.

[15] Y. Tao, P. Sun, and A. Boukerche, "A delay-based deep learning approach for urban traffic volume prediction," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, Jun. 2020, pp. 1–6, doi: 10.1109/ICC40277.2020.9149018.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[17] X. Wan, H. Liu, H. Xu, and X. Zhang, "Network traffic prediction based on LSTM and transfer learning," *IEEE Access*, vol. 10, pp. 86181–86190, 2022, doi: 10.1109/ACCESS.2022.3199372.

[18] G. E. Dahl, Dong Yu, Li Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, Jan. 2012, doi: 10.1109/TASL.2011.2134090.

[19] W. Zhuang and Y. Cao, "Short-term traffic flow prediction based on CNN-BILSTM with multicomponent information," *Applied Sciences*, vol. 12, no. 17, Aug. 2022, doi: 10.3390/app12178714.

[20] S. Zhou *et al.*, "A hybrid deep learning model for short-term traffic flow pre-diction considering spatiotemporal features," *Sustainability*, vol. 14, no. 16, Aug. 2022, doi: 10.3390/su141610039.

[21] M. Kamiński and K. Szabat, "Adaptive control structure with neural data processing applied for electrical drive with elastic shaft," *Energies*, vol. 14, no. 12, Jun. 2021, doi: 10.3390/en14123389.

[22] H. Dhayne, R. Haque, R. Kilany, and Y. Taher, "In search of big medical data integration solutions-a comprehensive survey," *IEEE Access*, vol. 7, pp. 91265–91290, 2019, doi: 10.1109/ACCESS.2019.2927491.

[23] P. Ziegler and K. R. Dittrich, "Data integration-problems, approaches, and perspectives," in *Conceptual Modelling in Information Systems Engineering*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 39–58.

[24] Z. He, L. Zhang, H. Zhang, and X. Liu, "Integrated modeling and control of urban road traffic networks," in *2017 6th Data Driven Control and Learning Systems (DDCLS)*, May 2017, pp. 733–738, doi: 10.1109/DDCLS.2017.8068164.

[25] S. Yousfi, D. Chiadmi, and M. Rhanoui, "Smart big data framework for insight discovery," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 10, pp. 9777–9792, Nov. 2022, doi: 10.1016/j.jksuci.2021.12.009.

[26] I. Eriksson, "Towards integrating crowdsourced and official traffic data." Uppsala Universitet, 2019.

[27] D. Mierzwinski, D. Walczak, M. Wolski, and M. Wrzos, "Video and sensor data integration in a service-oriented surveillance system," *Scalable Computing: Practice and Experience*, vol. 12, no. 1, pp. 93–104, 2011.

[28] L. Shen, J. Lu, D. Geng, and L. Deng, "Peak traffic flow predictions: exploiting toll data from large expressway networks," *Sustainability*, vol. 13, no. 1, Dec. 2020, doi: 10.3390/su13010260.

[29] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[30] R. Pereira, G. Carvalho, L. Garrote, and U. J. Nunes, "Sort and deep-SORT based multi-object tracking for mobile robotics: evaluation with new data association metrics," *Applied Sciences*, vol. 12, no. 3, Jan. 2022, doi: 10.3390/app12031319.

BIOGRAPHIES OF AUTHORS

Nadia Slimani Engineer graduated from Mohammedia School of Engineering (EMI) in Rabat and currently Ph.D. student in Mohammedia School of Engineering. She has a long experience with a national transport operator in Morocco. She also attended the higher cycle of management at the ISCAE of Rabat. Her research and publication interests include machine learning, traffic forecasting and intelligent transportation systems. She can be contacted at email: slimani.nadia@gmail.com. https://www.researchgate.net/profile/Nadia-Slimani-2.

Siham Yousfi holds a Ph.D. in computer science from Mohammedia School of Engineering (EMI) in Rabat and currently Professor in School of Information Science and a member of Computer Systems and Productivity Team (EMI) and Meridian Team (ESI). In addition, she is a reviewer for leading computer science journals. She can be contacted at email: siham.yousfi@gmail.com. https://www.researchgate.net/profile/Siham-Yousfi.

Mustapha Amghar holder of a Ph.D. in industrial computing from Liege University in Belgium as well as a university habilitation in research from the Mohammedia School of Engineering in Morocco. He is currently Professor in Electrical Department at Mohammedia School of Engineering. For more than 30 years, he has been carrying out consulting missions for public and private organizations. He can be contacted at email: amghar@emi.ac.ma. https://www.researchgate.net/profile/Mustapha-Amghar.

Nawal Sbiti holder of a Ph.D. in automation and industrial computing from Mohammedia School of Engineering in Morocco as well as a university habilitation in research from the same institution. She is currently Professor in Electrical Department at Mohammedia School of Engineering. For more than 30 years, she has been carrying out consulting missions for public and private organizations. She can be contacted at email: sbiti@emi.ac.ma.