2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application

Short-Term Traffic Flow Forecast Using Regression Analysis and Graph Convolutional Neural Networks

Daniel Klosa
Center for Industrial Mathematics
University of Bremen
Bremen, Germany
dklosa@uni-bremen.de

Amin Mallek
Center for Industrial Mathematics
University of Bremen
Bremen, Germany
amallek@uni-bremen.de

Christof Büskens
Center for Industrial Mathematics
University of Bremen
Bremen, Germany
bueskens@math.uni-bremen.de

***Abstract*—Short-term forecast of different traffic attributes is one of the fundamental tools used in transportation planning. Precisely, accurate predictions of traffic flow are often the basis of an efficient traffic management. In the paper at hands we shed a light on short-term traffic flow forecast in urban arterial roads in Bremen (Germany). This case-study uses real-data collected from 7 loop detectors installed in downtown. To deal with this, we propose two different models, namely, Linear Regression and Graph Convolutional Neural Networks. The models are separately applied to 11 weeks of data, three of these are dedicated to test the performance of both models. Experimental results show that the models are closely competitive and they reach, in average over the whole test-set, around 19% mean absolute percentage error during morning and evening peak times.**

***Index Terms*—Traffic Flow; Regression Analysis; Graph Convolutional Neural Networks; Short-term Forecast; Deep Learning; Machine Learning.**

I. Introduction

Intelligent transportation systems (ITS) play a significant role in organizing, controlling and planning the traffic flow inside or outside of cities. Basically, smart cities heavily rely on those systems to optimize the circulation process and render city districts easily and quickly accessible from everywhere. One of the main elements ITS is concerned with is traffic flow volume in different city roads and further highways surrounding the city. Various tools are usually used to measure and monitor the traffic flow including inductive loop detectors, video image processing, microwave and laser radars and other techniques related to Internet of Things (IoT). The data collected by the aforementioned tools (which is usually Big Data), is often used to build different types of models for analysis and prediction purposes. Many researchers investigated broad topics of Intelligent Transportation with a major focus on predicting the key factors that impact the traffic, for instance, traffic flow volume, traffic speed, traffic state, etc. In the present paper we draw attention to short-term traffic flow volume forecasting. Accurate prediction of this key element is a stepping stone for other kinds of tasks such as travel time forecast and traffic state prediction.

In our study, we use two different methods to deal with short-term forecasting of traffic flow volume in Bremen city (Germany). This is part of a project we conduct to model and forecast traffic flow in the city, wherein the chief goal is to reduce CO_2 emissions in Bremen by controlling traffic lights. In order to capture the linear relations in the working data, we designed a Regression Analysis model based on different time frames aiming to learn the linear dependencies between data points. To cope with the non-linear characteristics in the data, a Graph Convolutional Neural Network is used. For the time being, both models were separately applied to data collected from 7 inductive loop detectors installed around urban arterial roads in downtown. The purpose of this paper is to compare the proposed models, evaluate the prediction results obtained so far and give insights on some future work. The rest of this article is structured as follows. We review some related work in Section II. In Section III, we detail the models devised in the paper. Afterwards, the used data is described in Section IV. Results and discussions are reported in Section V. The paper is concluded with some future directions and perspectives in Section VI.

II. Literature review

The problem we consider has gained a lot of interest among researchers and has been investigated in many papers with different approaches. One of the common approaches often used when one deals with time series is ARIMA and its variants. This model has been solely applied as in [9] and also hybridized with some other approaches such as in [15]. Moreover, other techniques such as Support Vector Regression (SVR) [11], [3], K-Nearest Neighbors (KNN) [2], [17], [19] and several hybridized methods [10], [18] were also explored. However, most recently researchers focus more on Deep Learning methods to improve forecasting precision.

Funded by the European Regional Development Fund (ERDF).
European Union
Investing in Bremen's Future
European Regional
Development Fund

978-1-6654-9457-1/21/$31.00 ©2021 IEEE
DOI 10.1109/HPCC-DSS-SmartCity-DependSys53884.2021.00212

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:39:01 UTC from IEEE Xplore. Restrictions apply.
2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application

Some of the early Deep Learning models such as stacked auto encodes (SAE) [12], long-short term memory (LSTM) and gated recurrent units (GRU) [5] used only time series data in the prediction process. Nevertheless, recent models employ more advanced techniques as in [7], where Traffic-Wave is used. The authors stack 1d dilated convolutions to increase the receptive field of the convolutions exponentially, which allows their model to extract short-term patterns in lower layers and long-term patterns in higher layers, outperforming the previous approaches. In Graph Wave-Net, as in [14], the Traffic-Wave idea is extended by implementing graph convolutions to capture spatial dependencies. Most recent Deep Learning techniques for traffic forecasting incorporate graph convolutions [1], [4], [6]. For an overview, the reader is referred to a survey that has been carried out in [8].

III. Methodology

This section comprises a detailed description of the forecasting models designed in the current article.

A. Regression Analysis

Linear Regression (precisely Polynomial Regression in our case), is a modeling approach that captures a certain relationship between variables, usually known as independent variables or predictors (input) and dependent variables (output). In cases when we deal with only one input variable the model is called Simple Linear Regression, however, when the number of independent variables escalates, we refer to the model as Multiple Linear Regression. Another variant of linear regression models, different than the latter, named Multivariate Linear Regression, also exists in the literature and is often involved when multiple dependent variables (output) are to be predicted. The relationships between the different variables are modeled using linear predictor functions whose unknown model parameters are generally estimated from the data.

Linear Regression models have been extensively used in various areas for multiple goals. Here we are interested in their applications in the field of Machine Learning where the main aim is to build a predictive model from an observed data-set, then use the learnt model to predict or forecast future (unseen) values. The general equation of a polynomial regression of degree n can be written (in two forms) as follows:

$$P(x) = \sum_{k=0}^{n} \alpha_k x^k + \epsilon = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \ldots + \alpha_n x^n + \epsilon$$

$P(x)$ and x are respectively the dependent variable and the independent variable. α is the vector of the corresponding coefficients to be learnt from the data, and ϵ is a random error term with mean zero; added for bias.

Our model is devised in 2-phases, the first one is to establish a Regression Analysis, then subdue the model parameters, in a second part, to a correction or a validation process in which the values are slightly modified. The regression model considers each week-day separately, for each of which a polynomial of degree 10 is learnt, this yields 7 functions. Moreover, each of the week-days is split up into 24 hours, to each hour a polynomial of degree 5 is associated. Thus, in total we have 175 functions (168+7). The Regression Analysis takes into account the mean of the values at a given instant over the training set, then excludes the values that are above $1.5 \cdot mean$ since polynomial regression is very sensitive to outliers. The shape of the daily regression is smoothly captured with degree 10 with no over-fitting. Additionally, degree 5 is chosen for the hourly regression as an attempt to grasp the shape of the fluctuations and try to match with it. The predicted flow values at a given instant are calculated as follows:

$$F(t) = \frac{4 \cdot F_{week-day}(t) + F_{hour}(t)}{5}$$

where $F(t)$ is the predicted traffic flow volume at instant t, $F_{week-day}(t)$ and $F_{hour}(t)$ are respectively the regression polynomials corresponding to the week-day and the hour of instant t. The correction phase gives high priority to the parameters learnt by the regression model, however it tries to make use of one week values and combine them with the regression values as follows:

$$Pred(t) = \frac{4 \cdot V_{reg}(t) + V_{week}(t)}{5}$$

Where $V_{week}(t)$ are the values at each instant t of the last week-days of the training set and $V_{reg}(t)$ are the output of the regression model at the same instant t. Experimentally, this latter showed a slight improvement in the prediction accuracy.

The built-up model has the ability to do short and long term forecasts very quickly, alongside that, it can be trained with a small amount of data. In contrast, the main drawbacks of the model are that it cannot accurately predict the sharp fluctuations without further data feeding and also is unable to predict any strange pattern that was not captured during the training phase.

B. Graph Convolutional Neural Network

The Graph Convolutional Neural Network (GCNN) used herein is based on the Global Spatial-Temporal Graph Convolutional Network (GSTGCN) by Liang et al. [6]. The architecture used in the present work is illustrated in Figure 1. The three temporal modules are each made up of $N_R = 3$ residual blocks incorporating dilated 1d convolutions (DCC) with dilation factors $d = 1, 2, 4$, in the same way as in [16], for learning short-term and long-term relationships in the temporal space. Dilated convolution blocks were also used in Traffic-Wave [7] and Graph Wave-Net [14]. They are followed by a weight normalization operation (WN) [13] to combat overfitting and a ReLU activation.

We use three input segments for each prediction; Let $x_{t_0} \in \mathbb{R}^{N \times D}$ be the traffic state at the current timestamp t_0, where $N = 7$ is the amount of sensors, $D = 1$ is

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:39:01 UTC from IEEE Xplore. Restrictions apply.

the amount of features (in this case flow) and $T_p \in \mathbb{N}$ is the amount of timestamps to be predicted. The recent-time segment $X_{recent} = (x_{t_0-T_h+1}, x_{t_0-T_h+2}, \ldots, x_{t_0}) \in \mathbb{R}^{N\times D\times T_h}, T_h \in \mathbb{N}$ includes the most recent timestamps and it is used to adapt to the current traffic state. The daily-periodic segment $X_{daily} \in \mathbb{R}^{N\times D\times T_d}$, $T_d \in \mathbb{N}$ incorporates the same segment to be predicted but on the previous $\frac{T_d}{T_p} \in \mathbb{N}$ days. This segment is chosen because traffic can be similar on consecutive days. The weekly-periodic segment $X_{weekly} \in \mathbb{R}^{N\times D\times T_w}, T_w \in \mathbb{N}$ also contains the same segment to be predicted but from the previous $\frac{T_d}{T_p} \in \mathbb{N}$ weeks on the same weekday. It makes use of repeating patterns in the historical data since the flow on the same weekday is generally similar. The three temporal modules are used for learning recent, daily-periodic and weekly-periodic features $Y_{recent}, Y_{daily}, Y_{weekly} \in \mathbb{R}^{N\times F\times T_p}$, where $N = 7$ is the amount of sensors, $F \in \mathbb{N}$ is the amount of features and $T_p \in \mathbb{N}$ is the amount of timestamps to be predicted. The spatial modules consist of a graph convolution incorporating the distance between sensors and a global correlated spatial mechanism based on the connectivity within the traffic network. The adjacency matrix deployed in the graph convolution is based on driving distances in between sensors. Accordingly, Dijkstra algorithm is applied to compute the driving distance $dist(i, j)$ from sensor i to sensor j. For the global correlated spatial mechanism, the connectivity $conn(i, j)$ of two sensors i and j is set to $conn(i, j) = \alpha = 2$ if there are at most two edges in between sensors i and j, otherwise $conn(i, j) = 1$. The resulting features of the three time-frames are fused and a fully connected neural network is applied. While Liang et al. in [6] use GSTGCN to predict traffic speed, the adapted architecture demonstrates through experimental testing that it is also capable to predict traffic flow.

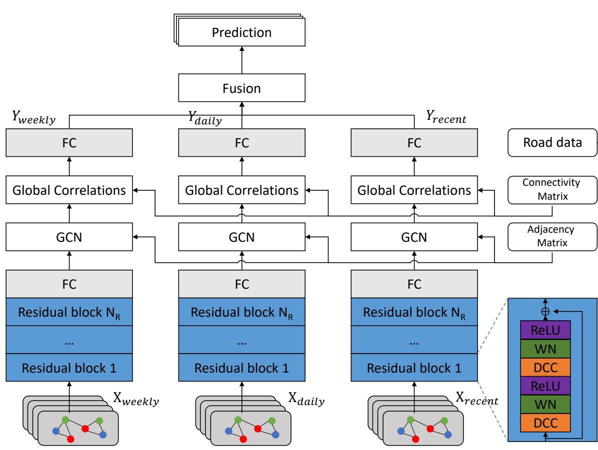

Fig. 1: The architecture of our Graph Convolutional Neural Network.

IV. Data description

The present paper is part of a project currently conducted at the Center for Industrial Mathematics (ZeTeM) of the University of Bremen, to model and forecast traffic flow in the city of Bremen (Germany). The project is launched in the context of the increased efforts to combat climate change. As stated above, the main goal of this project is to forecast a short-term traffic state to allow decision-makers to take actions in order to reduce CO_2 emissions due to traffic. The Traffic Management Center (VMZ) of Bremen is an associated partner in the project and it is the main provider of the data we use. Figure 2 presents Bremen city map and displays in red bullets the location of the detectors installed all over the city to record traffic data.

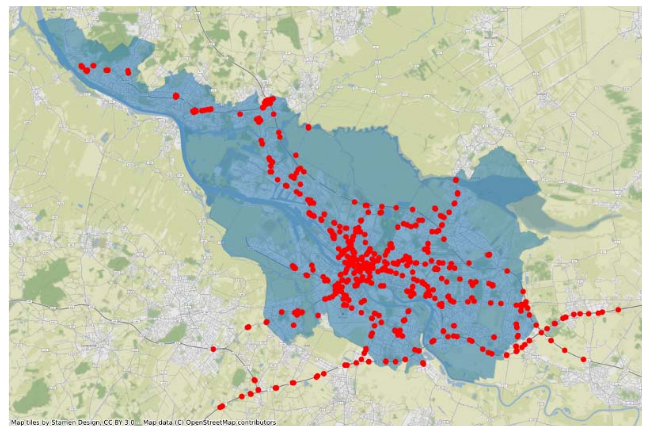

Fig. 2: Loop detectors installed in the city of Bremen.

In the study reported in this paper, we focus on probably the most congested area in the city: downtown. We selected a junction in front of the main train station (Bremen Hauptbahnhof), tram station, bus station, etc. The chosen junction is itself a

Fig. 3: The chosen junction (Red bullets correspond to the detectors location).

big challenge because of the large amount of traffic lights and signs present in this area, about which, unfortunately, we don't

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:39:01 UTC from IEEE Xplore. Restrictions apply.

possess any data. In that location 7 inductive loop detectors (form MS217 to MS223) are installed in the surroundings to collect traffic data as shown in Figure 3.

Measurements take place every 90 seconds, however, in our study we use 10-minutes accumulation. The sample data we consider in the present work includes the time frame from 9 April 2018 to 24 June 2018. The chosen dates correspond to the period with the least missing data (less than 2% of the data is missing). The used data covers a period of 11 weeks split up into 8 weeks for training the models and 3 weeks for testing. Precisely, the models are trained with data from 9 April 2018 to 3 June 2018 and tested for the period going from 4 June 2018 to 24 June 2018. Figures 4 and 5 exhibit random samples of data per day and per week.

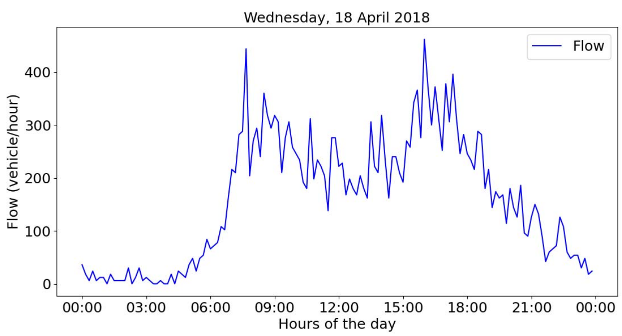

Fig. 4: One-day traffic flow data (detector MS223).

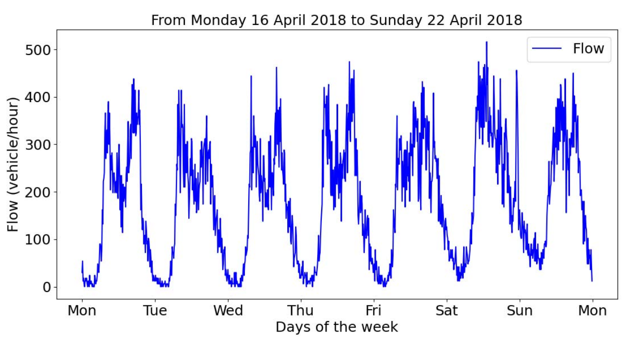

Fig. 5: One-week traffic flow data (detector MS223).

Since the data is collected from detectors placed in signalized urban arterial roads, the fluctuations in the data are very sharp and can jump between extreme values in short periods of time. In our project, we focus on different attributes related to traffic and transportation, however, in this paper we discuss only traffic flow forecast. Note that traffic flow volume is given by the number of vehicles traversing a detector per hour.

V. Results and discussions

In this section, we report and discuss the output of the implemented models. Before doing so, in order to evaluate the accuracy of the predictions made by both models, two metrics are deployed: Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE). These are given by the following formulas:

$$MAPE = \frac{100}{n} \cdot \sum_{i=1}^{n} \left| \frac{m(t)_i - p(t)_i}{m(t)_i} \right|$$

$$MAE = \frac{1}{n} \cdot \sum_{i=1}^{n} |m(t)_i - p(t)_i|$$

such that $m(t)$ is the real value of traffic flow measured at instant t and $p(t)$ is the value predicted by the model. n is the number of predictions.

TABLE I: Prediction results over the test-set considering morning peak hours only (06:00-09:00).

	MAPE(%)		MAE	
Detector ID	Reg. Ana.	GCNN	Reg. Ana.	GCNN
MS217	15.50	16.11	78.93	65.65
MS218	25.27	34.15	25.39	28.14
MS219	16.50	13.74	75.86	56.65
MS220	20.09	20.02	77.47	73.36
MS221	22.28	30.34	26.11	32.07
MS222	22.21	27.08	39.50	40.20
MS223	23.46	20.93	53.56	47.12
Average	20.75	23.19	53.83	49.02

TABLE II: Prediction results over the test-set considering evening peak hours only (16:00-19:00).

	MAPE(%)		MAE	
Detector ID	Reg. Ana.	GCNN	Reg. Ana.	GCNN
MS217	11.77	11.77	96.81	96.02
MS218	20.85	19.90	58.78	48.26
MS219	11.04	11.22	84.80	86.62
MS220	17.19	15.38	79.51	71.50
MS221	19.81	18.98	57.56	55.46
MS222	20.01	19.48	61.91	59.05
MS223	18.64	17.31	52.98	48.67
Average	17.04	16.29	70.33	66.51

TABLE III: Prediction results over the test-set considering all-day hours.

	MAPE(%)		MAE	
Detector ID	Reg. Ana.	GCNN	Reg. Ana.	GCNN
MS217	15.81	17.03	63.05	61.95
MS218	28.30	32.13	28.81	29.02
MS219	18.01	19.62	60.22	58.12
MS220	24.19	25.19	61.39	58.95
MS221	29.49	29.69	38.98	36.67
MS222	29.67	30.42	39.91	38.15
MS223	27.91	27.80	37.92	34.99
Average	24.76	25.98	47.18	45.40

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:39:01 UTC from IEEE Xplore. Restrictions apply.

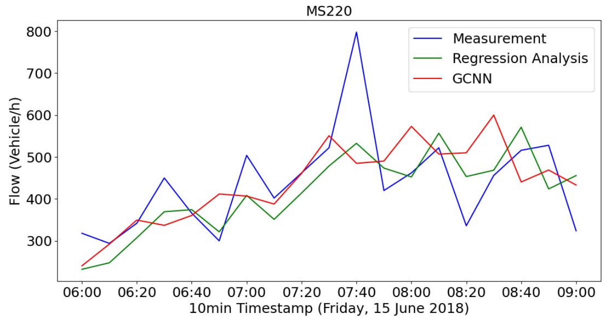

Fig. 6: Flow prediction in morning peak on a randomly chosen working-day (detector MS220)

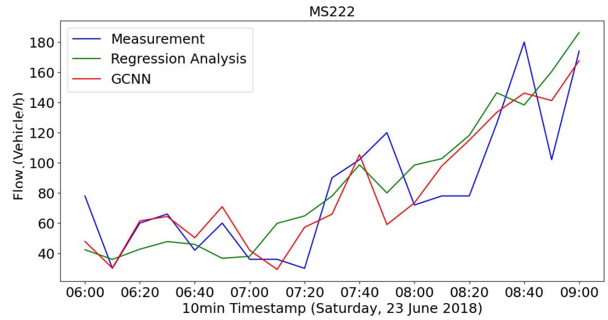

Fig. 7: Flow prediction in morning peak on a randomly chosen weekend-day (detector MS222).

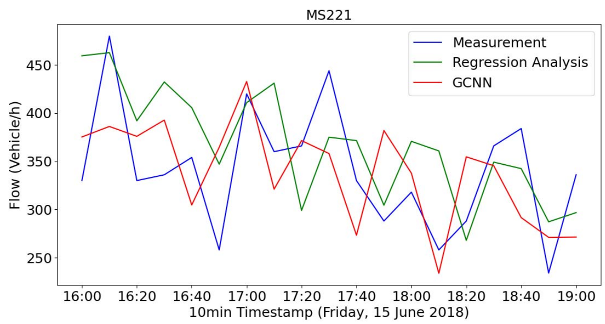

Fig. 8: Flow prediction in evening peak on a randomly chosen working-day (detector MS221).

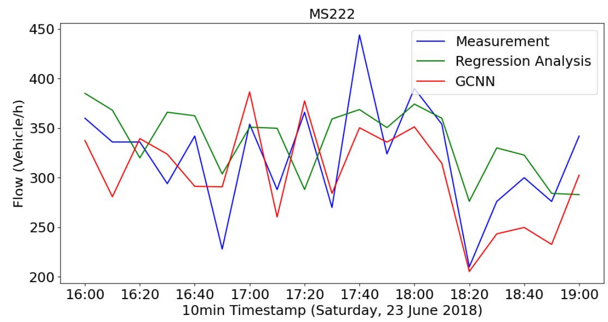

Fig. 9: Flow prediction in evening peak on a randomly chosen weekend-day (detector MS222).

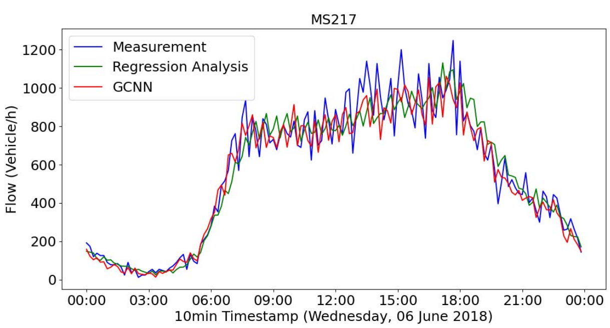

Fig. 10: Flow prediction on a randomly chosen working-day (detector MS217)

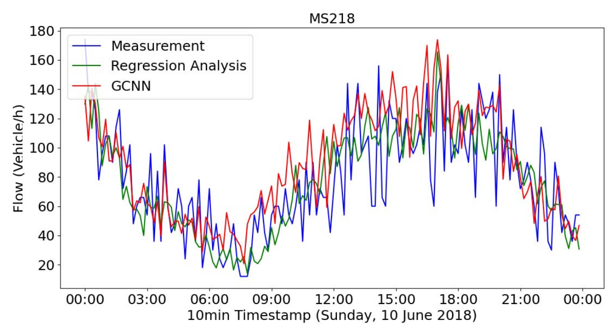

Fig. 11: Flow prediction on a randomly chosen weekend-day (detector MS218)

The discussion of the outcome of the Regression Analysis and the Graph Convolutional Neural Network (GCNN) models is carried out according to all-day hours and traffic peak hours in the morning and in the evening. We report the results of 3 weeks of testing from 4 June 2018 to 24 June 2018. The focus is on morning and evening peaks where a significant amount of vehicles is flowing. The morning peak is set to be between 06:00 and 09:00. In the evening, 3 hours of traffic peak is chosen between 16:00 and 19:00. The results are exhibited per detector as shown in Tables I, II and III.

Tables I and II recapitulate the results of the experimental tests per detector for the morning and evening peaks respectively. The Regression Analysis model is applied to each detector's data separately, thus, the learnt functions are in total 1225. However, GCNN takes into account all detectors data simultaneously. For GCNN we set a one-hour forecast horizon, whereas the Linear Regression model performs one-day prediction at a time. We can clearly see that the prediction accuracy differs from one detector to another. This mainly pertains to the flow volume where the detector is placed,

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:39:01 UTC from IEEE Xplore. Restrictions apply.

because some of them records from multiple-lane roads (roads do not have the same number of lanes). When the flow volume is high, the MAE tends to take higher values, and vice versa. This also is the reason why we have higher MAE during peak hours than all-day hours. Athwart, the MAPE, measuring a relative error, decreases during peak times. In average, both models produce $24\% - 25\%$ MAPE for all-day hours, which decreases to around 19% during morning and evening peaks. As mentioned above, since we are dealing with a signalized junction situated in downtown, the reached accuracy is satisfactory, especially during the most congested times of the day. It is noteworthy that our data differs from the one used in recent works, wherein data-sets are mostly from the Caltrans Performance Measure System (PeMS). The just-mentioned system records highway traffic data in California, which lacks the strong uncertainty about a vehicle's behavior (taken direction) at a given junction. Furthermore, apart from the noise created by traffic light signals, this kind of behaviour constantly changes and is very hard to predict.

For visualization purposes, Figures 10, 11, 6, 7, 8 and 9 illustrate the performance of both models on randomly chosen detectors on different days (weekend and working-days). Although the exhibited samples show that the models can accurately track the measured flow trend, they also display the weakness of the models, which is the inability of predicting occasional peaks (such as in Figure 6 at 07:40). Another data-related weakness, is the sharp fluctuations that appear from time to time (refer to Figure 11). As mentioned above, unfortunately, we don't have any data about these unusual traffic patterns.

VI. Conclusion and Future Directions

This article dealt with forecasting traffic flow volume at a single signalized traffic junction in Bremen (Germany). We focused on a very congested location, in which 7 loop detectors are installed. In order to forecast traffic flow in this region, we proposed two different models: Linear Regression and Graph Convolutional Neural Network based models. On a 3 weeks test-set, we measured the accuracy of both models deploying MAE and MAPE as performance indicators. The experimental results showed that the models are closely competitive and produce satisfactory forecasts with 19% error in average for the peak times during the day and with averagely 25% for all-day hours. To improve upon this, we plan to benefit from both models strength, and possibly, hybridize between them. Additionally, collecting traffic light data could help in predicting the fluctuating traffic patterns.

References

[1] Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in Neural Information Processing Systems*, 33, 2020.

[2] Pinlong Cai, Yunpeng Wang, Guangquan Lu, Peng Chen, Chuan Ding, and Jianping Sun. A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting. *Transportation Research Part C: Emerging Technologies*, 62:21–34, 2016.

[3] Manoel Castro-Neto, Young-Seon Jeong, Myong-Kee Jeong, and Lee D Han. Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert systems with applications*, 36(3):6164–6173, 2009.

[4] Jun Fu, Wei Zhou, and Zhibo Chen. Bayesian graph convolutional network for traffic prediction, 2021.

[5] Rui Fu, Zuo Zhang, and Li Li. Using lstm and gru neural network methods for traffic flow prediction. In *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 324–328, 2016.

[6] Liang Ge, Siyu Li, Yaqian Wang, Feng Chang, and Wu Kunyan. Global spatial-temporal graph convolutional network for urban traffic speed prediction. *Applied Sciences*, 10:1509, 02 2020.

[7] Donato Impedovo, Vincenzo Dentamaro, Giuseppe Pirlo, and Lucia Sarcinella. Trafficwave: Generative deep learning architecture for vehicular traffic flow prediction. *Applied Sciences*, 9(24):5504, 2019.

[8] Weiwei Jiang and Jiayun Luo. Graph neural network for traffic forecasting: A survey. *arXiv preprint arXiv:2101.11174*, 2021.

[9] S Vasantha Kumar and Lelitha Vanajakshi. Short-term traffic flow prediction using seasonal arima model with limited input data. *European Transport Research Review*, 7(3):1–9, 2015.

[10] Wan Li, Jingxing Wang, Rong Fan, Yiran Zhang, Qiangqiang Guo, Choudhury Siddique, and Xuegang Jeff Ban. Short-term traffic state prediction from latent structures: Accuracy vs. efficiency. *Transportation Research Part C: Emerging Technologies*, 111:72–90, 2020.

[11] Longshun Liu. A short-term traffic flow prediction method based on svr. In *2021 2nd International Conference on Urban Engineering and Management Science (ICUEMS)*, pages 1–4. IEEE, 2021.

[12] Yisheng Lv, Y. Duan, Wenwen Kang, Zhengxi Li, and F. Wang. Traffic flow prediction with big data: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16:865–873, 2015.

[13] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[14] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling, 2019.

[15] Hanyu Yang, Xutao Li, Wenhao Qiang, Yuhan Zhao, Wei Zhang, and Chang Tang. A network traffic forecasting method based on sa optimized arima–bp neural network. *Computer Networks*, 193:108102, 2021.

[16] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions, 2015.

[17] Lun Zhang, Qiuchen Liu, Wenchen Yang, Nai Wei, and Decun Dong. An improved k-nearest neighbor model for short-term traffic flow prediction. *Procedia-Social and Behavioral Sciences*, 96:653–662, 2013.

[18] Yanru Zhang, Yunlong Zhang, and Ali Haghani. A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model. *Transportation Research Part C: Emerging Technologies*, 43:65–78, 2014.

[19] Zuduo Zheng and Dongcai Su. Short-term traffic volume forecasting: A k-nearest neighbor approach enhanced by constrained linearly sewing principle component algorithm. *Transportation Research Part C: Emerging Technologies*, 43:143–157, 2014.

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:39:01 UTC from IEEE Xplore. Restrictions apply.