Grenze International Journal of Engineering and Technology, June Issue

Neural Network-based Traffic Situation Classification for Intelligent Transportation Systems

Kanchan[1], Vivek Srivastava[2] and Rakesh Ranjan[3]
[1-3]Department of Computer Science and Engineering, ABES Engineering College, Ghaziabad, Uttar Pradesh, India
Email: viveksrivastava@abes.ac.in

***Abstract*— The intelligent transportation systems (ITS) rely heavily on traffic state identification for congestion management, traffic control, and real-time decision-making. The increasing availability of traffic sensor data has enabled the use of data-driven learning frameworks to capture nonlinear traffic dynamics. This paper proposes a traffic classification system based on neural networks combining vehicle count and temporal features to classify traffic into low, normal, heavy, and high levels. The model incorporates feature engineering based on temporal attributes and is trained using a feedforward neural network with Adam optimization and cross-entropy loss. Experimental results demonstrate an accuracy of approximately 82%, with confusion matrix analysis confirming strong class discrimination. The framework is effective for real-time traffic monitoring and smart city applications.**

***Keywords*— Data-Driven Analytics, Deep Learning, Intelligent Transportation Systems, Smart Mobility, Time-Series Analysis, Traffic Prediction, Urban Traffic Flow.**

I. INTRODUCTION

Cities are getting more crowded, and with everyone driving, traffic is just getting worse.all that traffic, you really need a good way to watch it and guess what it's going to do. Modern Intelligent Transportation Systems (ITS). Traffic jams cause delays, use more fuel, increase pollution, and raise accident risks. So, predicting traffic flow accurately is key for traffic control and smart planning [1, 2]. When you think about traffic, it's really just a series of events over time.It's tough using old information to guess what's coming next. Let's say that the flow of traffic (i.e. the flow of vehicles) can be expressed in this way:

$$X = \{x1, x2, \dots, xT\}$$

where $xt \in \mathbb{R}^{+}$ at that time t. T the volume of traffic at time t. The function $f(\cdot)$ is our goal. $f(\cdot)$ is to learn, so that we can predict the traffic of the next time:

$$\hat{x}\ t+1 = f(xt, xt-1, \dots, xt-n)$$

where $x\hat{}t+1$ Predicted is the traffic volume and n is the size of the look-back window. Traffic that happens at the beginning of the road. Prediction models used statistical methods like the Autoregressive Integrated. Moving Average (ARIMA), which assumes linearity and stationarity in traffic dynamics. However, real-world traffic shows strong non-linearity, temporal dependency and abrupt shifts due to events or network interactions. This reduces the efficacy of such models [3].

Grenze ID: 01.GIJET.12.2.164_24

Deep learning technologies are extensively used to automatically discover complicated patterns from large-scale traffic data, addressing these issues. Recent studies Hybrid deep learning architectures that incorporate temporal modelling, attention, and convolution techniques improve traffic forecast accuracy dramatically. In [4], enhanced heuristic-based traffic flow labelling algorithms were launched in the US to anticipate congestion with greater accuracy. It caught multi-scale traffic patterns using atrous convolution and convolution. A hybrid attention network was implemented. This paper emphasised the necessity of accurate traffic scenario labelling is crucial for successful prediction in dynamic metropolitan areas. As autonomous and connected cars become more prevalent, integration of heterogeneous data streams has created new opportunities for prediction. In the study [5], analyzed multi-source vehicle data streams from the perspective of autonomous vehicle forensics to predict traffic behavior and avoid accidents. The report highlighted traffic prediction is becoming a crucial component of the intelligent vehicle ecosystem, encompassing sensing, communication, and decision-making, rather than a standalone endeavour. Looking at it from a network point of view, efficient vehicular communication helps make traffic predictions more accurate. In [6], smart transport systems, adaptive methods like choosing the right relay vehicle and tuning parameters help make data transmission in VANETs more reliable.Because of this, the traffic data used in prediction algorithms is better quality and more up-to-date. Recurrent Neural Networks (RNNs) are ideal for traffic time-series modelling because they can maintain past data. GRU, a streamlined and efficient RNN variation, is gaining popularity in traffic prediction due to its low computing complexity and high temporal learning capabilities. [7].
The GRU updates its hidden state using gating mechanisms defined as

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z), \; r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r),$$

$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h), \; h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t ,$$

where z_t and r_t denote the update and reset gates, respectively, and $\odot$ represents element-wise multiplication. Inspired by the success of GRU-based time-based modeling and recent advances in intelligent traffic analytics, this work uses hourly junction-level traffic data to create a robust model that accurately predicts traffic for shorter periods of time. By learning temporal dependencies and periodic patterns inherent in traffic flow data, the proposed approach aims to support real-time traffic management and smart city application s.

II. LITERATURE REVIEW

In past few years, the major focus in research related to Intelligent Transportation Systems has been on artificial intelligence-based traffic forecasting models that can effectively capture the spatial and temporal cor- relations inherent in road networks.Graph-based deep learning methods have emerged as a critical direction, with information fusion within graph neural networks (GNNs) showing a significant improvement in traffic forecasting accuracy by modeling interdependencies between intersections and interactions across the net- work [8]. To make temporal representation more effective, spatio-temporal graph convolutional networks with self-attention mechanisms have been proposed, which are able to dynamically load timed dependencies and increase the model's resilience in variable traffic conditions [9]. Alongside these efforts, comprehensive reviews have systematically analyzed statistical, Machine learning and deep learning techniques, showing that nonlin- ear traffic patterns are better modeled by deep learning structures. [10]. Uncertainty-aware traffic forecasting has also received special attention in recent times. Bayesian deep learning models integrating convolutional and recurrent networks exhibit better performance in interrupted and random traffic scenarios, as they pro- vide definitive predictions as well as probabilistic estimates [11]. In the same vein, hybrid frameworks, which combine convolutional, recurrent, and attention mechanisms, have been proposed so that spatial correlations and long-term dependencies can be used in conjunction, leading to a significant improvement in urban traffic forecasting functions [12]. Additionally, hybrid deep learning structures integrating Long short-term mem- ory (LSTM) networks and convolutional neural networks (CNNs) with gated recurrent units (GRUs) have shown effectiveness in reducing forecast error on various traffic datasets [13]. Concurrently, machine learn- ing models using multi-source sensor data and spatial interactions have also been explored to improve traffic flow forecasting in smart mobility environments [14]. More recently, multi-factor fusion strategies using graph convolutional networks combined with recurrent temporal modeling have been proposed to capture complex spatial–temporal dependencies more comprehensively [15]. Spatio–temporal graph convolutional networks have shown markedly better performance than sequence-based models only, as they directly (explicitly) encode the road network topology during the learning process [16]. Furthermore, hybrid deep learning frameworks that incorporate convolutional, recurrent, and attention-based components are consistently demonstrating strong ability to accurately model non-linear (nonlinear) traffic dynamics[17]. The preliminary and system- atic study [18] conducted by Smith and Demetsky shows that there is a significant decline in the accuracy of short - term traffic forecasting in congestion conditions, which prompted the need for adaptive and data-driven approaches. Van Lint et al. [19] proposed a neural network in the state space model, which explicitly took into account traffic dynamics and uncertainty. As a result, better robustness was achieved compared to traditional regression-based models. Their work established that neural networks are a viable option for real-time traffic prediction. With the rise of machine learning, Support Vector Regression (SVR) became a popular technology. Wu et al. [20] used the SVR for travel time forecasting and proved that it outperforms historical averages and parametric models, especially in changing traffic conditions. Traffic forecasting research underwent a dramatic change with the introduction of deep learning. By estimating traffic flow using deep neural networks, Polson and Sokolov [21] showed that deep structures outperform shallow models because they are better at capturing hierarchical temporal patterns. Subsequently, Recurrent Neural Networks (RNN) advanced this area by explic- itly modeling temporal dependencies. Duan et al. [22] used RNN for short-term traffic flow forecasting and reported its stability over different timeframes. Long Short-Term Memory (LSTM) networks gained notoriety for their ability to lessen the vanishing gradient issue. Hochreiter and Schmidhuber [23] LSTM was originally proposed, which was later widely adopted in traffic forecasting studies to capture long-range temporal depen- dencies. Graph-based models were presented with the aim of integrating spatial dependencies between road sections. A spatiotemporal graph convolutional network (STGCN) was proposed by Yu et al [24] . Where state-of-the- art performance was attained on common benchmark datasets and traffic was successfully mod- eled as a graph structure. Similarly, Li et al. [25] presented the Diffusion Convolutional Recurrent Neural Network (DCRNN), in which graph diffusion processes were combined with recurrent modeling, traffic makes it possible to effectively capture both spatial and temporal correlations in networks. Attention mechanisms have also been studied to improve the interpretability and performance of the model. Qin et al. [26] presented a dual-stage attention-based recurrent neural network, which enables the focus to be on relevant time phases and characteristics during the forecast.In recent times, transformer-based structures have also been studied for traffic forecasting. Wu et al. [27] demonstrated that attention-only models is capable of capturing long-term dependency, leading to large-scale traffic forecasting.

TABLE I: COMPARATIVE SUMMARY OF AUTHENTIC TRAFFIC PREDICTION STUDIES

Ref.	Authors (Year)	Data Type	Method Used	Key Contribution
[18]	Smith & Demetsky (1997)	Loop detector data	Statistical models	Early congestion- aware forecasting
[19]	Van Lint et al. (2005)	Highway traffic	State-space NN	Robust short-term prediction
[20]	Wu et al. (2004)	Travel time data	SVR	Improved generalization
[21]	Polson & Sokolov (2017)	Urban traffic	Deep neural networks	Hierarchical temporal learning
[22]	Duan et al. (2016)	Traffic flow	RNN	Temporal depen-dency modeling
[23]	Hochreiter & Schmidhuber (1997)	Sequential data	LSTM	Long-term memory handling
[24]	Yu et al. (2017)	Sensor networks	STGCN	Spatial-temporal graph learning
[25]	Li et al. (2018)	Road networks	DCRNN	Graph diffusion modeling
[26]	Qin et al. (2017)	Multivariate time series	Attention-RNN	Feature/time-step attention
[27]	Wu et al. (2020)	Large-scale traffic	Transformer	Long-range dependency capture

III. METHODOLOGY

This section explains the proposed traffic classification framework, including data preprocessing, feature engi - neering, neural network architecture, training strategy, and evaluation protocol. The objective is to classify traffic conditions into four discrete levels, namely low, normal, heavy, and high, using traffic volume and temporal features.

A. Dataset Description

This dataset is made up of 2,976 traffic observations collected at fixed time intervals. Each record contains the calculations of the different vehicle categories and their respective temporal attributes. The target variable Traffic Situation reflects the traffic level derived from the total vehicle number. The dataset has no missing values.

Assume that the dataset is represented as follows:

$$D = \{(xi, yi)\}N$$

where N = 2976, xi ∈ R7 is the feature vector, and yi ∈ {0, 1, 2, 3} denotes the encoded traffic class.

B. Feature Engineering

To capture temporal traffic patterns, a number of time-aware attributes are extracted from raw timestamp attributes. The final feature vector is defined as follows:

$$x = [x1, x2, \ldots, x7],$$

where x1: Total vehicle count, x2: The time of day (0–23), x3: Weekday (0–6), x4: Morning rush hour indicator, x5: Evening rush hour indicator, x6: Night-time indicator, x7: Weekend indicator.

Binary time indicators are defined as follows:

x4 = { 1, if 7 ≤ Hour ≤ 9
0, otherwise}
x5 = {1, if 16 ≤ Hour ≤ 19
0, otherwise}
x6 = {1, if Hour ≥ 22 or Hour < 4 0, otherwise}
x7 = {1, if Weekend 0, if Weekday}

C. Data Encoding and Normalization

Categorical traffic labels have been converted into numerical class indices through label encoding. Feature normalization is done by the standard scaling method, which can be expressed as follows:

$$x'_i = \left(\frac{x_j - \mu_j}{\sigma_j}\right)$$

Calculations from the training set j. j^{th} represent the feature's mean and standard deviation, respectively. We split the dataset into three parts: one for training, one for checking our work, and a final one for testing, following a specific ratio.We split 70% here, 15% there, and 15% for the final portion, employing the stratified sampling technique to guarantee a fair distribution of classes among all subgroups.

D. Neural Network Architecture

A feedforward neural network is employed for traffic classification, as the task involves instantaneous condition assessment rather than sequence prediction. The model architecture consists of: Input layer with 7 neurons, ReLU activation and 64 neurons make up the first buried layer, the dropout layer has a 0.3 rate, ReLU activation and 32 neurons make up the second buried layer, dropout layer with a rate of 0.2, the output layer has 4 neurons and uses the Softmax activation function.

Let h_1 and h_2 denote hidden layer activations:

$$h_1 = \mathrm{ReLU}(W_1x + b_1),\ h_2 = \mathrm{ReLU}(W_2h_1 + b_2),$$

$$y\hat{} = \mathrm{Softmax}(W_3h_2 + b_3),$$

where ˆy represents class probability estimates.

This neural network consists of a total of 2,724 trainable parameters

E. Model-based Training Method

The Adam optimizer is used to train the model, and the learning rate is set at 0.001. The categorical forecasting function is optimized through the definition of the Sparse Categorical Cross-Entropy loss function is:

$$\mathcal{L} = -\sum_{i=1}^{N} \log P(y_i|x_i)$$

Shows the probability estimated by the model. A batch size of 32 has been used for training up to a maximum of 100 epochs. The following callbacks have been used to improve generalization capacity and training stability:

- Early termination with 15 epochs of patience,
- A decrease in learning rate on the validation loss plateau,

- Model checkpointing based on validation accuracy.

F. Evaluation Metrics

Standard multi-class classification metrics are used to evaluate model performance on an unknown test set. Overall accuracy is computed as

$$Score_c = \left(\left(\frac{1}{N}\right) \sum_{\{i=1\}}^{\{N\}} \mathbb{I}(\hat{y}_i = y_i), \frac{|P_c \cap T_c|}{P_c}, \frac{|P_c \cap T_c|}{T_c}, \frac{2|P_c \cap T_c|}{|P_c| + |T_c|} \right)$$

The confusion matrix is used to analyze the behavior of the prediction according to each class, while con - fidence calibration and permutation-based feature importance are employed to assess model reliability and interpretability.

G. Deployment and Decision Logic

The trained model is packaged along with the feature scaler and label encoder for deployment. For real -time inference, the forecast for traffic conditions is

$$\hat{y} = \arg\max_k P(y = k \mid \mathbf{x}),$$

with associated confidence

$$C = \max_k P(y = k \mid \mathbf{x}).$$

A control decision rule is defined such that traffic control actions (e.g., gate opening) are triggered when the predicted class corresponds to heavy or high traffic conditions.

Algorithm 1 Traffic Level Classification

Require: Traffic dataset D Ensure: Predicted traffic class $\hat{y}$

- Load traffic dataset D
- Extract temporal features (Hour, Day of Week)
- Generate rush-hour, night, and weekend indicators
- Construct feature vector X
- Encode traffic labels and normalize features
- Split data into training, validation, and test sets
- Train feedforward neural network
- Select best model using validation accuracy
- Predict traffic class $\hat{y}$
- Trigger control action if $\hat{y}$ is heavy or high

H. Training Convergence Analysis

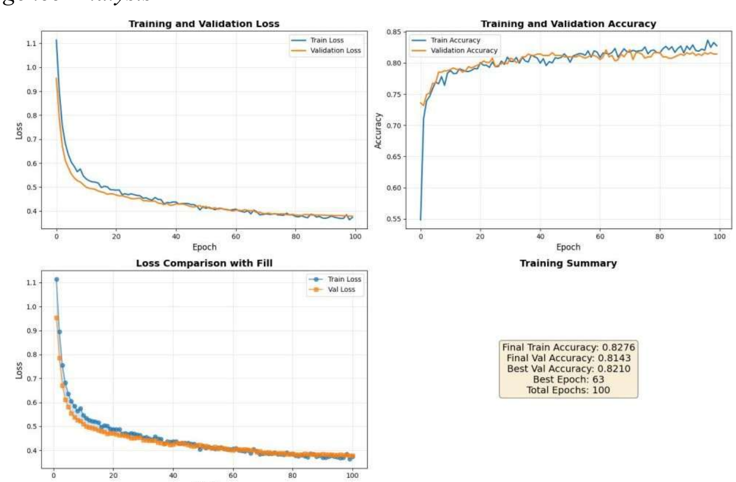

Fig. 1 Loss and accuracy curves for training and validation throughout epochs, curves across epochs, along with a summary of convergence statistics.

In order to examine the behavioural attributes of learning and convergence in the proposed model, epochs were used to track the accuracy and loss of the training and validation. Fig. 1 demonstrates how these metrics have changed throughout the training process. Looking at the loss curves, both the training and validation loss steadily"It goes down as the practice and checking losses go down." epochs, which represent the ongoing capacity for learning and optimization. The minimal difference between training and validation loss suggests that the model is not overfit. On the same note, the accuracy curves confirm the gradual improvement. The validation accuracy right behind the training accuracy is the training accuracy of good generalization of unseen data. The highest validation accuracy of about 82.1% is obtained in the region of the 63 rd few epochs after which the performance ceases to increase. On the whole, the convergence behavior proves that the chosen network architecture and optimization. strategy can be applied to classify traffic levels and train models with a high level of reliability.

IV. Findings and Conversation

To evaluate the model's ability to generalize, its performance in classifying traffic levels was tested using a dedicated dataset. The model achieved an overall classification accuracy of approximately 82%, indicating reliable identification among various traffic conditions. Class-wise analysis makes it clear that the model performs better in recognizing dominant traffic situations in the dataset. The low performance observed in the low traffic class can be explained by class imbalance and feature overlap during off-peak periods.
Overall, the results of the experiments prove that the proposed model is usable in intelligent transportation systems for real-time traffic monitoring and decision-making.

A. Analysis of Confusion Matrix

Confusion matrix analysis was performed on the test dataset to assess the classification performance of the proposed model. Fig. 2 shows the confusion matrix in the form of Absolute Counts and Normalized Percentages for each traffic class.
Since the majority of the data is properly recognized, the findings demonstrate how effectively the model works in both typical and high traffic situations. This is to be expected as these two traffic situations make up the majority of the information and their traffic patterns are clear and easy to understand.
The misclassification has been observed primarily between nearby traffic levels, especially between low and normal, and high and heavy traffic situations. These types of errors reflect gradual changes in real-world traffic flow, not sudden shifts, which is a common challenge in traffic classification.
The percentage-based confusion matrix makes it clear that all classes are dominated by diagonal elements, which proves the robustness of the proposed model. Overall, the confusion matrix analysis shows that the model performs reliably and consistently in various traffic conditions, making it suitable for actual use in intelligent transportation systems.

V. Conclusion and Upcoming Projects

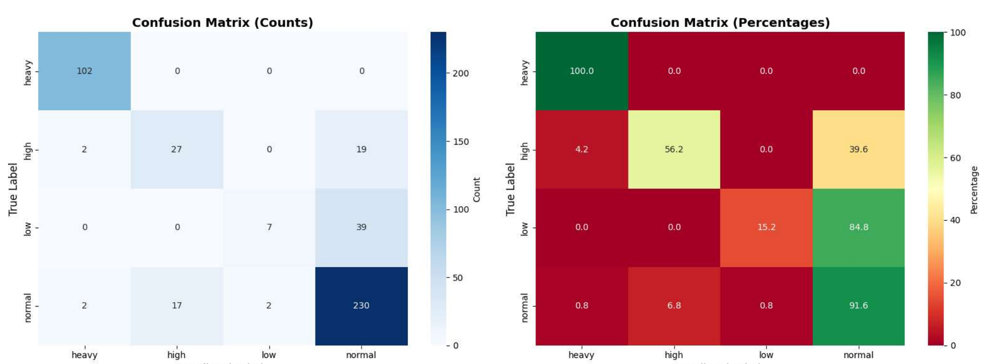

Fig. 2 Confusion matrix for traffic level classification: (left) absolute prediction counts and (right) normalized percentage-based representation.

The research proposes a neural network-based traffic level classification architecture that combines both vehicle number information and time-related features to identify a traffic situation as low, normal, heavy, or very high levels. The suggested model exhibited consistent training and attained consistent classification accuracy, and the total accuracy was about 82 percent on test data other than that used during the training process. The model's ability to explain the temporal fluctuations in traffic and the fluctuating vehicle density was demonstrated empirically. The confusion matrix and the class-wise analysis revealed good performance on the dominant conditions of traffic and the analysis on feature importance confirmed the importance of the traffic volume and time-specific features. These results point to the findings indicate that the suggested strategy is appropriate to real-time traffic. intelligent transportation systems monitoring and decision support. The next step in the development of work will be related to the further expansion of the proposed framework with the addition of spatial. data of traffic at junctions and combine sequential models to represent long-time temporal relationships. Also, the application of the model to the real-life traffic control scenarios and its efficiency during the active traffic conditions are the interesting possibilities of subsequent research.

REFERENCES

[1] Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C.: Short-term traffic forecasting: Where we are and where we're going. Transportation Research Part C: Emerging Technologies 43, 3–19 (2014)

[2] Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.-Y.: Traffic flow prediction with big data: A deep learning approach. IEEE Transactions on Intelligent Transportation Systems 16(2), 865–873 (2015)

[3] Billings, S.A.: Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio- Temporal Domains. John Wiley & Sons, ??? (2013)

[4] Srivastava, V., Mishra, S., Gupta, N.: Traffic flow labelling for congestion prediction with improved heuris- tic algorithm and atrous convolution-based hybrid attention networks. Mobile Networks and Applications (2024) https://doi.org/10.1007/s11036-024-02304-0

[5] Srivastava, V., Mishra, S., Gupta, N., Albalawi, E., Basheer, S.: Autonomous vehicle forensics: Inves- tigating data streams for traffic prediction and incident mitigation. IEEE Transactions on Consumer Electronics (2025)

[6] Kumar, V., Srivastava, V., Gupta, N., M ¨a kel ¨a , J.: Adaptive priority and optimum parameters based relay vehicle selection scheme in vehicular ad hoc networks. Computers and Electrical Engineering 126 (2025)

[7] Cho, K., Merri¨enboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder–decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1724–1734 (2014)

[8] Ahmed, S.F.: Enhancement of traffic forecasting through graph neural network information f usion. Transportation Research Part C (2024)

[9] Zhang, Y., Li, H.: Spatial–temporal self-attention graph convolutional network for traffic flow prediction. Applied Sciences (2024)

[10] Liu, X., Wang, J.: A comprehensive review of traffic flow forecasting based on deep learning paradigms. Journal of Traffic and Transportation Engineering (2025)

[11] Fu, F., Chen, Z.: Urban traffic volume prediction using bayesian deep learning. Sustainability (2024)

[12] Wang, Y., Zhao, K.: Traffic flow prediction via a hybrid cnn–lstm–attention model. Future Internet (2025)

[13] Singh, V., Sharma, A.: A novel cnn–gru–lstm based deep learning model for accurate traffic prediction. Journal of Intelligent Systems (2025)

[14] Abduljabbar, R., Dia, H.: Machine learning traffic flow prediction models for smart mobility. Smart Cities (2025)

[15] Chen, Y., Huang, L.: Spatial–temporal multi-factor fusion graph convolution network for traffic prediction. Scientific Reports (2025)

[16] Yu, B., Yin, H.: Spatio–temporal graph convolutional networks for traffic flow prediction. Transportation Research Part B (2025)

[17] Kumar, S., Patel, R.: Hybrid deep learning frameworks for traffic flow forecasting. IEEE Transactions on Intelligent Transportation Systems (2025)

[18] Smith, B.L., Demetsky, M.J.: Traffic flow forecasting: Comparison of modeling approaches. Journal of Transportation Engineering 123(4), 261–266 (1997)

[19] Van Lint, J.W.C., Hoogendoorn, S.P.: Accurate freeway travel time prediction with state-space neural networks. Transportation Research Part C 13, 379–401 (2005)

[20] Wu, C.H., Ho, J.M., Lee, D.T.: Travel time prediction using support vector regression. IEEE Transactions on Intelligent Transportation Systems 5(4), 276–281 (2004)

[21] Polson, N.G., Sokolov, V.O.: Deep learning for short-term traffic flow prediction. Transportation Research Part C 79, 1–17 (2017)

[22] Duan, Y., Lv, Y., Wang, F.Y.: Travel time prediction with recurrent neural networks. IEEE ITSC (2016)

[23] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 9(8), 1735–1780 (1997)

[24] Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks. IJCAI (2018)

[25] Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network. ICLR (2018)

[26] Qin, Y., Song, D., Chen, H.: A dual-stage attention-based recurrent neural network. IJCAI (2017)

[27] Wu, N.e.a.: Connecting the dots: Multivariate time series forecasting with attention. KDD (2020)