Traffic Flow Prediction in Urban Networks: Integrating Sequential Neural Network Architectures

Eva Lieskovska, Maros Jakubec, Pavol Kudela
University Science Park, University of Zilina, Zilina, Slovak Republic

***Abstract*—The rapid growth of urban areas has significantly compounded traffic challenges, amplifying concerns about congestion and the need for efficient traffic management. Accurate short-term traffic flow prediction remains important for strategic infrastructure planning within these expanding urban networks. This study explores a Transformer-based model designed for traffic flow prediction, conducting a comprehensive comparison with established models such as Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (BiLSTM), Bidirectional Gated Recurrent Unit (BiGRU), and Time-Delay Neural Network (TDNN). Our approach integrates traditional time series values with derived time-related features, enhancing the model's predictive capabilities. The aim is to effectively capture temporal dependencies within operational data. Despite the effectiveness of existing models, internal complexities persist due to diverse road conditions that influence traffic dynamics. The proposed Transformer model consistently demonstrates competitive performance and offers adaptability when learning from longer time spans. However, the simpler BiLSTM model proved to be the most effective when applied to the utilized data.**

Keywords—Traffic flow; short-term prediction; machine learning; transformer

I. Introduction

Urbanization and the subsequent surge in vehicular traffic pose challenges to the efficiency and sustainability of urban transportation networks. The intricate interplay of dynamic factors, including population growth, urban expansion, and evolving commuter behaviours, necessitates innovative solutions for managing traffic flow. In particular, the advent of advanced predictive models has emerged as a cornerstone in addressing the complexities inherent in urban traffic dynamics [1], [2].

Traffic flow prediction, an important component of intelligent transportation systems, facilitates proactive traffic management, congestion alleviation, and resource optimization. This predictive capability is increasingly crucial in urban planning and policymaking. Precise insights into future traffic patterns empower decision-makers to devise effective strategies for infrastructure development, traffic routing, and overall enhancement of urban mobility. Understanding population behaviours within transport models, especially in relation to mode choice for trips, forms a critical aspect that can influence the precision and application of predictive traffic models [3].

In the domain of traffic flow prediction, the quest for accurate, adaptive, and efficient models has intensified, given the important role of predictive systems in optimizing urban transportation networks. Traditional models, including Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Bidirectional Recurrent models and Convolutional Neural Networks (CNNs) such as Time-Delay Neural Network (TDNN), have significantly contributed to unravelling temporal dependencies within traffic data [4].

The introduction of the Transformer architecture [5], initially developed for Natural Language Processing (NLP) tasks, has paved the way for sequence modelling in various domains. Known for its distinctive attention mechanisms, this architecture revolutionises sequential data processing by employing self-attention mechanisms. This allows for a deeper comprehension of intricate relationships within sequences. The ability to discern temporal correlations has highlighted its potential application in traffic flow prediction models [6]–[8].

In this study, we undertake the task of forecasting future vehicle counts based on historical observations, with a specific focus on univariate traffic flow forecasting. The objective is to harness the capabilities of a transformer, which excels at discerning intricate traffic dynamics. The aim is to analyse how well it can decode complex traffic patterns by capturing time-related nuances and dependencies within the traffic data. Furthermore, we compare the performance of the transformer with the established neural networks for sequence modelling, such as LSTM, Bidirectional Long Short-Term Memory (BiLSTM), Bidirectional Gated Recurrent Unit (BiGRU), and TDNN. The incorporation of temporal features and the evaluation of distinct past observation intervals might yield additional insights for our analysis.

The paper is organized as follows: Section II provides an overview of existing traffic flow prediction models, Section III offers a detailed description of the prediction models, Section IV explains the experimental setup and evaluation methodologies, and Section V presents an analysis and comparative assessment of results. In Section VI, the discussion of the results is presented, and Section VII concludes with remarks that outline implications for future research.

II. Related Works

The evolution of time series prediction has been marked by advancements in data analysis, machine learning, and computational power. Initially, time series prediction relied on statistical (or parametric) methods such as Autoregressive (AR) and Moving Average (MA) models [9], [10]. These models are the building blocks of the Autoregressive Integrated Moving

Average (ARIMA) model [11], which remains a common approach for time series prediction to date. These methods assumed that future values depend linearly on past observations and aimed to capture the underlying trends and patterns.

Another frequently used method is employing Kalman filtering [12], [13]. Due to dynamic traffic conditions and the nonlinear nature of traffic flow, parametric methods may struggle to effectively capture traffic features. As a result, there has been a shift in focus towards non-parametric machine learning methods in the field of traffic flow forecasting [4]. Decision trees, k-nearest neighbour (k-NN) [14], Support Vector Machines, and Neural Networks (NNs) started making their way into the domain. However, challenges remained in handling the temporal dependencies inherent in time series data.

The resurgence of interest in neural networks, particularly Recurrent Neural Networks (RNNs), marked a significant milestone. RNNs, with their ability to capture sequential dependencies, demonstrated improved performance in time series prediction tasks. LSTMs are a type of RNN, which have shown the ability to extract complex correlations in non-linear traffic data and capture long-term dependencies. The study conducted in [15] compares the LSTM architecture with models such as random walk, support vector regression, wavelet neural network, and the stacked autoencoder, emphasizing its favourable outcomes in short-term traffic flow prediction. The hybrid LSTM proposed in [16] optimizes its structure and parameters to adapt to various traffic scenarios. Comparative analysis reveals that the hybrid LSTM model outperformed other typical models (Fuzzy C-Means, Kalman filter and LSTM) in terms of prediction accuracy. This improvement in accuracy was achieved with only a marginal increase in processing time compared to LSTM model.

The performance comparison between LSTM and its simplified counterpart GRU, indicates that GRU outperformed LSTM when the past observation sequences were small [17]. On the other hand, LSTM performed better with more complex datasets and required the use of extended sequences to predict future traffic volume. A comparative analysis with benchmark models proposed in [18], including ARIMA, LSTM, BiLSTM, and GRU, indicates the superior performance of the BiGRU model. The bidirectional model utilizes preceding and succeeding time sequences to extract additional traffic flow information. Notably, deep learning methods, including Bi-GRU, outperformed the traditional ARIMA model in prediction accuracy, particularly during peak periods. However, the BiGRU model exhibited a slight lag in traffic flow prediction.

Recent advances in time series forecasting using Transformer models are gaining traction in the field of traffic forecasting. Known for their prowess in cross-sequence tasks, these models have been refined to predict temporal data, fundamentally transforming conventional methodologies by optimizing computing processes and capturing extensive dependencies. Cai et al. [6] focused on addressing spatio-temporal dependencies in traffic forecasting. Their Traffic Transformer architecture, inspired by the Transformer framework and Graph CNNs, adeptly managed periodicity, and spatial dependencies. It showcased superior performance with real-world traffic datasets. Reza et al. [7] introduced a multi-head attention-based transformer model for traffic flow forecasting. The model demonstrates greater efficiency in capturing prolonged traffic flow patterns compared to recurrent-based models. However, to achieve optimal performance, the proposed transformer required substantial amounts of training data. Existing studies predominantly concentrate on short-term predictions, creating a gap in long-term traffic forecasting research. Tedjopurnomo et al. [8] stress the significance of extending prediction to 24 hours for better congestion planning. To overcome limitations in current recurrent structure-based models for long-term traffic prediction, they introduce a modified Transformer model named TrafFormer, incorporating time and day embedding. Experimental results highlight the superior performance of their proposed model compared to existing hybrid neural network models.

III. Methods

This section delineates the intricacies of sequential neural network architectures—LSTM, BiLSTM, BiGRU, TDNN, and Transformer—applied in the domain of traffic flow prediction models.

A. *Long Short-Term Memory and Bidirectional Long Short-Term Memory*

LSTM, an extension of a Vanilla RNNs, presents a robust architecture aimed at resolving the limitations of conventional RNNs in capturing long-range dependencies. Addressing the vanishing gradient problem inherent in RNNs, LSTM units incorporate a memory cell (c_t) that persists and evolves over time steps.

At each time step, LSTM units navigate through three gates: the forget gate (f_t), input gate (i_t), and output gate (o_t). These gates modulate the flow of information, orchestrating the update and retention of information within the cell state. The LSTM architecture is shown in Fig. 1.

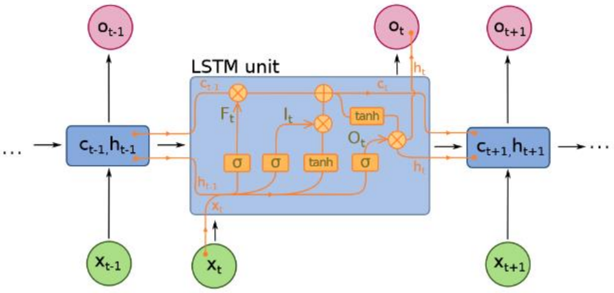

Fig. 1. Graphical visualization of the functioning of the LSTM unit.

Each gate within the LSTM unit serves a distinctive function:

- Forget gate (f_t): This gate regulates the relevance of past information, allowing the LSTM unit to decide the degree of retention or discarding of prior information from the cell state.

- Input Gate (i_t): Responsible for modulating incoming information, the input gate enables the selective update of the cell state based on the present input sequence and the preceding state.
- Output gate (o_t): Governing the flow of information from the cell state to generate the output, this gate ensures the controlled dissemination of relevant information.

In the context of traffic flow prediction models, LSTM networks exhibit remarkable proficiency in capturing and predicting complex traffic dynamics over prolonged periods, owing to their capacity to capture long-term dependencies within sequential traffic data.

BiLSTM extends the capabilities of LSTM by incorporating bidirectional processing, allowing information to flow both forward and backward within the network. BiLSTM units consist of two LSTM layers: one processes the input sequence forward in time, while the other processes the sequence in reverse. Each BiLSTM unit operates with two sets of gates similar to LSTM: forget gates $(\overrightarrow{f_t}, \overleftarrow{f_t})$, input gates $(\overrightarrow{i_t}, \overleftarrow{i_t})$, and output gates $(\overrightarrow{o_t}, \overleftarrow{o_t})$ for the forward and backward directions, respectively. This dual directionality enables the network to capture dependencies in both past and future contexts simultaneously.

By leveraging information from both past and future contexts, BiLSTM units excel in comprehensively understanding the sequential nature of data. In the domain of traffic flow prediction models, BiLSTM architectures demonstrate enhanced capabilities in capturing complex temporal dependencies, leveraging bidirectional information flow to predict traffic patterns with improved accuracy, especially when dealing with nuanced traffic dynamics influenced by historical and future context [19].

B. Gated Recurrent Unit and Bidirectional Gated Recurrent Unit

GRU presents an alternative architecture to LSTM, designed to capture long-range dependencies in sequential data. GRU units comprise two gates: reset gate (r_t) and an update gate (z_t), effectively regulating the flow of information within the network. The reset gate determines how much of the past information to forget, while the update gate modulates the blending of new input with the previous state. The GRU architecture is shown in Fig. 2.

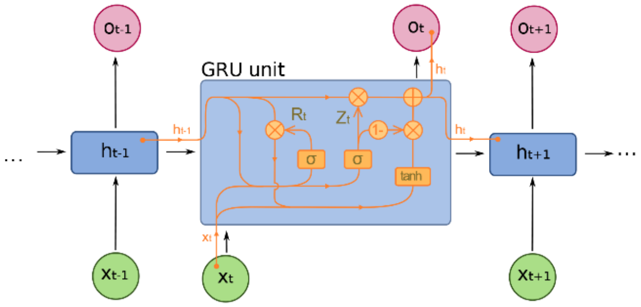

Fig. 2. Graphical visualization of the functioning of the GRU unit.

Unlike LSTM, GRU units do not possess a separate cell state, simplifying the architecture while preserving its capacity to capture temporal dependencies. GRU units are adept at learning from sequential data due to their simplified structure, making them particularly suitable for traffic flow prediction models. Their ability to balance the preservation and update of past information allows for effective modelling of traffic dynamics, enabling the prediction of flow patterns with a focus on essential temporal relationships.

BiGRU extends the GRU architecture to process information bidirectionally. Similar to BiLSTM, BiGRU incorporates two sets of GRU layers that process input sequences in both forward and backward directions. BiGRU units maintain the characteristics of GRU but leverage bidirectional information flow, allowing simultaneous exploration of past and future contexts [20].

In this work, bidirectional RNNs were employed to improve training efficiency by simultaneously processing the input sequence in both forward and backward directions (see Table I). The BiGRU and BiLSTM models comprise two bidirectional recurrent layers, and their outputs are aggregated using global average pooling. For comparison, we also included the classical LSTM model, which comprises three sequential LSTM layers, an aggregating LSTM layer, and densely connected layers.

TABLE I. CONFIGURATION OF RECURRENT MODELS

Layer	LSTM	BiLSTM	BiGRU
1.	LSTM()	Bidirectional(LSTM)	Bidirectional(GRU)
2.	Dropout(0.2)	Dropout(0.2)	Dropout(0.2)
3.	LSTM()	Bidirectional(LSTM)	Bidirectional(GRU)
4.	Dropout(0.2)	Dropout(0.2)	Dropout(0.2)
5.	LSTM()	GlobalAvgPooling()	GlobalAvgPooling()
6.	Dropout(0.2)	Dense()	Dense()
7.	LSTM()	Dense(1)	Dense(1)
8.	Dropout(0.2)		
9.	Dense()		
10.	Dense(1)		

C. Time-Delay Neural Network

TDNN represents a specialized class of feedforward neural networks designed for modelling temporal sequences. These networks utilize fixed-size time windows to capture intricate temporal dependencies embedded within sequential data. Unlike recurrent counterparts such as LSTM or GRU, TDNNs employ distinct convolutional layers, each capturing unique temporal abstractions within the input data.

Operating through convolutional layers that traverse the input sequence, TDNNs adeptly extract features within predefined time windows or delays. These localized features then undergo further processing across subsequent layers, culminating in higher-level representations that encapsulate the temporal intricacies within the data. By focusing on local patterns across diverse time scales, TDNNs excel in capturing short and medium-term dependencies inherent in sequential data. The complete architecture of the TDNN used in our experiments is detailed in Table II.

TABLE II. CONFIGURATION OF TDNN MODEL

Layer	TDNN
1.	TDNNLayer([-2,2])
2.	TDNNLayer([-2,0,2])
3.	TDNNLayer([-3,0,3])
4.	TDNNLayer([0])
5.	TDNNLayer([0])
6.	Flatten()
7.	Dense(32)
8.	Dense(1)

D. Transformer

Transformers have emerged as a paradigm-shifting architecture within neural networks, initially recognized for their success in NLP tasks. Unlike traditional RNNs, Transformers process input data in parallel, disassembling it into smaller tokens embedded within high-dimensional vectors. These vectors are then passed through multiple layers, utilizing a mechanism called self-attention to focus on important input segments. This intrinsic mechanism empowers Transformers to capture long-range dependencies and effectively model the underlying structures of natural language. The utilization of Transformers in traffic flow prediction represents a frontier where their prowess in capturing contextual relationships and long-range dependencies can significantly contribute to the evolution of precise traffic flow prediction models.

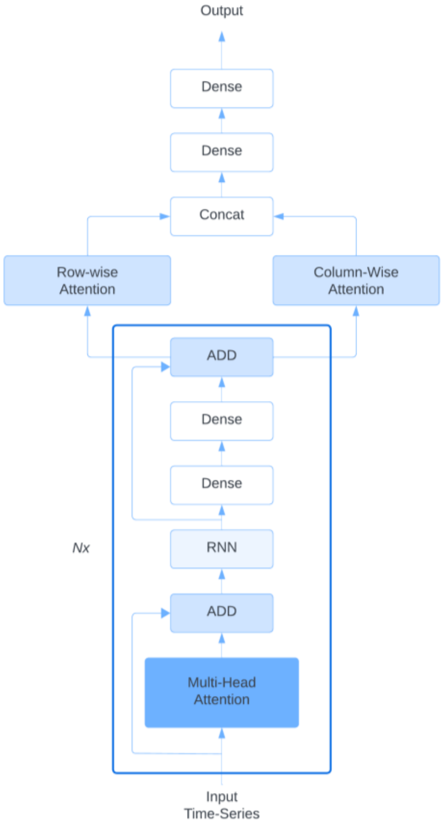

Fig. 3. Transformer architecture.

The Fig. 3 shows the proposed architecture of a single-block Transformer (where *Nx* represents the block ID), consisting of following sub-layers: a multi-head self-attention mechanism, an LSTM layer, and fully connected feed-forward network. In our implementation, we opted for the use of two transformer blocks based on experimental findings. The output of the last Transformer block is aggregated using row-wise and column-wise attention pooling and is then fed to the final dense layers. The model takes as input either the one-dimensional time series or two-dimensional time series × number of features.

IV. EXPERIMENTS

This section provides an overview of the experimental setup, dataset specifics, training strategies, and evaluation metrics crucial for both the development and assessment of the performance of the neural network architectures used in traffic flow prediction.

A. Dataset

The traffic dataset [21] used in this study is publicly available on the Kaggle online platform. This dataset consists of a collection of time series data, recording vehicle counts at hourly intervals across four distinct junctions. The features within this dataset include DateTime, Junction Type, Vehicle Count, and ID. The temporal span of data collection varies, encompassing observations from November 2015 to June 2017 for three junctions and from January 2017 to June 2017 for the remaining junction. Overall, this dataset comprises a total of 48,100 observations, providing insights into the hourly vehicular traffic across multiple junctions. In this study, data from junction number one was selected for experimentation.

Preprocessing techniques, including Z-score normalization and differencing with a one-week window span, were employed to mitigate inherent temporal patterns and trends within the dataset. Normalization addresses the issue of diversity in the value ranges of time series data, which is suboptimal for neural network input. The stationarity of the data was assessed using the Augmented Dickey-Fuller test.

In addition to time series values, we also included derived time-related features such as the month, hour, day of the week, weekend indicator, and lag features representing the values from the previous hour and the same hour on the previous day.

B. Network Setup and Training

The experiments were conducted on a hardware platform, encompassing the environmental parameters listed in Table III.

TABLE III. EXPERIMENTAL SETUP

Parameters	Configuration
CPU	Intel Core i9-12900HX
GPU	nVidia GeForce RTX 3080 Ti
GPU memory size	16GB
RAM	64GB
Operating systems	Win11
Deep learning architecture	Tensorflow 2.10.1

Training of the neural networks—LSTM, BiLSTM, BiGRU, TDNN, and Transformer—entailed parameter tuning. These models were systematically constructed with iterative exploration into diverse epochs, learning rates, batch sizes, and optimizer choices. Furthermore, the Halving Grid Search algorithm was used to narrow down the search for optimal settings through successive halving. The key parameters governing model training are detailed in Table IV.

TABLE IV. KEY PARAMETERS DURING MODEL TRAINING

Parameters	Setup
Epochs	500
Early stopping patience	10
Momentum	0.99
Learning rate	0.001
Weight decay	0.0005
Batch size	128
Optimizer	Adam/Lion

C. Metrics

The evaluation metrics are important in assessing the efficacy of traffic flow prediction models developed using neural networks. While analytical or theoretical validation of these models proves challenging, error metrics play a crucial role in assessing their performance [22].

The evaluation metric used in this work is the Mean Squared Error (MSE) and Mean Absolute Error (MAE). MSE is a common metric employed to measure the average squared difference between the actual and predicted values (1). A higher MSE indicates greater prediction error.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_{ref_i} - y_{pred_i})^2 \qquad (1)$$

where, n denotes the number of values. In this study, the Root Mean Square Error (RMSE) was utilized, which is the square root of MSE. This choice was made because RMSE shares the same scale as the original target variable.

The squaring of deviations in MSE significantly impacts the results, especially for extreme values. MSE exhibits higher sensitivity to these outliers. Conversely, for proximal values, squaring produces even smaller values, indicating their reduced significance rendering MSE less sensitive to nearby values. Therefore, an additional metric was employed to assess the performance of the models.

MAE operates similarly to MSE and represents the average positive deviation between predicted values and reference values. MAE is computed as the average absolute difference between predicted and reference values in Eq. (2) for n instances:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_{ref_i} - y_{pred_i}\right| \qquad (2)$$

MAE provides a single value encapsulating all absolute deviations. The MAE metric does not amplify the effect of outliers since it considers absolute differences without squaring.

V. EXPERIMENTAL RESULTS

In this section, we present and analyse the experimental results obtained from applying various time-series forecasting models. The outcomes of the model evaluation are detailed in Table V, encompassing results for five distinct models: LSTM, BiLSTM, BiGRU, TDNN, and Transformer. For each model, performance is assessed across two time intervals—6 hours and 12 hours. Past observations from the last t hours served as input, and predictions for the subsequent time point (t + 1 hour) were generated. Furthermore, two experimental settings were employed: time series modelling using simple sequences, and time series modelling with additional features. During the evaluation phase, we conducted 10 successive model trainings, and the results of the best model are reported.

The results reveal variations in the models' predictive capabilities under different forecasting horizons. In most cases, models that make predictions based on the past 6-hour time interval achieved better results. The Transformer model appears to perform well when forecasting based on longer time spans. The complexity of the proposed Transformer might handle intricate inputs more efficiently.

In general, the inclusion of time features in the learning process resulted in improved error metrics. By integrating temporal information, models acquire the capability to leverage inherent temporal patterns and dependencies within time series data. The best results for each time interval (columns) are highlighted in bold. Among the considered models, BiLSTM, BiGRU, and the proposed Transformer proved to be the most effective, with BiLSTM achieving the highest performance. This outcome can be attributed to the fact that a simpler model is more suitable for a smaller database. While the proposed Transformer model consistently demonstrates competitive performance, particularly evident with MAE values ranging from 0.1695 to 0.1714, it may be better suited for larger datasets. The utilization of pretraining could potentially further enhance its performance.

TABLE V. COMPARISON OF THE EXPERIMENTAL RESULTS

Model	**Metrics**	**Time series**		**Time series × features**	
		6h	12h	6h	12h
LSTM	RMSE MAE	0.2388 0.1720	0.2399 0.1725	0.2365 0.1694	0.2383 0.1703
BiLSTM	RMSE MAE	0.2380 0.1708	0.2392 0.1717	0.2350 0.1688	0.2363 0.1692
BiGRU	RMSE MAE	0.2361 0.1704	0.2398 0.1715	0.2359 0.1692	0.2368 0.1699
TDNN	RMSE MAE	0.2386 0.1720	0.2406 0.1735	0.2388 0.1724	0.2394 0.1727
Transformer	RMSE MAE	0.2385 0.1714	0.2367 0.1711	0.2363 0.1711	0.2376 0.1695

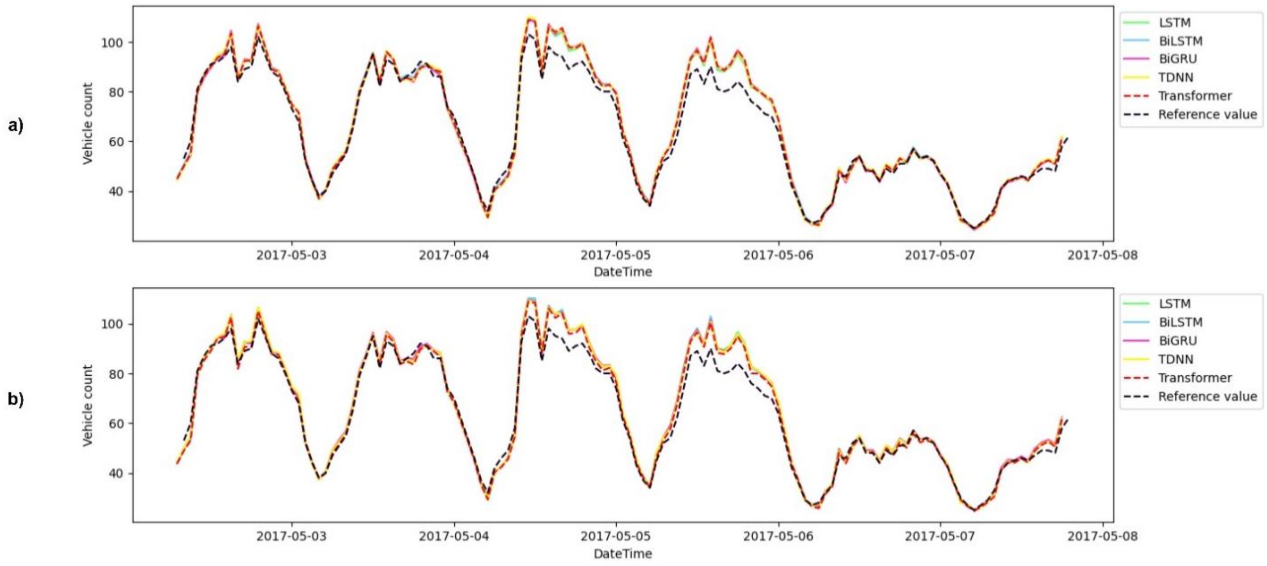

Fig. 4. Five-day prediction comparison across various sequence models: a) simple time series prediction; b) time series prediction with additional features.

VI. Discussion

The evaluation of LSTM, BiLSTM, BiGRU, TDNN, and a modified Transformer over two time intervals (6 hours and 12 hours) and across two experimental settings, including time series modelling with simple sequences and time series modelling with additional features, has provided insights into their predictive capabilities. The effectiveness of models is influenced by the choice of the past time horizon. Notably, most of the models learning from a 6-hour time span demonstrated superior performance compared to those learning from 12 hours. The inherent complexity of the Transformer architecture enables it to effectively capture temporal dependencies, making it particularly well-suited for forecasting based on longer time spans.

The predicted outcomes of all models without the use of time features are visualized in Fig. 4(a). Upon comparison with the addition of time features in Fig. 4(b), subtle improvements in prediction accuracy can be observed. The visualized days start from Tuesday and extend until midday on Sunday. The predicted values closely mimic the real-world values, with one notable exception: the models have learned to anticipate an increase in the number of vehicles on Thursdays and Fridays. Including supplementary information about holidays or non-working days might improve the model's decision-making process, especially in pinpointing the busiest traffic days of the week related to holiday travel.

The prediction outcomes for the proposed Transformer model and the best performing BiLSTM are illustrated in Fig. 5. For a more detailed perspective, only two days are displayed, revealing a distinct decline in the number of vehicles from Friday to Saturday. The incorporation of temporal features (see Fig. 5(b)) to some extent helped align the predicted values more closely with the actual values.

Selecting between BiLSTM and Transformer for time series prediction relies on the characteristics of the data and the available computational resources. While BiLSTM is a type of RNN that can capture temporal dependencies in sequential data, Transformer is a type of attention-based NN that can process sequential data in parallel, resulting in faster training times. The size of a dataset can influence the performance difference between a BiLSTM and a Transformer. The Transformer can be well-suited for transfer learning, particularly when pre-trained on large datasets, making it valuable for tasks involving limited labelled data.

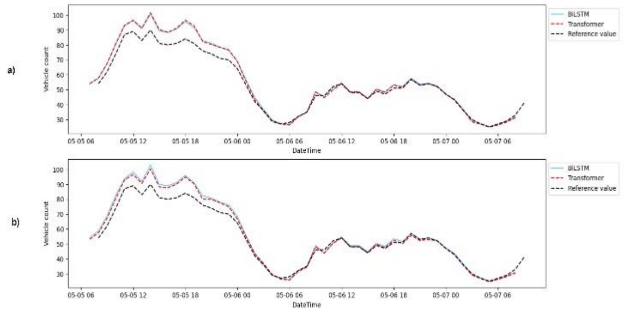

Fig. 5. The prediction outcomes of BiLSTM and Transformer models: a) simple time series prediction; b) time series prediction with additional features.

VII. Conclusion

In this study, we conducted an analysis of various time-series forecasting models, including LSTM, BiLSTM, BiGRU, TDNN, and modified Transformer. The evaluation encompassed two time intervals (6 hours and 12 hours) and two experimental settings: time series modelling using simple sequences and time series modelling with additional features. Our findings indicate variations in the predictive capabilities of the models under different forecasting horizons. Notably, models learning from a 6-hour time interval generally outperformed those learning from 12 hours. The Transformer model demonstrated efficacy in longer time spans, showcasing its ability to handle intricate inputs efficiently due to its inherent complexity.

The integration of time features into the learning process often resulted in improvements in error metrics. This enhancement arises from the models' capacity to leverage temporal patterns within time series data. Among the considered models, BiLSTM, BiGRU, and the proposed Transformer emerged as the most effective, with BiLSTM achieving the highest performance.

In our future work, the potential of transfer learning and improved fine-tuning will be explored. Moreover, evaluating other time series datasets may provide additional insights into the proposed analysis. The findings of this study can contribute to the broader understanding of model selection and optimization in time series forecasting, with implications for both research and practical applications in urban planning and traffic management systems.

References

[1] A. Boukerche and J. Wang, ‘Machine Learning-based traffic prediction models for Intelligent Transportation Systems’, Comput. Netw., vol. 181, p. 107530, Nov. 2020, doi: 10.1016/j.comnet.2020.107530.

[2] R. S. Joshi et al., ‘State-of-the-art reviews predictive modeling in adult spinal deformity: applications of advanced analytics’, Spine Deform., vol. 9, no. 5, pp. 1223–1239, Sep. 2021, doi: 10.1007/s43390-021-00360-0.

[3] M. Cingel, M. Drliciak, J. Celko, K. Zabovska, ‘Modal Split Analysis by Best-Worst Method And Multinominal Logit Model.’, presented at the Transport Problems: an International Scientific Journal . 2023, Vol. 18 Issue 1, p55-65. 11p.

[4] D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. M. Choudhury, and A. K. Qin, ‘A Survey on Modern Deep Neural Network for Traffic Prediction: Trends, Methods and Challenges’, IEEE Trans. Knowl. Data Eng., vol. 34, no. 4, pp. 1544–1561, Apr. 2022, doi: 10.1109/TKDE.2020.3001195.

[5] K. He, X. Zhang, S. Ren, and J. Sun, ‘Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition’, in Computer Vision – ECCV 2014, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014, pp. 346–361. doi: 10.1007/978-3-319-10578-9_23.

[6] L. Cai, K. Janowicz, G. Mai, B. Yan, and R. Zhu, ‘Traffic transformer: Capturing the continuity and periodicity of time series for traffic forecasting’, Trans. GIS, vol. 24, no. 3, pp. 736–755, 2020, doi: 10.1111/tgis.12644.

[7] S. Reza, M. C. Ferreira, J. J. M. Machado, and J. M. R. S. Tavares, ‘A multi-head attention-based transformer model for traffic flow forecasting with a comparative analysis to recurrent neural networks’, Expert Syst. Appl., vol. 202, p. 117275, Sep. 2022, doi: 10.1016/j.eswa.2022.117275.

[8] D. A. Tedjopurnomo, F. M. Choudhury, and A. K. Qin, ‘TrafFormer: A Transformer Model for Predicting Long-term Traffic’. arXiv, Mar. 02, 2023. doi: 10.48550/arXiv.2302.12388.

[9] S. Xu and B. Zeng, ‘Network Traffic Prediction Model Based on Auto-regressive Moving Average’, J. Netw., vol. 9, no. 3, pp. 653–659, Mar. 2014, doi: 10.4304/jnw.9.3.653-659.

[10] M.-C. Tan, S. C. Wong, J.-M. Xu, Z.-R. Guan, and P. Zhang, ‘An Aggregation Approach to Short-Term Traffic Flow Prediction’, IEEE Trans. Intell. Transp. Syst., vol. 10, no. 1, pp. 60–69, Mar. 2009, doi: 10.1109/TITS.2008.2011693.

[11] N. L. Nihan and K. O. Holmesland, ‘Use of the box and Jenkins time series technique in traffic forecasting’, Transportation, vol. 9, no. 2, pp. 125–143, Jun. 1980, doi: 10.1007/BF00167127.

[12] J. Guo, W. Huang, and B. M. Williams, ‘Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification’, Transp. Res. Part C Emerg. Technol., vol. 43, pp. 50–64, Jun. 2014, doi: 10.1016/j.trc.2014.02.006.

[13] S. V. Kumar, ‘Traffic Flow Prediction using Kalman Filtering Technique’, Procedia Eng., vol. 187, pp. 582–587, Jan. 2017, doi: 10.1016/j.proeng.2017.04.417.

[14] Y. Chen, Y. Zhang, and J. Hu, ‘Multi-Dimensional traffic flow time series analysis with self-organizing maps’, Tsinghua Sci. Technol., vol. 13, no. 2, pp. 220–228, 2008, doi: 10.1016/S1007-0214(08)70036-1.

[15] H. Shao and B.-H. Soong, ‘Traffic flow prediction with Long Short-Term Memory Networks (LSTMs)’, in 2016 IEEE Region 10 Conference (TENCON), Nov. 2016, pp. 2986–2989. doi: 10.1109/TENCON.2016.7848593.

[16] Y. Xiao and Y. Yin, ‘Hybrid LSTM Neural Network for Short-Term Traffic Flow Prediction’, Information, vol. 10, no. 3, Art. no. 3, Mar. 2019, doi: 10.3390/info10030105.

[17] L. C. Das, ‘Traffic Volume Prediction using Memory-Based Recurrent Neural Networks: A comparative analysis of LSTM and GRU’. arXiv, Mar. 22, 2023. doi: 10.48550/arXiv.2303.12643.

[18] S. Wang, C. Shao, J. Zhang, Y. Zheng, and M. Meng, ‘Traffic flow prediction using bi-directional gated recurrent unit method’, Urban Inform., vol. 1, no. 1, p. 16, Dec. 2022, doi: 10.1007/s44212-022-00015-z.

[19] R. L. Abduljabbar, H. Dia, and P.-W. Tsai, ‘Unidirectional and Bidirectional LSTM Models for Short-Term Traffic Prediction’, J. Adv. Transp., vol. 2021, p. e5589075, Mar. 2021, doi: 10.1155/2021/5589075.

[20] C. Chai et al., ‘A Multifeature Fusion Short-Term Traffic Flow Prediction Model Based on Deep Learnings’, J. Adv. Transp., vol. 2022, p. e1702766, May 2022, doi: 10.1155/2022/1702766.

[21] Fedesoriano, Traffic Prediction Dataset, February 2021.Retrieved from https://www.kaggle.com/datasets/fedesoriano/traffic-prediction-dataset.

[22] N. A. M. Razali, N. Shamsaimon, K. K. Ishak, S. Ramli, M. F. M. Amran, and S. Sukardi, ‘Gap, techniques and evaluation: traffic flow prediction using machine learning and deep learning’, J. Big Data, vol. 8, no. 1, p. 152, Dec. 2021, doi: 10.1186/s40537-021-00542-7.