Short Term Traffic Flow Forecast Based on CM-GRU Networks*

1st Zhongzheng Guo
Institute of Automation,
Harbin University of Science And Technology.
Harbin, China
the State Key Laboratory
for Management and Control
of Complex Systems,
Institute of Automation,
Chinese Academy of Sciences.
Beijing, China
guozhongzheng2017@ia.ac.cn

2nd Weifeng Zhong
Institute of Automation,
Harbin University of Science And Technology.
Harbin, China
zwfhlg@163.com

3rd Fenghua Zhu
the State Key Laboratory
for Management and Control
of Complex Systems,
Institute of Automation,
Chinese Academy of Sciences.
Beijing, China
fenghua.zhu@ia.ac.cn

4th Xiaoshuang Li
the State Key Laboratory
for Management and Control
of Complex Systems,
Institute of Automation,
Chinese Academy of Sciences.
Beijing, China
lixiaoshuang2017@ia.ac.cn

5th Fei-yue Wang
the State Key Laboratory
for Management and Control
of Complex Systems,
Institute of Automation,
Chinese Academy of Sciences.
Beijing, China
feiyue.wang@ia.ac.cn

6th Gang Xiong
the State Key Laboratory
for Management and Control
of Complex Systems,
Institute of Automation,
Chinese Academy of Sciences.
Beijing, China
gang.xiong@ia.ac.cn

***Abstract*—Intelligent transportation systems (ITS) have developed for a long time. The rise of deep learning has brought new vitality of the ITS. However, traffic flow data is usually time-correlated and highly randomized. The data distribution will also change dynamically. To actualize the forecasting of traffic flow accurately, we use the historical traffic information to predict the messages of the traffic flow at any time interval. This paper proposes an efficient traffic flow forecast architecture based on deep learning. The method combines the gated recurrent unit (GRU, a type of recurrent neural network) layers and one-dimension convolution layers. Since the performance of these models has a strong dependence on hyper-parameters, this paper conducts a large-scale search of the hyper-parameter space. At the same time, experiments on flow data show that the method proposed in this paper can achieve a better prediction accuracy. Experiments also show lower test errors compared with the existing approaches.**

***Index Terms*—Deep learning, Neural networks, traffic flow prediction, Intelligent Transportation Systems.**

I. INTRODUCTION

The ACP methods [6] consist of artificial systems, computational experiments and parallel execution and an extension named cyber physical social systems (CPSS [7]) are one of possible solutions to modeling the complicated environment and extract knowledge from the systems. The theory are widely used in many other fields [8]–[10].

In the traffic fileds, artificial societies [1] are critical for intelligent transportation systems (ITS) [2]–[4] because the systems usually have chaotic and complexity characteristics [5]. The construction process of artificial societies is to restore the original connection. Moreover, the connection is represented by a nonlinear mapping approximately. The accurately future state of the traffic flow is the first step to build the systems. With the information, the reasons for route choice behavior [11] will be clearer to understand. The traffic flow information also can be associated with driver postures for in-vehicle driving activities [12].

Real-time traffic flow forecasts [13] are also the premise of achieving dynamic traffic guidance [5]. Also, we mainly concentrate on short-term prediction within 30 minutes.

Artificial neural network (ANN) [14] models are one of the nonlinear mappings to fit the historical data. Most of ANN models are usually self-evolving through large scale data-driven learning methods. However, a specific set of hyper-parameters determined models can only predict a particular pattern of data. And its stability of prognostication will vary with the parameter initialization method and optimization algorithm.

This paper proposes a model called CM-GRUs. The model

*This work was supported by National Key R&D Program of China 2018YFB1700000, National Natural Science Foundation of China U1811463, 61533019, 61773381 and 91720000, the Beijing Municipal Science and Technology Commission Program under Grant D171100000317002. **Fenghua Zhu is the corresponding author**.

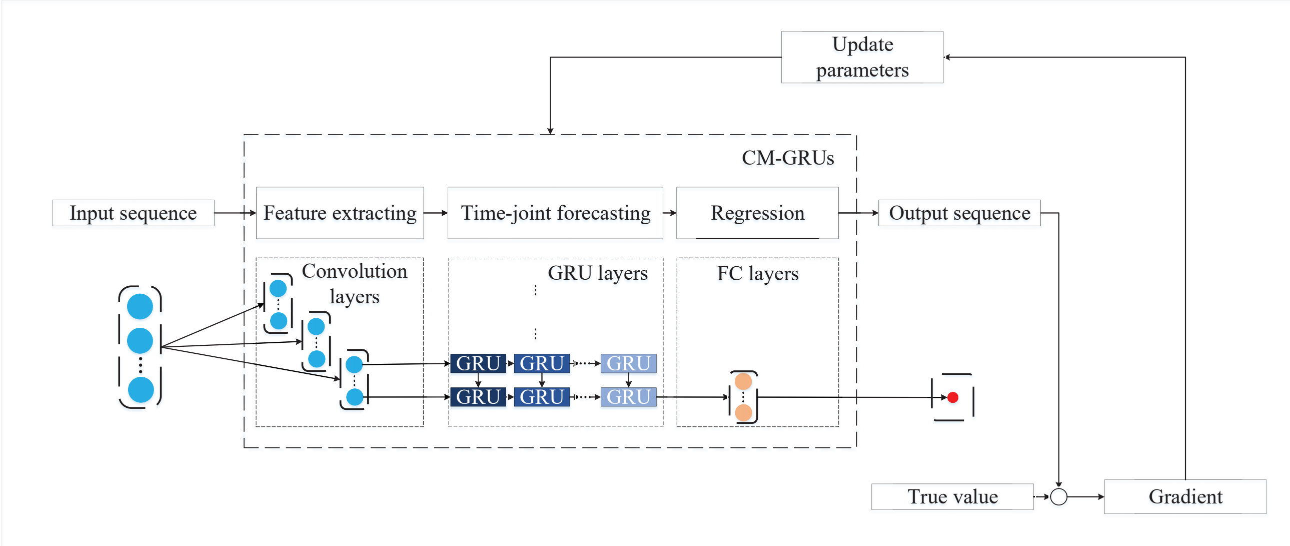

Fig. 1. The CM-GRU networks proposed in this paper.

represents the relation between the history state and future information of the traffic flow sequence. The traffic flow data contains temporal characteristic, and we use convolution layers to extract the temporal feature of traffic flow. To process the time-related data, we use GRU layers to forecast the future information with previous time sector. Our main contributions are as follows:

- We propose a novel traffic flow prediction model called CM-GRU. The model combines one-dimension convolution layers and GRU layers. The module extracts the temporal feature of traffic data. They can fully fuse the temporal characteristics of traffic flow samples.
- We use the temporal feature of traffic flow extracted by one-dimension convolution layers to predict short-term traffic flow. And the end-to-end deep learning architecture we proposed predicts short-term traffic flow accurately without manually extract features.
- We balance the performance and complexity through a large-scale structure search on real-world data. The results demonstrate that our method performed better than the other existing methods.

The rest of this paper is as follows: Section II focuses on some existing data-driven methods with respect to traffic flow prediction. Section III introduces the details of the method proposed in this paper. Section IV concentrates on the experiments and data analysis. Section V summarizes the full paper and draws conclusions.

II. BACKGROUND

In this section, the development of the traffic flow forecasting fields in recent years will be introduced.

The earlier works are based on statistical aspects, such as Autoregressive Integrated Moving Average Model (ARIMA) [15]–[17]. However, traffic flow data is highly stochastic and non-linear. The ARIMA algorithm cannot fit the data with multiple patterns because it is based on the linear relationship.

Furthermore, some machine learning methods, such as K-NN [18], [19], SVM [20], [21], and so on, have also been proposed. Zhang et al. [18] present a k-nearest neighbor (KNN) model to predict the short-term traffic flow data in an urban area. However, the algorithm does not analyze the feature of the data. Castro et al. [22] proposed a support vector regression (SVR) algorithm to predict traffic flow. They explained the non-linear traffic flow feature of a certain area.

Deep learning technique has attracted much attention recently. Many deep learning methods have been proposed for traffic flow forecasts, such as DBN [23], SAE [2], [24], RNN models [19], [25], [26] like LSTM [25]–[27] and GRU [28]. Koesdwiady et al. [29] proposed deep belief network (DBN) algorithm to predict traffic flow with weather information. Lv et al. [30] were applied to short-term traffic flow prediction with stacked auto-encoder (SAE). Their paper is the first time that the SAE approach is used to represent traffic flow features for prediction. Chen et al. [31] proposed a long-short term memory (LSTM) based method for traffic flow data prediction. They use LSTM to get the time trend of traffic flow data. The analysis of the data is not comprehensive, which leads to an inaccurate traffic flow prediction.

Dai et al. propose DeepTrend 2.0 [32] to forecast the detrended data with graph neural networks. Lin et al. [33] propose a predictive model based on generative adversarial networks [34]. This model can learn diverse representations for better prediction.

Much great progress has been made by the previous researchers, and we try to develop a more complex and deeper structure for predicting the data with better performance.

III. METHODOLOGY

In this section, our forecast methods of traffic flow data are introduced.

A. GRU blocks

In our works, gated recurrent unit (GRU) [35] is used to construct the forecast model preliminarily. GRU is a variance of recurrent neural networks with a gating mechanism.

We assume each sample x_t is an input of a GRU block, which mainly consists of update gate and reset gate [36]. And the update gate [36] z_t for time step t is expressed as Eq. (1).

$$z_t = \sigma_g(W_z x_t + U_z h_{t-1} + b_z) \tag{1}$$

In addition, the calculation of the reset gate is expressed as Eq. (2).

$$r_t = \sigma_g(W_r x_t + U_r h_{t-1} + b_r) \tag{2}$$

After the two gates are calculated, we introduce a new memory content expressed as Eq. (3).

$$h_t^{'} = tanh(W x_t + r_t \odot U h_{t-1}) \tag{3}$$

Finally, the output of GRU can be expressed as Eq. (4).

$$h_t = (1 - z_t) \odot h_t^{'} + z_t \odot h_{t-1} \tag{4}$$

In summary, the GRU layers can be expressed as Eq. (5) with many GRU blocks.

$$h^i = GRU(h^i) \tag{5}$$

B. CM-GRU network

Traditional GRU blocks forecasting sequence directly. In this paper, we proposed convolution multi-layer GRU (CM-GRU) networks to forecast the future information of sequence.

We assume the input sequence of the model is as Eq. (6).

$$X_{in} = \{x_{in}^1, x_{in}^2, ..., x_{in}^t, ..., x_{in}^m\} \tag{6}$$

Firstly, we use a convolutional operation [14] to extract the features f of the input sequence. This process is shown in Eq. (7).

$$f = X_{in} * K \tag{7}$$

where K is a window of convolution called kernel. K^T is the transpose of kernel. l is the length of kernel. And $*$ is convolution operation.

Furthermore, the i-th feature can be expressed as Eq. (8).

$$f_i = \sum X_{in}[i : (i + l)] K^T \tag{8}$$

Each feature will be pooled by a pooling window to saving the calculate costs.

After the features are extracted, we use several GRU layers to associate the features extracted from the input sequence. The formula of forward propagation is expressed as Eq. (9).

$$\begin{aligned} h^1 &= GRU^1(feture^i) \\ h^2 &= GRU^2(h^1) \\ &...... \\ \hat{f}_i &= GRU^j(h^j) \end{aligned} \tag{9}$$

Where j is the layers number.

Finally, a fully-connected (FC) layer will be placed to regress the feature to a single value.

$$output = \sum_{min}^{max} (W_d \ \hat{f} + b_d) \tag{10}$$

In this case, the whole process of forward propagation of CM-GRU network can be expressed as Eq. (11).

$$output = CMGRU_{params}(X_{in}) \tag{11}$$

where $params$ is the set of trainable parameters.

After the forward propagation, the parameters $params$ will be updated by a gradient-based method called backpropagation. Eq. (12) shows the procedure.

$$params_e = params_{e-1} + \alpha \ \nabla CMGRU_{params} \tag{12}$$

Where e is training epochs. α is the learning rate. $\nabla CMGRU_{params}$ is the gradient of the parameters.

In summary, the whole model structure of CM-GRU proposed in this paper is shown in Fig. 1.

IV. EXPERIMENTS

A series of experiments is designed as follows to verify the methods.

A. Data description

Caltrans Performance Measurement System (PeMS) is a website that provides information on the traffic flow of the California highway in the United States. All of the data provided on the website are collected through a vehicle detector system (VDS). By capturing the traffic flow data provided by the PeMS, this paper will establish a dataset depend on actual road sections.

The data set is from August 1st, 2016 to September 30th within two months. the data is collected per 5 min (including 35034 samples). Each sample contains an input sequence consists of the historical vehicle numbers of the traffic flow and a vehicle numbers in the future time stamps. The sequence length in each sample is a hyper-parameter need to be adjusted.

The data are divided into 3 parts consist of a training set, a validation set, and a test set. The training set is utilized to fitting a suitable model that can reflect the historical information to the future state. The validation set is used to avoid over-fitting. The test set is to test the model and provide persuasive evidence in the efficiency of the model. In this paper, 60% of the samples are used as a training set, and the remaining 40% is divided evenly as the validation set and test set. The absence rate of missing data is 0.02%, so the missing data will be filled with 0 merely.

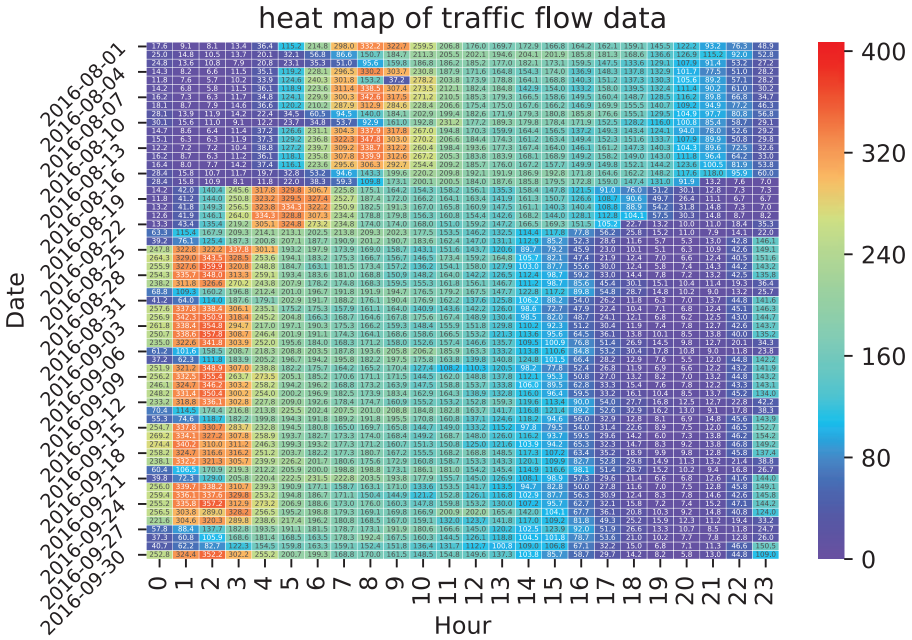

Fig. 2. The heat map of the data used in our paper. Features of different date probably have significant diversity according to the figure.

B. Performance metrics

In order to evaluate the performance of the proposed methods, we use three performance metrics to evaluate the pre diction results [37]. The mean absolute percentage error (MAPE), root mean square error (RMSE) to evaluate the performance can be expressed as Eq. (13), Eq. (14) and Eq. (15).

$$RMSE = \sqrt{\frac{\sum_{n=1}^{T}(True_t - Prediction_t)^2}{N}} \tag{13}$$

$$MAPE = \frac{\sum_{n=1}^{T}(True_t - Prediction_t)}{Prediction_t} \tag{14}$$

$$R^2 = 1 - \frac{\sum_{n=1}^{T}(Prediction_t - True_t)^2}{\sum_{n=1}^{T}(True_t - \bar{True_t})^2} \tag{15}$$

C. Model architecture

We use an automated machine learning framework called Hyperopt [38] to conduct a large number search of the hyper-parameters. The search space can be expressed in Tab. I.

The CM-GRU model we proposed with optimal structure is shown in Tab. II. In addition, we found that the 'Adamax' optimizer is fast to train parameters. So we select the 'Adamax' optimizer to reduce the time costs of the training process.

Fig. 3. The satellite map of the data used in our paper. We use a station in Junipero Serra Fwy, San Jose, California to test the method we proposed.

D. Performance comparison of different methods

Comparative experiments are designed in this paper by using a neural network and traditional machine learning methods for prediction, including LSTM, GRU, LR, SVR, etc.

The comparison results are shown in Table III.According to Table. III, the performance of CM-GRU proposed in the paper is the best among the three stations. The LSTM model is comparable to the GRU, but its complex structure leads to inefficiency. And the performance of traditional machine learning model is far worse than the neural network, which ultimately affects the performance of forecastability in these models.

Fig. 4 shows the performance comparison of different

TABLE I
THE SEARCH SPACE OF THE PROPOSED MODEL.

The type of hyper-parameters	search space
length of the input sequence	1,2,3,...,25
nodes number of the 1-D convolution-1st	8,16,24,...,48
nodes number of the 1-D convolution-2nd	8,16,24,...,48
nodes number of the GRU-1st	5,10,15,...30
nodes number of the GRU-2nd	5,10,15,...30
nodes number of the GRU-3rd	5,10,15,...30
optimizer	'SGD'; 'Adam'; 'Adamax'

TABLE II
MODEL ARCHITECTURE OF THE PROPOSED MODEL. THE '?' REFERS TO THE BATCH SIZE OF THE SAMPLES, AND IT WILL BE DETERMINED ONLY WHEN THE MODEL IS TRAINING.

Layer type	Output shape	Number of parameters
Input layer	(?, 19, 1)	0
1-D convolution-1st	(?, 17, 32)	128
1-D convolution-2nd	(?, 15, 16)	1552
Max pooling	(?, 5, 16)	0
GRU-1st	(?, 5, 10)	810
GRU-2nd	(?, 5, 10)	630
GRU-3rd	(?, 5, 10)	630
GRU-4th	(?, 10)	630
Dense	(?, 1)	11

models for part of the test data. CM-GRU module is the best of all models compared with SVR, LSTM and GRU.

TABLE III
PERFORMANCE OF ALL MODELS

Models	$RMSE$	$MAPE$ (%)	R^2 (%)
SVR	28.86	27.2	96.7
LSTM	18.43	14.3	97.3
GRU	17.21	13.3	98.1
CM-GRU (Ours)	**10.83**	**5.1**	**99.6**

V. CONCLUSION AND DISCUSSION

In this paper, we proposed a feasible model called CM-GRU Networks. We evaluated the prediction ability of the corresponding hyperparameters under different settings by using RMSE, MAE, and MRE, then compared their abilities of the three types of LSTMs. And we found that deep learning methods are very suitable to do the traffic flow forecast between different datasets and in complex conditions. From the respect of test errors, CM-GRU model outperforms to other methods. $RMSE$, $MAPE$, and R^2 can reach a minimum of 10.83, 5.06%, 99.57%. The mean RMSE of the proposed CM-GRU is 37% lower than the GRU model. In summary, combining convolution layers with GRU layers will achieve better results, allowing the model to be robust and efficient training.

At the same time, works in this paper have certain deficiencies. In the future, larger datasets will be trained by us and a more comprehensive and detailed tuning of hyperparameters is planned to make the model more stable and the results more credible. Using asynchronous architecture will accelerate the training procedure. In addition, according to the traffic conditions collected from the web, we can do the forecast more efficient and duly.

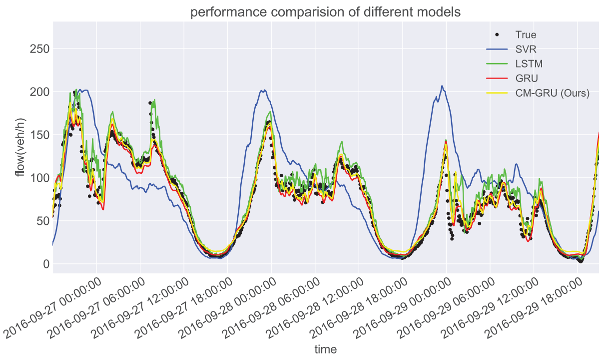

Fig. 4. Forecast performance of different models.

REFERENCES

[1] P. Ye, S. Wang, and F.-Y. Wang, "A general cognitive architecture for agent-based modeling in artificial societies," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 1, pp. 176–185, 2017.

[2] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.

[3] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624–1639, 2011.

[4] G. Dimitrakopoulos and P. Demestichas, "Intelligent transportation systems," *IEEE Vehicular Technology Magazine*, vol. 5, no. 1, pp. 77–84, March 2010.

[5] D. Chen, "Research on traffic flow prediction in the big data environment based on the improved rbf neural network," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2000–2008, 2017.

[6] S. Wang, X. Wang, P. Ye, Y. Yuan, S. Liu, and F.-Y. Wang, "Parallel crime scene analysis based on acp approach," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 1, pp. 244–255, 2018.

[7] F.-Y. Wang, Y. Yuan, C. Rong, and J. J. Zhang, "Parallel blockchain: An architecture for cpss-based smart societies," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 2, pp. 303–310, 2018.

[8] Y. Yuan, F.-Y. Wang, and D. Zeng, "Competitive analysis of bidding behavior on sponsored search advertising markets," *IEEE Transactions on Computational Social Systems*, vol. 4, no. 3, pp. 179–190, 2017.

[9] F.-Y. Wang, "Computational social systems in a new period: A fast transition into the third axial age," *IEEE Transactions on Computational Social Systems*, vol. 4, no. 3, pp. 52–53, 2017.

[10] J. Li, X. Ni, and Y. Yuan, "The reserve price of ad impressions in multi-channel real-time bidding markets," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 2, pp. 583–592, 2018.

[11] L. Li, S. Wang, and F.-Y. Wang, "An analysis of taxi drivers route choice behavior using the trace records," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 2, pp. 576–582, 2018.

[12] Y. Xing, C. Lv, Z. Zhang, H. Wang, X. Na, D. Cao, E. Velenis, and F.-Y. Wang, "Identification and analysis of driver postures for in-vehicle driving activities and secondary tasks recognition," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 1, pp. 95–108, 2017.

[13] X. Luo, L. Niu, and S. Zhang, "An algorithm for traffic flow prediction based on improved sarima and ga," *KSCE Journal of Civil Engineering*, vol. 22, no. 10, pp. 4107–4115, 2018.

[14] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[15] B. M. Williams, "Multivariate vehicular traffic flow prediction: evaluation of arimax modeling," *Transportation Research Record*, vol. 1776, no. 1, pp. 194–200, 2001.

[16] M. S. Ahmed and A. R. Cook, *Analysis of freeway traffic time-series data by using Box-Jenkins techniques*, 1979, no. 722.
[17] M. Van Der Voort, M. Dougherty, and S. Watson, "Combining kohonen maps with arima time series models to forecast traffic flow," *Transportation Research Part C: Emerging Technologies*, vol. 4, no. 5, pp. 307–318, 1996.
[18] L. Zhang, Q. Liu, W. Yang, N. Wei, and D. Dong, "An improved k-nearest neighbor model for short-term traffic flow prediction," *Procedia-Social and Behavioral Sciences*, vol. 96, pp. 653–662, 2013.
[19] X. Luo, D. Li, Y. Yang, and S. Zhang, "Spatiotemporal traffic flow prediction with knn and lstm," *Journal of Advanced Transportation*, vol. 2019, 2019.
[20] X. Luo, D. Li, and S. Zhang, "Traffic flow prediction during the holidays based on dft and svr," *Journal of Sensors*, vol. 2019, 2019.
[21] C. Luo, C. Huang, J. Cao, J. Lu, W. Huang, J. Guo, and Y. Wei, "Short-term traffic flow prediction based on least square support vector machine with hybrid optimization algorithm," *Neural Processing Letters*, pp. 1–18, 2019.
[22] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert systems with applications*, vol. 36, no. 3, pp. 6164–6173, 2009.
[23] L. Zhao, Y. Zhou, H. Lu, and H. Fujita, "Parallel computing method of deep belief networks and its application to traffic flow prediction," *Knowledge-Based Systems*, vol. 163, pp. 972–987, 2019.
[24] H.-C. Shin, M. R. Orton, D. J. Collins, S. J. Doran, and M. O. Leach, "Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1930–1943, 2012.
[25] N. G. Polson and V. O. Sokolov, "Deep learning for short-term traffic flow prediction," *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 1–17, 2017.
[26] Z. Zhao, W. Chen, X. Wu, P. C. Chen, and J. Liu, "Lstm network: a deep learning approach for short-term traffic forecast," *IET Intelligent Transport Systems*, vol. 11, no. 2, pp. 68–75, 2017.
[27] R. Fu, Z. Zhang, and L. 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)*. IEEE, 2016, pp. 324–328.
[28] D. Zhang and M. R. Kabuka, "Combining weather condition data to predict traffic flow: a gru-based deep learning approach," *IET Intelligent Transport Systems*, vol. 12, no. 7, pp. 578–585, 2018.
[29] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, 2014.
[30] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2014.
[31] Y.-y. Chen, Y. Lv, Z. Li, and F.-Y. Wang, "Long short-term memory model for traffic congestion prediction with online open data," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 132–137.
[32] X. Dai, R. Fu, E. Zhao, Z. Zhang, Y. Lin, F.-Y. Wang, and L. Li, "Deeptrend 2.0: A light-weighted multi-scale traffic prediction model using detrending," *Transportation Research Part C: Emerging Technologies*, vol. 103, pp. 142–157, 2019.
[33] Y. Lin, X. Dai, L. Li, and F.-Y. Wang, "Pattern sensitive prediction of traffic flow based on generative adversarial framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2395–2400, 2018.
[34] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
[35] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.
[36] J. Cheng, P. Li, Z. Ding, S. Zhang, and H. Wang, "Sentiment classification of chinese microblogging texts with global rnn," in *2016 IEEE First International Conference on Data Science in Cyberspace (DSC)*. IEEE, 2016, pp. 653–657.
[37] Y. Liu, H. Zheng, X. Feng, and Z. Chen, "Short-term traffic flow prediction with conv-lstm," in *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 2017, pp. 1–6.
[38] J. Bergstra, D. Yamins, and D. D. Cox, "Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms," in *Proceedings of the 12th Python in science conference*. Citeseer, 2013, pp. 13–20.