Hyperparameter Analysis of Temporal Graph Convolutional Network Model Applied to Traffic Prediction

Jing Huang, Lei Chen(✉), Yuan An, Kailiang Zhang, and Ping Cui

Jiangsu Province Key Laboratory of Intelligent Industry Control Technology, Xuzhou University of Technology, Xuzhou 221018, China
chenlei@xzit.edu.cn

Abstract. GCN based on time and space is an essential part of smart city construction because it can capture the spatiotemporal dynamics and effectively analyze the traffic data to get the best prediction results. In the specific operation of the model, the adjustment and optimal selection of super parameters can make the model provide the best results, thus saving time, cost and computing power. When it comes to the prediction scenarios with low computational power and urgent demand, the existing super parameter search methods and optimization models lack efficiency and accuracy. Therefore, this paper proposes a super parameter search and optimization method based on cross validation, which can efficiently and accurately optimize the parameters, and select the best parameters by using the similarity between the learning and training errors corresponding to each super parameter To improve the prediction ability of the model. Through the verification of the actual data set, the model runs well, and can provide the best prediction results for the traffic flow and other scenarios dominated by spatiotemporal state.

Keywords: GCN · Machine learning · Hyperparameters optimization · Traffic prediction

1 Introduction

Accurate traffic forecast can help travelers to arrange their travel reasonably, improve the operation efficiency of traffic network, alleviate traffic congestion, improve other related service functions of the city, improve the utilization rate of road network and energy utilization rate, and reduce the emission of various traffic pollutants, which is an essential part of smart city construction. In 5g era, some new methods are proposed to improve network routing and measurement [1–3]. Based on an effective user behavior and traffic analysis method [4–7], a new scheduling strategy is proposed to improve resource utilization [8–11] and efficiency [12, 13]. A new traffic reconstruction method is proposed to approve the service quality of end users [14–19]. However, the traffic speed is ahead of time. Network communication is very important to improve the virtual communication oriented to time-varying network structure. Traffic flow prediction is a

H. Song and D. Jiang (Eds.): SIMUtools 2020, LNICST 369, pp. 681–693, 2021.
https://doi.org/10.1007/978-3-030-72792-5_53

typical spatiotemporal data prediction problem. How to mine the hidden spatiotemporal patterns from these complex and nonlinear spatiotemporal data and analyze these patterns to extract valuable information, In recent years, with the rapid development of deep learning [20–22], the deep learning network model has attracted people's attention because it can capture the dynamic. The traffic data are analyzed and the best results are obtained, among which the representative model is t-gcn. In the aspect of deep neural network model, the adjustment of super parameters is a necessary technology Yes, the so-called hyperparameters are the framework parameters in the deep neural network model. These super parameters act as knobs and can be tuned during model training, In order to make our model provide the best results, we can judge what kind of training state the current model is by observing the monitoring indicators such as loss and accuracy in the training process, and timely adjust the super parameters to train the model more scientifically, which can improve the resource utilization rate. In essence, hyperparametric search is an iterative process constrained by computational power, money and time. In the case of limited computing power, money and time resources, everyone wants to get the best model. However, in the spatiotemporal model represented by t-gcn, due to the special needs of spatiotemporal state and demand urgency, there is still a lack of effective super parameter search and optimization model to achieve this goal.

Therefore, based on the deep neural network learning model represented by t-gcn model, this paper proposes a super parameter search and optimization method. By studying the RMSE, ACC and loss results of the learning and training model, the relationship among the two important optimization parameters (learning rate, batch size) is analyzed, and the performance of the model in the actual operation process is analyzed Finally, the influence of these three parameters on the traffic prediction performance is analyzed. The proposed search and optimization method can effectively and accurately optimize the parameters, and provide the best prediction results for traffic flow and other scenarios dominated by spatiotemporal state.

2 Related Work

In recent years, with the rapid development of deep learning [20–22], the deep neural network model has attracted people's attention because it can capture the dynamic characteristics of traffic data and obtain the best results. Rilett et al. [23] used feedforward neural networks to perform traffic prediction tasks. Huang et al. [24] proposed a network structure composed of deep belief network (DBN) and regression model, and verified that the network could capture random features from traffic data of multiple data sets, which improved the accuracy of traffic prediction. In addition, because the recursive neural network (RNN) and its variant LSTM and gated recursive unit (GRU) can effectively utilize the self-cycling mechanism, they can learn the time dependence relationship well and obtain good prediction results [25, 26]. Considering the spatiotemporal dependence of urban traffic, a time-based graph convolutional neural network (T-GCN) model is proposed [27], which combines graph convolutional network and gated recursive unit. The graph convolutional network is used to capture the topological structure of the road network for modeling spatial dependence. The gated recurrent unit is used to capture the dynamic change of traffific data on the roads for modeling temporal dependence. As a new deep neural network model [28–30], T-GCN contains multiple hyperparameters that need to be selected and set in advance. Hyperparameter optimization problem can be defined as: for the hyperparameters to be set in the model, find the optimal hyperparameters setting, so that the deep learning model based on hyperparameter setting training has the optimal performance evaluation index.

The performance of deep neural networks is well known to be sensitive to the setting of their hyperparameters. Hyperparameter optimization is a very difficult problem in developing deep learning algorithms. Optimization of the hyperparameters in each algorithm are necessary to obtain the highest accuracy. Among the hyperparameters of deep neural network, the most important parameters are learning rate, batch size, optimizer and training epoch. Among them, batch size and learning rate directly determine the weight update of the model, and from the perspective of optimization, they are the most important parameters affecting the performance of the model. Many experts and scholars have done some research on the optimization method of super parameters in deep learning model. Deep learning models are typically trained using stochastic gradient descent or one of its variants. It has been observed that when using large batch sizes there is a persistent degradation in generalization performance - known as the "generalization gap" phenomena. To solve the problem that the traditional optimization algorithm cannot converge to the optimal solution (or the critical point in the non-convex setting), Sashank J. Reddi et al. [31] propose new variants of the ADAM algorithm which not only fix the convergence issues but often also lead to improved empirical performance. In order to improve the defects of the traditional super parameter self-optimization method, Nitish Shirish Keskar and Richard Socher [32] propose the SWATS units, a simple strategy which Switches from Adam to SGD The when a triggering condition is satisfied. The results show that the strategy is capable of closing the generalization on a gap between SGD and Adam Majority of the tasks and does not increase the number of hyperparameters in the optimizer. Hoffer et al. [33] studied the method of optimizing the hyperparameter batch size in deep learning. This method can achieve the effect of eliminating generalization and improving model performance, but this method is only for traditional deep learning models and has a single application object. Whether it is suitable for hyperparameter optimization in the temporal graph convolutional neural network model applied in traffic prediction needs further study. Goyal et al. [34] studied the optimization method of the superparameter batch size and learning rate by using the stochastic gradient descent method, and the research results showed that the superparameter optimization improved the accuracy and expansion rate of the visual discrimination model. Due to the essential difference between the urban traffic data and the Image Net data set, whether the optimization method can be directly migrated to the T-GCN model remains to be further studied. In literature [35, 36], the parameter optimization method of Adam, which is a common optimizer, is discussed in detail, including the characteristics of superparameter optimizer and the methods to improve performance. Literatures [37, 38] studied and analyzed the trend of learning rate curve and established an optimization model of super parameter learning rate, which reduced the learning time on the premise of maintaining the accuracy. However, the model has a single super parameter and a high limitation, which cannot be directly used in the new traffic flow prediction analysis related to T-GCN. Cardona-Escobar et al. [39] present an automatic framework for hyperparameter selection in Convolutional Neural Networks. In order to achieve fast evaluation of several hyperparameter combinations, prediction of learning curves using non-parametric regression models is applied. Results show that this forecasting method is able to catch a complete behavior offuture iterations in the learning process. The optimal combination of superparameters in the deep network learning model is a research direction worthy of our reference.

As a new deep neural network learning model, the optimal setting of its super parameters directly determines the traffic prediction performance of the model. In this paper, we studied and discussed in detail the influence of super parameter learning rate and batch size on the performance of T-GCN model, and provided the optimal super parameter selection.

3 Principle of T-GCN Model

The traffific information is a general concept which can be traffific speed, traffific flow, or traffific density. Without loss of generality, T-GCN model use traffific speed as an example of traffific information in experiment section. The T-GCN model establishes a traffic prediction model based on spatial dependence and spatial dependence:

3.1 Spatial Dependency Modeling

The T-GCN model use the GCN model [40–45] to learn spatial features from traffific data. The calculation process is shown in Eq. 1, $X(t)$ represents the traffic condition matrix at time t. We need to train a function $h(\cdot)$, which operates on the matrix of T times in the past, outputs the matrix of T times in the future, and makes traffic prediction.

$$\left[X^{(t-T+1)}, \cdots X^{(t)}; G\right] \underset{\rightarrow}{h(\cdot)} \left[X^{(t+1)}, \cdots X^{(t+T)}\right] \tag{1}$$

The T-GCN model use an unweighted graph $G = (V, E)$ to describe the topological structure of the road network, and we treat each road as a node, where V is a set of road nodes, $V = \{v_1, v_2, \cdots v_N,\}$, N is the number of the nodes, and E is a set of edges. The edge relation between nodes is represented by the adjacency matrix $A \in R^{N \times N}$.The adjacency matrix contains only elements of 0 and 1. The element is 0 if there is no link between two roads and 1 denotes there is a link.

In this research, the 2-layer GCN model [41] is chosen to obtain spatial dependence, which can be expressed as:

$$\mathrm{f(X, A)} = \sigma(\hat{\mathrm{A}}\mathrm{ReLU}(\hat{\mathrm{A}}\mathrm{XW}_0)\mathrm{W}_1) \tag{2}$$

where $\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ denotes pre-processing step, $W_0 \in R^{P \times H}$ represents the weight matrix from input to hidden layer, P is the length of feature matrix, and H is the number of hidden unit, $W_1 \in R^{N \times T}$ represents the weight matrix from hidden to output layer. $f(X, A) \in R^{N \times T}$ represents the output with the prediction length T, and ReLU() standing for REctified Linear Unit, which is a frequently used activation layer in modern deep neural networks, $\sigma(\cdot)$ represents the sigmoid function for a nonlinear model.

3.2 Time Dependent Modeling

Obtaining time dependence is another key problem in traffic forecasting. In the T-GCN model, the GRU model is selected to obtain the time dependence of traffic data [46, 47]. As shown in Fig. 1, h^{t-1} denotes the hidden state at time t−1, x^t denotes the traffific information at time t,r is the reset gate, z is the update gate, and h^t is output state at time t. The GRU obtains the traffific information at time t by taking the hidden status at time $t-1$ and the current traffific information as inputs. While capturing the traffific information at the current moment, the model still retains the changing trend of historical traffific information and has the ability to capture temporal dependence.

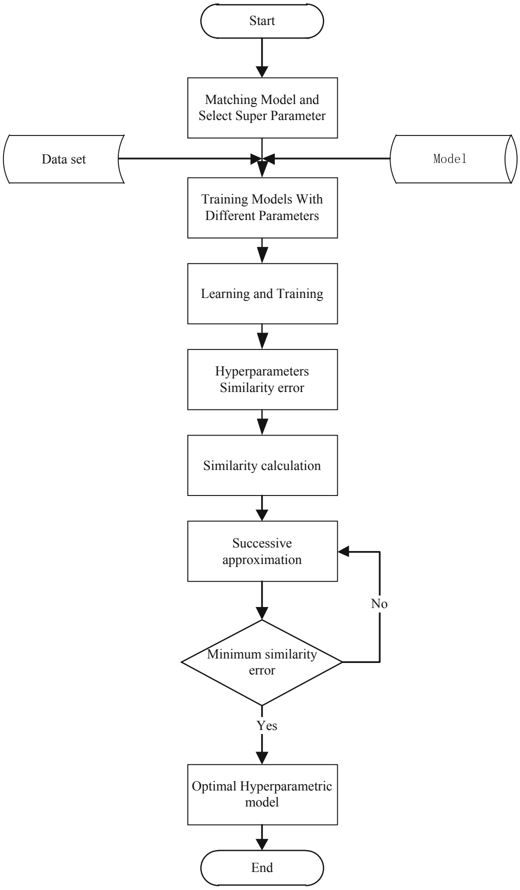


Fig. 1. Process optimization.

4 Deep Neural Network Hyperparameter and Performance Evaluation Model

4.1 Hyperparameter

In the deep neural network model, there are two sets of parameters: one is called elementary parameter, such as the weight and bias of the convolution layer or the full connection layer; the other is hyperparameter, such as the learning rate during network training and the coefficient of the L2 regularization item in the loss function. In practical application, to achieve good performance of deep neural network, it is very dependent on the selection of a good set of super parameters. However, in the training of deep neural network, the automatic learning is usually only the basic parameters, and the super parameters are mostly tried and selected in an optimized way. Among the hyperparameters of deep neural network, the most important parameters are learning rate and batch size.

Learning Rate
Learning raterefers to the extent of updating network weight in the optimization algorithm. If the learning rate is too high, the model may not converge, and the loss and loss constantly oscillate up and down. If the learning rate is too small, the convergence rate of the model will be slow, which requires longer training. Typically, the learning rate is randomly configured by the user. At best, users can only configure the best learning rate based on previous experience. Therefore, it is difficult to get a good learning rate.

Batch Size
Batch size is the number of samples sent into the model for each training neural network. In the convolutional neural network, the large number of batches can usually make the network converge faster, but due to the limitation of memory resources, too large a batch may lead to insufficient memory or the crash of the program kernel.

4.2 Optimization Method

In this paper, an efficient and accurate parameter optimization method is proposed to improve the prediction ability of the model. The main feature of the method is to learn and train directly on the original data set, and to select the best result by using the similarity of learning and training errors corresponding to each super parameter.

As shown in Fig. 1, the specific search and optimization process is as follows: model $\{(input_i, output_i), i = 1, 2, 3, \cdots, \mathrm{N}\}$ with super parameter x_l is trained successively on all data set $k(\bullet|\mathrm{x})$, and l trained candidate hyperparametric model $\{ \mathrm{k}(\bullet|\mathrm{x}_l), l = 1, 2, \cdots, L\}$ is obtained, Among them, $input_i$, $\mathrm{out}put_i$ is the input and output of the i data set, x in model $\mathrm{k}(\bullet|\mathrm{x})$ is the super parameter to be optimized, the candidate super-parameter set is $\mathrm{X} = \{x_l, l = 1, 2, \cdots, L\}$, and l is the number of super parameters; Through the model $k(\bullet|\mathrm{x}_\mathrm{l})$ of each super parameter x_l in $\mathrm{X} = \{x_l, l = 1, 2, \cdots, L\}$, the error in each data set $(input_i, output_i)$ is obtained: the difference between the predicted value $k(\mathrm{inp}ut_i|\mathrm{x}_\mathrm{l})$ of the model and the real result $output_i$ is recorded as $\mathrm{w}^{(\mathrm{x}_\mathrm{l})}$; Calculate the variance similarity matrix $\mathrm{T} = (\mathrm{t}_{\mathrm{mn}})_{L\times L}$, where $\mathrm{m}, n \in \{1, 2, \cdots, L\}$, t_{mn} in the matrix is $t_{mn} = \frac{1}{N}\sum_{i=1}^{N} Q(w^{(x_m)} \cdot w^{(x_n)} > 0)$, where e s Q(•) the indicator function; The symmetric similarity of the first parameter is the average value of all rows P and $2l - p$ in the directional similarity matrix. Row P number needs to satisfy $1 \leq p \leq l$, and the symmetric similarity degree $TT(x_l)$ of x_l is calculated according to $\frac{1}{\min(1, L-l+1)} \sum_{\substack{m+n=2l \\ m \leq n}} t_{mn}$.

The hyperparameter with minimum symmetric similarity is regarded as the optimal hyperparameter x* and returned.

4.3 Performance Evaluation Model

We selected three indexes of Root Mean Squared Error (RMSE), Accuracy (ACC) and loss function (loss) to intuitively evaluate the influence of superparameters in the model on the prediction performance. In model evaluation, accuracy is the most commonly used measurement. Its advantage is the classification of data samples, but the disadvantage is that it can only conduct surface analysis and can not identify deception. Therefore, we introduce rmse and loss degree to further assist the evaluation. Among them, variance refers to the statistical limit of the model. If the model is over trained or too complex for the given training data set, it will cause high rmse (over fitting), and the prediction performance of the model will be very poor. In the standard state, the loss of accuracy is the lowest. Our goal is to maximize the similarity between the model predictions and the results shown in the training data.

Among them, RMSE measurement and prediction error, loss function represents the error between the actual traffic speed and the predicted value. The larger the RMSE and loss value is, the worse the prediction effect is, while the smaller the value is, the better the prediction effect is. Accuracy means the accuracy of the prediction. The specific calculation formula is:

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{MN}} \sum_{\mathrm{j}=1}^{\mathrm{M}} \sum_{\mathrm{i}=1}^{\mathrm{N}} (\mathrm{y_i^j} - \hat{\mathrm{y}}_\mathrm{i}^\mathrm{j})^2} \tag{3}$$

$$\mathrm{ACC} = 1 - \frac{\|\mathrm{Y} - \hat{\mathrm{Y}}\|\mathrm{F}}{\|\mathrm{Y}\|\mathrm{F}} \tag{4}$$

$$\mathrm{loss} = \left\| \mathrm{Y}_t - \hat{\mathrm{Y}}_t \right\| + \lambda \mathrm{L}_{reg} \tag{5}$$

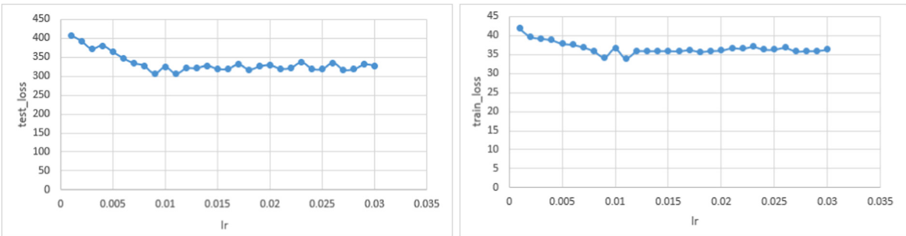
Where y_i^j and $\hat{y}_i^j$ represent the real traffific information and predicted one of the jth time sample in the ith road. is M the number of time samples; N is the number of roads; Y and $\hat{Y}$ represent the set of y_i^j and $\hat{y}_i^j$ respectively. L_{reg} is the L2 regularization term that helps to avoid an overfifitting problem and λ is a hyperparameter.

5 Experiments

In order to verify the influence of parameter optimization on the prediction performance in the GCN model, this paper conducts experiments on one real dataset (SZ—taxi dataset), takes the road network traffic speed as the input parameter of the model, and obtains the parameter optimization scheme through comparison experiments and analysis of experimental results. Data set SZ-taxi is the track of shenzhen taxi on January 31, 2015. 156 main roads in luohu district were selected as the research area. The experimental data mainly include two parts: one is the adjacency matrix of 156 * 156, which describes the spatial relationship between roads. The other is the eigenmatrix, which describes the change of speed on each road with time. Each row represents a road, and each column represents the speed of traffic on the road at different time periods. In the experiments, the input data is normalized to the interval [0, 1].

5.1 The Influence of Learning Rate on Prediction Performance

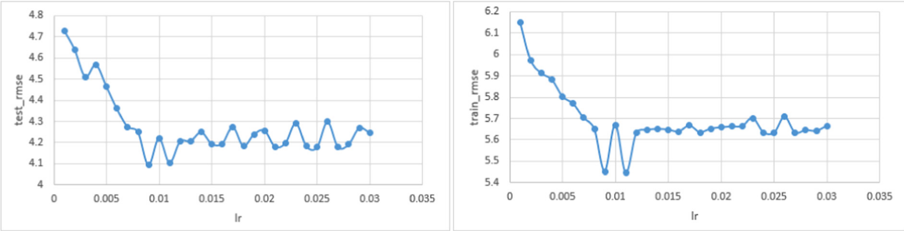
In the experiment, we manually adjusted and set the batch_size to 20, the training epoch to 550, the hidden units to 100, the time length of inputs to 12, the time length of prediction to 3 and the rate of training set to 0.8. In previous studies, the learning rate was usually set to 0.001, and the influence of the change of learning rate on the prediction performance was not studied. In this paper, we studied how the change of learning rate affected the prediction performance of the model on the premise of other parameters being fixed. Referring to the range of learning rates in deep learning, in this paper, the parameter range of the Initial learning rate was [0.001, 0.03], and the parameters were successively increasing by 0.001. The influence of learning rate on the loss function is shown in Fig. 2.



(a) (b)

Fig. 2. The trend of the loss function under different learning rate in the training and test set based on SZ-taxi dataset. (a) Changes in loss function in the test set. (b) Changes in loss function in the training set.

The results show that in the initial stage, train_loss and test_loss decrease slightly with the increase of learning rate, then tend to be stable, and then overall tend to be stable with small fluctuations. A relatively stable and rational loss function value can be obtained by setting the learning rate as 0.015. The variation trend of RMSE with learning rate is shown in Fig. 3.



(a) (b)

Fig. 3. The trend of the RMSE under different learning rate in the training and test set based on SZ-taxi dataset. (a) Changes in RMSE in the test set. (b) Changes in RMSE in the training set.

The results show that RMSE decreases significantly with the increase of learning rate in the initial stage, and then tends to be stable despite of fluctuations. When the learning rate is set to 0.015, the best root-mean-square error can be obtained. The trend of the prediction accuracy of the GCN model with the learning rate is shown in Fig. 4.

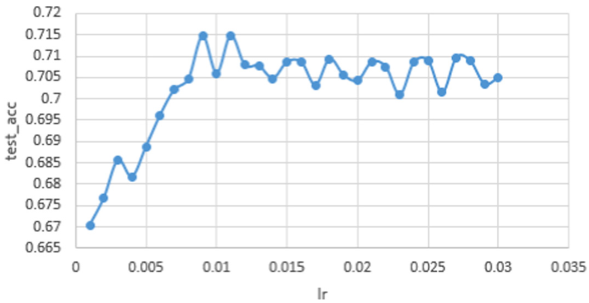


Fig. 4. The trend of the ACC under different learning rate based on SZ-taxi dataset.

The results show that with the increase of the learning rate, ACC presents an increasing trend, and when the learning rate is greater than 0.0125, ACC tends to be stable. At the learning rate of 0.015, a more rational prediction accuracy can be obtained. To sum up, in the GCN traffic prediction model, setting the learning rate as 0.015 can obtain better prediction performance.

5.2 The Influence of Batch_Size on Prediction Performance

In the process of studying the prediction performance of batch_size on model,we manually adjusted and set learning rate to 0.015, the training epoch to 500, the hidden units to 100, the time length of inputs to 12, the time length of prwediction to 3 and the rate of training set to 0.8. According to the principle of batch size Settings in depth study, in this experiment, the batch_size parameter range is [20, 30, 40, 50, 60, 70, 80, 90, 100, 110]. The influence trend of batch_size on the loss function is shown in Fig. 5.

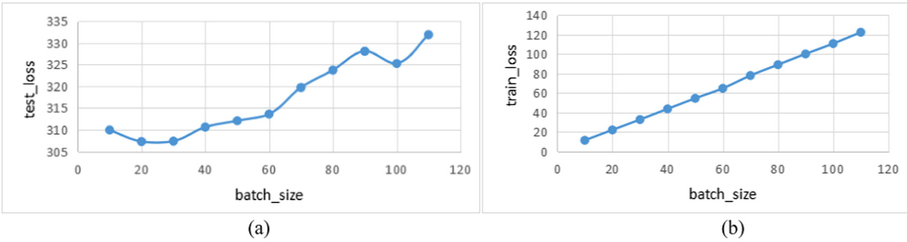


Fig. 5. The trend of the loss function under different batch_size in the training and test set based on SZ-taxi dataset. (a) Changes in loss function in the text set. (b) Changes in loss function in the training set.

And it turns out, with the increase of batch_size, the loss function shows an upward trend. When the value of batch_size is 32, a smaller loss function value can be obtained. The variation trend of RMSE value and ACC value with the learning batch size is shown in Fig. 6.

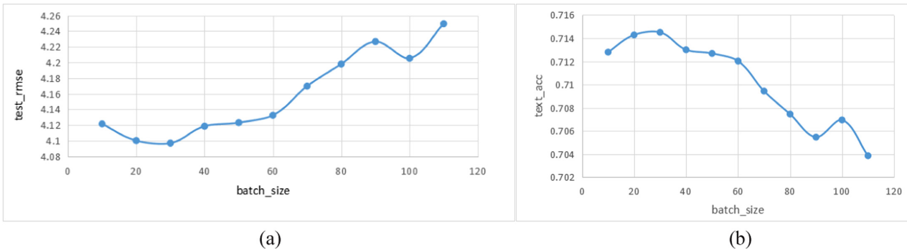


Fig. 6. The trend of the RMSE and ACC under different batch_size based on SZ-taxi dataset. (a) The variation trend of RMSE value with batch size. (b) The variation trend of ACC value with batch size.

The results show that RMSE value shows an upward trend with the increase of learning batch size, while ACC value shows a downward trend with the increase of learning batch size. When the learning batch size is 32, the minimum RMSE value and the maximum ACC value can be obtained, that is, the best prediction performance can be obtained. To sum up, in the GCN traffic prediction model, setting the batch_size as 32 can obtain better prediction performance.

6 Conclusion

This paper mainly studies the influence of super parameters on the prediction performance of traffic prediction model. Through a super parameter search and optimization method based on cross validation, the influence of three important factors, such as learning rate, learning batch size and training period, on traffic prediction performance is analyzed and discussed in detail. In the aspect of model evaluation, rmse, acc and loss function are used to evaluate the optimization degree of model performance. The results show that the super parameter search and optimization method can effectively and accurately optimize and select parameters, the model runs well, and can provide the best prediction results for traffic flow and other scenarios dominated by spatiotemporal state.

Acknowledgements. This work is partly supported by Jiangsu technology project of Housing and Urban-Rural Development (No. 2018ZD265, No. 2019ZD039, No. 2019ZD040, No. 2019ZD041).

References

1. Zhang, K., Chen, L., An, Y., et al.: A QoE test system for vehicular voice cloud services. Mob. Netw. Appl. (2019). https://doi.org/10.1007/s11036-019-01415-3
2. Wang, F., Jiang, D., Qi, S.: An adaptive routing algorithm for integrated information networks. China Commun. **7**(1), 196–207 (2019)
3. Huo, L., Jiang, D., Lv, Z., et al.: An intelligent optimization-based traffic information acquirement approach to software-defined networking. Comput. Intell. 1–21 (2019)
4. Chen, L., Jiang, D., Bao, R., Xiong, J., Liu, F., Bei, L.: MIMO Scheduling effectiveness analysis for bursty data service from view of QoE. Chin. J. Electron. **26**(5), 1079–1085 (2017)
5. Jiang, D., Wang, Y., Lv, Z., et al.: Big data analysis-based network behavior insight of cellular networks for industry 4.0 applications. IEEE Trans. Ind. Inform. **16**(2), 1310–1320 (2020)
6. Jiang, D., Huo, L., Song, H.: Rethinking behaviors and activities of base stations in mobile cellular networks based on big data analysis. IEEE Trans. Netw. Sci. Eng. **1**(1), 1–12 (2018)
7. Chen, L., et al.: A lightweight end-side user experience data collection system for quality evaluation of multimedia communications. IEEE Access **6**(1), 15408–15419 (2018)
8. Chen, L., Zhang, L.: Spectral efficiency analysis for massive MIMO system under QoS constraint: an effective capacity perspective. Mob. Netw. Appl. (2020). https://doi.org/10.1007/s11036-019-01414-4
9. Wang, F., Jiang, D., Qi, S., et al.: A dynamic resource scheduling scheme in edge computing satellite networks. Mob. Netw. Appl. 1–12 (2019)
10. Jiang, D., Huo, L., Lv, Z., et al.: A joint multi-criteria utility-based network selection approach for vehicle-to-infrastructure networking. IEEE Trans. Intell. Transp. Syst. **19**(10), 3305–3319 (2018)
11. Jiang, D., Zhang, P., Lv, Z., et al.: Energy-efficient multi-constraint routing algorithm with load balancing for smart city applications. IEEE Internet Things J. **3**(6), 1437–1447 (2016)
12. Jiang, D., Li, W., Lv, H.: An energy-efficient cooperative multicast routing in multi-hop wireless networks for smart medical applications. Neurocomputing **220**, 160–169 (2017)
13. Jiang, D., Wang, Y., Lv, Z., et al.: Intelligent optimization-based reliable energy-efficient networking in cloud services for IIoT networks. IEEE J. Select. Areas Commun. 1–6 (2019)
14. Jiang, D., Wang, W., Shi, L., et al.: A compressive sensing-based approach to end-to-end network traffic reconstruction. IEEE Trans. Netw. Sci. Eng. **5**(3), 1–12 (2018)
15. Jiang, D., Huo, L., Li, Y.: Fine-granularity inference and estimations to network traffic for SDN. PLoS ONE **13**(5), 1–23 (2018)
16. Wang, Y., Jiang, D., Huo, L., et al.: A new traffic prediction algorithm to software defined networking. Mob. Netw. Appl. 1–10 (2019)
17. Qi, S., Jiang, D., Huo, L.: A prediction approach to end-to-end traffic in space information networks. Mob. Netw. Appl. 1–10 (2019)
18. Huo, L., Jiang, D., Qi, S., et al.: An AI-based adaptive cognitive modeling and measurement method of network traffic for EIS. Mob. Netw. Appl. 1–11 (2019)

19. Huo, L., Jiang, D., Zhu, X., et al.: An SDN-based fine-grained measurement and modeling approach to vehicular communication network traffic. Int. J. Commun. Syst. 1–12, (2019)
20. Silver, D., et al.: Mastering the game of Go with deep neural networks and tree search. Nature **529**(7587), 484–489 (2016)
21. Silver, D., et al.: Mastering the game of go without human knowledge. Nature **550**(7676), 354–359 (2017)
22. Moravˇcik, M., et al.: DeepStack: Expert-level artifificial intelligence in heads-up no-limit poker. Science **356**(6337), 508–513 (2017)
23. Park, D., Rilett, L.R.: Forecasting freeway link travel times with a multilayer feedforward neural network. Comput.-Aided Civil Infrastruct. Eng. **14**(5), 357–367 (1999)
24. Huang, W., Song, G., Hong, H., Xie, K.: Deep architecture for traffific flow prediction: Deep belief networks with multitask learning. IEEE Trans. Intell. Transp. Syst. **15**(5), 2191–2201 (2014)
25. Fu, R., Zhang, Z., Li, L.: Using LSTM and GRU neural network methods for traffific flow prediction. In: 31st Youth Academic Annual Conference China Association Automation (YAC), Wuhan, China, pp. 324–328 (2016)
26. Van Lint, J.W.C., Hoogendoorn, S.P., van Zuylen, H.J.: Freeway travel time prediction with state-space neural networks: modeling statespace dynamics with recurrent neural networks. Transp. Res. Rec. **1811**(1), 30–39 (2002)
27. Zhao, L., Song, Y., Zhang, C., et al.: T-GCN: a temporal graph convolutional network for traffic prediction. IEEE Trans. Intell. Transp. Syst. **21**(9), 3848–3858 (2018)
28. Ding, L., Huang, Z., Chen, G.: An FPGA implementation of GCN with sparse adjacency matrix. In: 2019 IEEE 13th International Conference on ASIC (ASICON) (2019)
29. Zheng, J., Li, D.: GCN-TC: combining trace graph with statistical features for network traffic classification. In: 2019 IEEE International Conference on Communications (ICC) (2019)
30. Li, Z., Xiong, G., Chen, Y.: A hybrid deep learning approach with GCN and LSTM for traffic flow prediction. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (2019)
31. Reddi, S.J., Kale, S., Kumar, S.: On the Convergence of Adam and Beyond (2019)
32. Keskar, N.S., Socher, R.: Improving Generalization Performance by Switching from Adam to SGD (2017)
33. Hoffer, E., Hubara, I., Soudrym D.: Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In: Advances in Neural Information Processing Systems, pp. 1731–1741 (2017)
34. Goyal, P., Dollar, P., Girshick, R.B., et al.: Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. arXiv: Computer Vision and Pattern Recognition (2017)
35. Keskar, N.S., Socher, R.: Improving generalization performance by switching from adam to sgd. arXiv preprint arXiv:1712.07628 (2017)
36. Reddi, S.J., Kale, S., Kumar, S.: On the convergence of adam and beyond (2018)
37. Smith, L.N.: Cyclical learning rates for training neural networks. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 464–472, IEEE (2017)
38. Smith, S.L., Kindermans, P.J., Ying, C., et al.: Don't decay the learning rate, increase the batch size. arXiv preprint arXiv:1711.00489 (2017)
39. Cardona-Escobar, A.F., Giraldo-Forero, A.F., Castro-Ospina, A.E., Jaramillo-Garzón, F.A.: Efficient hyperparameter optimization in convolutional neural networks by learning curves prediction. In: Mendoza, M., Velastín, S. (eds.) CIARP 2017. LNCS, vol. 10657, pp. 143–151. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75193-1_18
40. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral fifiltering. Proc. Adv. Neural Inf. Process. Syst., 3844–3852 (2016)
41. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks (2016). https://arxiv.org/abs/1609.02907

42. Bruna, J., Zaremba, W., Szlam, A., Lecun, Y.: Spectral networks and locally connected networks on graphs. https://arxiv.org/abs/1312.6203 (2013)
43. Ma, Y., et al: High performance graph convolutional networks with applications in testability Analysis. In: ACM/IEEE Design Automation Conference (DAC), Las Vegas, NV, pp. 18:1–18:6 (2019)
44. Forecasting road traffic speeds by considering area-wide spatio temporal dependencies based on a graph convolutional neural network (GCN). In: 2019 Chinese Control Conference (CCC) (2019)
45. Wu, C., Chai, L., Yang, J., Sheng, Y.: Facial expression recognition using convolutional neural network on graphs. In: The 38th China Control Conference, pp. 90–94 (2019)
46. Cho, K., van Merrienboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: Encoder-decoder approaches. https://arxiv.org/abs/1409.1259 (2014)
47. Chung, J., Gulcehre, C., Cho, K.H., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. https://arxiv.org/abs/1412.3555 (2014)