Data & Knowledge Engineering 146 (2023) 102193

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

A new approach to COVID-19 data mining: A deep spatial–temporal prediction model based on tree structure for traffic revitalization index

Zhiqiang Lv [a,c,1], Xiaotong Wang [a,1], Zesheng Cheng [a,*], Jianbo Li [a], Haoran Li [a,b], Zhihao Xu [a,b]

[a] *College of Computer Science & Technology, Qingdao University, Qingdao 266071, China*
[b] *Institute of Ubiquitous Networks and Urban Computing, Qingdao 266070, China*
[c] *Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China*

ARTICLE INFO

Dataset link: https://github.com/qdu318/DeepTree-TRI.git

Keywords:
COVID-19
Traffic revitalization index
Spatial–temporal model
Directional feature
Hierarchical feature

ABSTRACT

The outbreak of the COVID-19 epidemic has had a huge impact on a global scale and its impact has covered almost all human industries. The Chinese government enacted a series of policies to restrict the transportation industry in order to slow the spread of the COVID-19 virus in early 2020. With the gradual control of the COVID-19 epidemic and the reduction of confirmed cases, the Chinese transportation industry has gradually recovered. The traffic revitalization index is the main indicator for evaluating the degree of recovery of the urban transportation industry after being affected by the COVID-19 epidemic. The prediction research of traffic revitalization index can help the relevant government departments to know the state of urban traffic from the macro level and formulate relevant policies. Therefore, this study proposes a deep spatial–temporal prediction model based on tree structure for the traffic revitalization index. The model mainly includes spatial convolution module, temporal convolution module and matrix data fusion module. The spatial convolution module builds a tree convolution process based on the tree structure that can contain directional features and hierarchical features of urban nodes. The temporal convolution module constructs a deep network for capturing temporal dependent features of the data in the multi-layer residual structure. The matrix data fusion module can perform multi-scale fusion of COVID-19 epidemic data and traffic revitalization index data to further improve the prediction effect of the model. In this study, experimental comparisons between our model and multiple baseline models are conducted on real datasets. The experimental results show that our model has an average improvement of 21%, 18%, and 23% in MAE, RMSE and MAPE indicators, respectively.

1. Introduction

1.1. *Traffic revitalization index*

The outbreak of the COVID-19 virus [1] in 2020 has had a huge impact on global economic development. Many countries have imposed restrictions on the transportation industry to slow the spread of the virus [2]. The urban Traffic Revitalization Index (TRI) [3] is an important indicator to measure the recovery status of urban traffic after being affected by the epidemic. It is obtained

[*] Corresponding author.
E-mail addresses: lvzhiqiang@ubinet.cn (Z. Lv), chengzesheng@ubinet.cn (Z. Cheng).
[1] Zhiqiang Lv and Xiaotong Wang are contributed equally to this work.

https://doi.org/10.1016/j.datak.2023.102193
Received 10 October 2022; Received in revised form 23 March 2023; Accepted 8 May 2023
Available online 16 May 2023

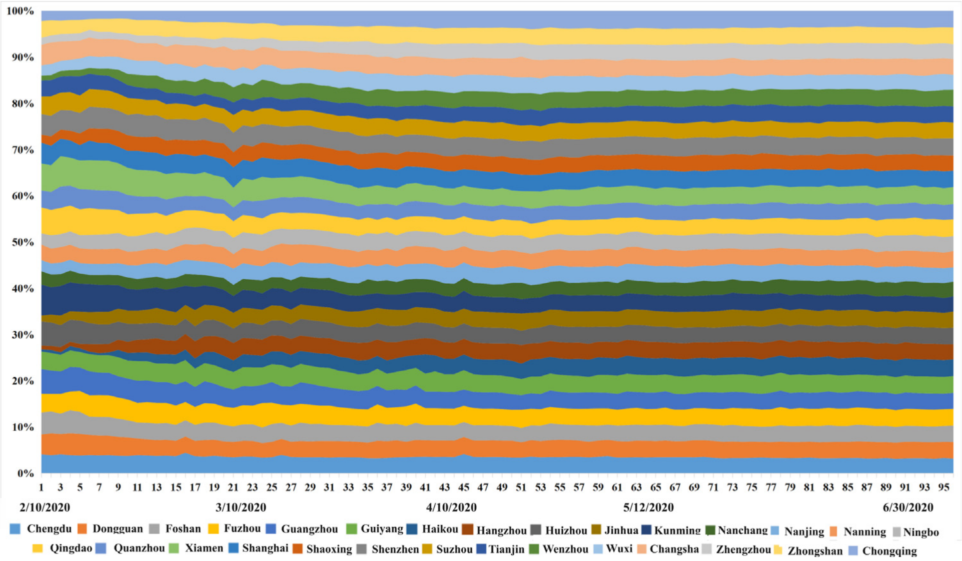


Fig. 1. The percent stacked area chart of TRI for Chinese core cities. The horizontal axis represents the date and the vertical axis represents the percentage.

through comprehensive analysis of data such as urban traffic trajectory status, traffic operation index and traffic commuting status. Fig. 1 shows the changes in TRI in 29 cities in China. In the early days of the COVID-19 epidemic, traffic control restrictions to control the spread of the epidemic caused a rapid decline in TRI in various cities [4]. As the epidemic stabilized, the TRI in each city gradually recovered and tended to balance. As of June 2020, the TRI of all cities was above 0.9, which means that the urban transportation industry has gradually returned to normal status [5].

The research on TRI is of great significance, because the TRI can not only directly reflect the urban traffic vitality [6], but also indirectly reflect the urban economic development. As an important part of the national economy, the transportation industry interacts with and promotes economic development. Transportation industry can drive economic activities, and economic development can promote the construction of the transportation industry [7]. The Gross Domestic Product (GDP) [8] is the market value of the production results of all economic activities in a region over a certain period of time. The GDP is an important indicator to measure urban development, and its development can reflect the effectiveness of the implementation of urban economic policies in improving urban economic development [9]. Fig. 2 shows the GDP and TRI data of Chinese core cities in the first two quarters of 2020. With the exception of individual cities whose TRI data declined due to the COVID-19 outbreak, most urban TRI data in the second quarter were higher than those in the first quarter. Under the influence of the recovery of traffic vitality, the GDP of most cities in the second quarter was significantly higher than that of the first quarter. Due to the dynamic correlation between TRI and GDP data, the TRI data can intuitively reflect the recovery of urban traffic, which to a certain extent reflects the state of urban economic development. Therefore, the prediction of TRI can refer to assisting the government to formulate measures according to the current situation to promote urban economic recovery and further development [10].

The prediction research of TRI can be regarded as a branch of traffic prediction, and there are two main approaches to solve this problem [11]. The first approach is to solve short-term traffic prediction problems based on the research of traditional mathematical statistics [12], such as time series models, parametric regression models [13], etc. The second approach is to train models based on deep learning. This kind of model method does not pursue strict mathematical derivation and physical definition [14], but attaches great importance to the fitting effect of real traffic flow. There are three main approaches of deep learning in the field of traffic prediction. The first approach is deep autoencoder neural network, which is mainly used for short-term traffic congestion prediction [15]. The second approach is the Recurrent Neural Network (RNN) [16], which is mainly used to solve the prediction problem of time series data. The third approach is the Convolutional Neural Network (CNN) [17], which are mainly used to capture the spatial correlation features of transportation networks. At present, the research of deep learning focuses on the fusion of different neural network structures, absorbing the advantages of different architectures, so that the model has better accuracy and stability [18]. The traffic condition of the target area is affected by the surrounding area or the further area, which makes the urban traffic area have a certain spatial dependence [19]. To solve the problem of spatial dependence of traffic data, researchers use graph topology [20] to represent traffic data, and set different adjacency matrices to represent graph topology according to different data characteristics.

1.2. Graph hierarchy weaknesses

Although the graph can represent the spatial relationship between the nodes of the transportation network, it cannot fully capture the hierarchical and directional features [21] of the topology. The graph with the spatial locations of cities as nodes can be processed

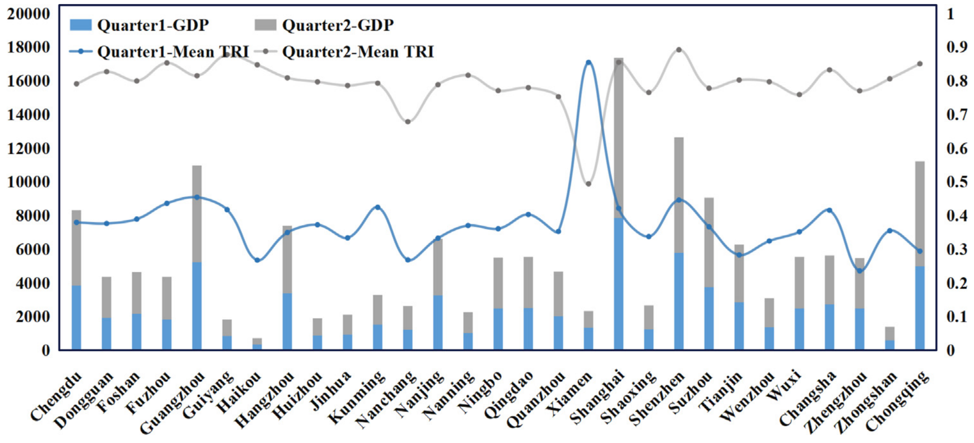


Fig. 2. The GDP and TRI data for the first two quarters of Chinese core cities in 2020. The abscissa is the urban name. The main ordinate is the GDP, whose unit is 100 million yuan. The secondary ordinate is TRI, whose value range is [0, 1].

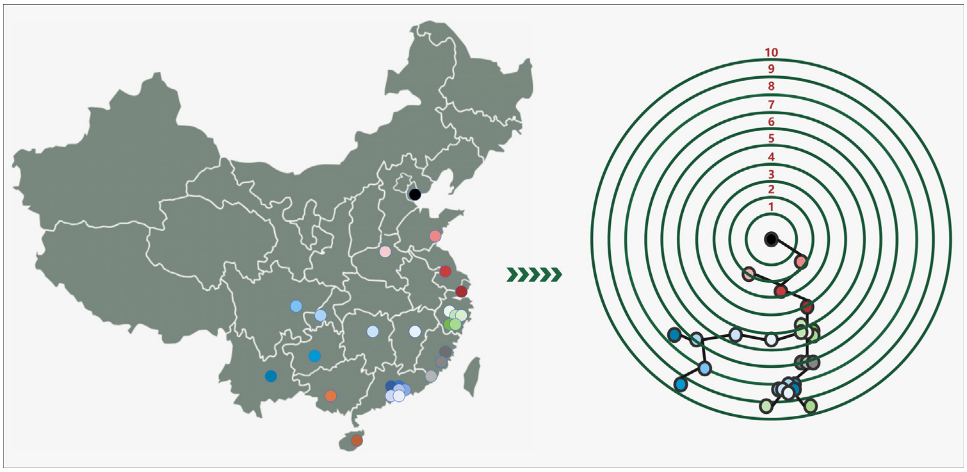


Fig. 3. Spatial distribution of core cities in China and hierarchical distribution of graph nodes.

by Graph Convolutional Network (GCN) [22] to capture the spatial correlation information among cities. In the application scenario of GCN, the hierarchical features of the graph are introduced to represent the degree of influence among nodes. The location distribution of Chinese core cities is shown in Fig. 3. Taking a node in Fig. 3 as the starting point and increasing the number of rings from the starting point outward. The purpose is to make the topology of the graph cover more nodes. However, the researchers found through experiments that the capturing process of the spatial characteristics of graph nodes will lose its meaning in a single-layer GCN when the number of layers in the graph exceeds 7 layers [23]. It can be seen from the above process that although the topological structure of the graph is used to represent the influence relationship among cities, the influence of cities with different distances on the central city is quite different. In the urban spatial relationship, the role of information transmission between cities decreases with the increase of distance [24]. At the same time, the traffic prediction researches based GCN usually represent traffic areas as nodes in an undirected graph, which leads to the lack of directional features between nodes in the modeling process.

1.3. Hierarchical characteristics among nodes

The hierarchy of city nodes is an important characteristic of relative spatial distribution, which is reflected in this study that there are different degrees of differences in the epidemic data changes of city nodes in different hierarchical areas. At the same time, the TRI data is affected by the development of the epidemic, so the hierarchy of nodes can be reflected in the changes in TRI data of nodes in different areas.

The Fig. 4(a) shows the location distribution of six cities with different hierarchies, and the six cities are divided into two city areas. The (c) and (b) respectively show the changes of daily new confirmed cases in different cities and the corresponding TRI data changes in city nodes in two city areas. Because the COVID-19 epidemic is highly contagious, the emergence of more obvious confirmed cases in a city has a high probability of causing obvious epidemic development in neighboring cities. The statistical results

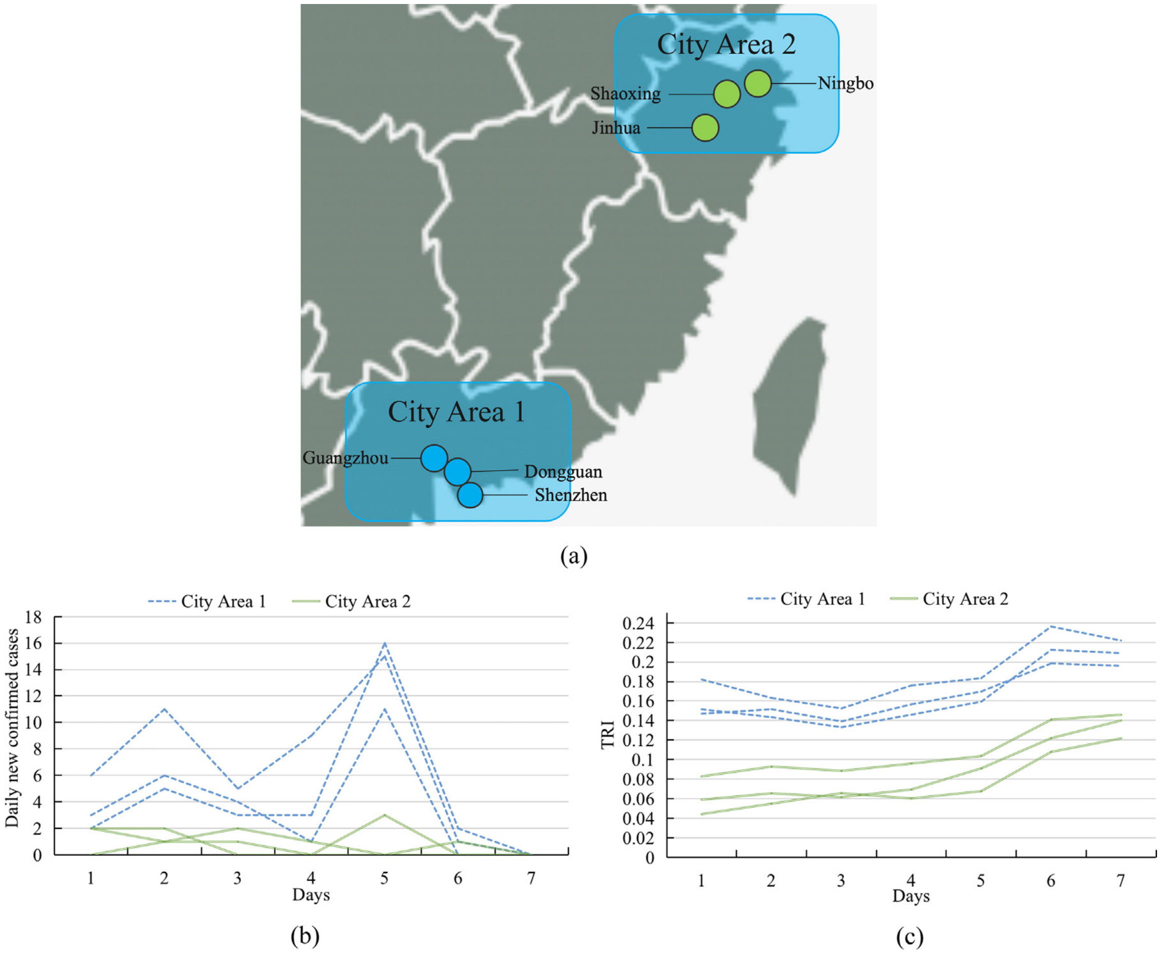


Fig. 4. The (a) shows the distribution of two different areas and the corresponding cities within the areas. The city area 1 include Shenzhen, Guangzhou, and Dongguan. The city area 2 include Ningbo, Shaoxing, and Jinhua. The (b) shows the changes of daily new confirmed cases in different cities. The (c) shows the changes of TRI data in different cities.

of the data in Fig. 4 show that the epidemic development status of city nodes and the TRI data changes in the same area have a certain degree of synchronization, which reflects that the nodes with relatively close spatial distribution relationships have almost the same level of hierarchical characteristics. This study expects to further capture the hierarchical characteristics between nodes on the basis of graph through tree structure. At the same time, we group the cities in the same area into one category and regard them as fully connected node regions. For example, although Guangzhou, Dongguan, and Shenzhen are three different cities, the distance among them is relatively close compared to the distance relationship among the 29 Chinese cores involved in the dataset of this study. From the overall perspective of the entire map of China, cities in the same area are gathered together and connected to each other. Therefore, this study regards the nodes in the same area as having fully connected properties when designing the directionality and hierarchy among the tree structure nodes.

1.4. Contributions

The TRI data of cities have high spatial correlation and time dependence, and the relative spatial distribution between cities has a certain degree of hierarchical relationship. Therefore, this study designs a deep spatial–temporal model (named DeepTree-TRI) that can combine urban spatial distribution and COVID-19 epidemic data to complete the prediction task of TRI data. The DeepTree-TRI is mainly composed of a temporal convolution module and a spatial convolution module, and makes full use of the matrix fusion module to achieve multi-scale fusion of COVID-19 epidemic data and TRI data. The specific contributions of this study are as follows.

- This study proposes a spatial convolution module. The core idea of this module is that the spatial distribution state of nodes is represented by a tree structure, which is to overcome the shortcomings of the graph structure. The tree structure consists of a space tree matrix and a plane tree matrix. The connectivity and distance relationships between nodes representing cities are reasonably transformed into directional and hierarchical relationships between nodes. At the same time, this study constructs a tree convolutional network acting on the tree structure.

- This study proposes a temporal convolution module consisting of multiple layers of dilated convolution and causal convolution [25]. The convolutional layers are connected through residual structures [26] to enforce the temporal dependencies between convolutional layers.
- This study proposes a matrix fusion module for COVID-19 epidemic data. This module integrates the TRI data with the confirmed cases, death cases, suspected cases, cured cases in the COVID-19 epidemic data at multiple scales. Because the TRI data has a high correlation with the changes of the COVID-19 epidemic, the matrix fusion module can further improve the prediction effect of the TRI data.
- This study conducts comparative experiments on DeepTree-TRI and other baselines in real datasets. The experimental results show that the DeepTree-TRI can fully integrate the spatial distribution features of cities, the temporal dependent features of TRI and the multi-scale impact features of COVID-19 epidemic data.

2. Related works

2.1. Prediction research of traffic revitalization index

The TRI is an important measure of urban transportation recovery in the wake of the COVID-19 outbreak. The deep spatial–temporal model proposed by Lv et al. [3] utilizes TRI data, COVID-19 epidemic data and the graph topology of urban spatial distribution to realize the prediction task of TRI. The model utilizes GCN to capture spatial correlation features of urban geographic locations. Various types of information in the COVID-19 epidemic data are fused with TRI data in different proportions. The DeepMeta-TRI model proposed by Wang et al. [27] is a deep spatial–temporal meta-learning model that uses external auxiliary information to predict TRI changes. The DeepMeta-TRI designs a meta-GCN module and a meta-temporal convolution module, which can incorporate external information from multiple contexts.

2.2. Prediction research of spatial–temporal model

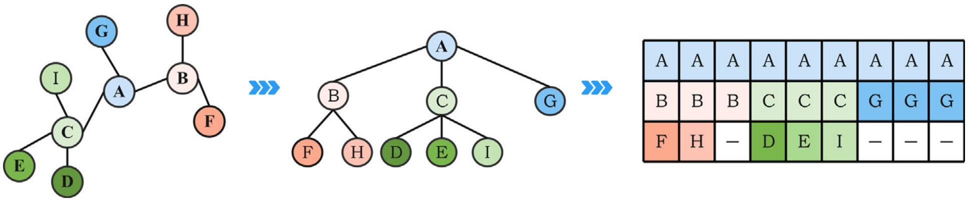
The initial TRI data research was based on the topology of the graph. This is because there is a strong spatial correlation between nodes representing city information or traffic information, traditional statistical methods, the CNN and the RNN cannot represent nodes well [28]. Therefore, the GCN is widely used in the representation of node spatial distribution. The basic GCN mainly includes spatial graph volumes that aggregate the information of adjacent nodes in the graph (such as diffusion convolutional recurrent neural network [29] and network models based on residual recursive [30]) and frequency graph convolution using graph signal processing and filters (such as global spatial–temporal network [31] and dynamic spatial–temporal graph convolutional neural network [32]). The influence of temporal dependent features is greater than spatial correlation features for time series prediction tasks. However, the models based GCN are not good at capturing the temporal dependent features of time series data. Therefore, researchers propose to combine GCN with RNN or temporal convolutional network to capture the spatial–temporal features of time series data [33]. The Temporal Graph Convolutional Network (T-GCN) proposed by Zhao et al. [34] builds an adjacency matrix from the connectivity relationship among nodes as the input of GCN. The T-GCN realizes the capture of spatial correlation features with the help of GCN and the capture of temporal dependent features with Gated Recurrent Unit (GRU). The GRU is an improved model of traditional RNN, which cannot overcome the problem of gradient disappearance in the recursive process. Therefore, Yu et al. [35] propose the Spatial–Temporal Graph Convolutional Network (STGCN). The STGCN consists of two temporal convolutional layers and one spatial convolutional layer. While the spatial convolutional layers use the same structure as T-GCN, the temporal convolutional layers are entirely composed of CNNs. The full application of STGCN to CNN makes it not only overcome the problem of gradient disappearance in the process of model training, but also improves the calculation speed when processing large-scale graph data with the help of the parallel computing capability of CNN. The Graph Convolutional Network based on Spatial–Temporal Attention (ASTGCN) proposed by Guo et al. [36] adds spatial–temporal attention mechanism and temporal periodicity features to STGCN. The spatial–temporal attention mechanism is used to learn dynamic spatial–temporal associations of traffic data. The temporal periodicity feature includes the temporal dependent features at three time steps.

Although the above models have certain effects in the prediction task of traffic data, they all use graphs as the main topology to represent the spatial distribution of cities. The adjacency matrix used to store the topological structure of the graph only involves the distance relationship between cities, which lacks the connectivity relationship between cities. More importantly, the above studies rely on the fact that the adjacency matrix of undirected graphs lacks the directional information interaction feature among nodes. The adjacency matrix of a graph also cannot build hierarchical relationships between nodes. Therefore, the process of GCN cannot fully capture the directional and hierarchical features among nodes [37].

3. Data design

3.1. Problem definition

The research goal of this study is to achieve the task of predicting future changes in TRI data in different cities, as shown in Eqs. (1), (2), (3) and (4). In the problem definition, cities are regarded as nodes, and the connectivity between cities is regarded as



A	A	A	A	A	A	A	A	A
B	B	B	C	C	C	G	G	G
F	H	–	D	E	I	–	–	–

Fig. 5. Plane tree matrix design.

an edge connecting two nodes. The N represents the total number of cities. The X_N^t represents the TRI data for the first t time steps in N cities. The M_N^t represents the COVID-19 epidemic data for the first t time steps in N cities. The T represents a tree structure representing the spatial distribution among nodes, which is represented by a spatial tree matrix. Each node will generate its corresponding two-dimensional plane tree matrix T' (The $l_{T'}$ and $w_{T'}$ represent the length and width of the plane tree matrix, respectively). The T is a three-dimensional matrix composed of all plane tree matrices, which endows the spatial feature information of TRI data by fusion with X_N^t. The f represents the modeling process of the entire task, which will eventually output the predicted TRI values y_N^{t+1} of N cities at the next time step.

$$y_N^{t+1} = f\left(X_N^t, T, M_N^t\right) \in \mathbb{R}^N \tag{1}$$

$$X_N^t = (x_N^1, x_N^2, \dots, x_N^t) \in \mathbb{R}^{N\times t} \tag{2}$$

$$T = (T_1', T_2', \dots, T_N') \in \mathbb{R}^{l_{T'}\times w_{T'}\times N} \tag{3}$$

$$M_N^t = (m_N^1, m_N^2, \dots, m_N^t) \in \mathbb{R}^{N\times t} \tag{4}$$

The COVID-19 epidemic data used in this study specifically includes confirmed cases (c), suspected cases (s), cured cases (u) and death cases (d). The data design process is shown in Eq. (5). The m_N^t represents the COVID-19 epidemic data of N cities at time step t. Four types of COVID-19 epidemic data are formed by the *concat*. The r represents the proportional relationship of the four types of COVID-19 epidemic data in m, which is to highlight the impact of different types of COVID-19 epidemic data on the final prediction results.

$$m_N^t = concat(c_N^t \times r_c, s_N^t \times r_s, u_N^t \times r_u, d_N^t \times r_d) \tag{5}$$

3.2. *Tree matrix design*

In order to solve the problem that the graph topology structure cannot realize the information transfer of hierarchical and directional features among nodes, this paper abstracts and regularizes the urban spatial data according to the tree structure. Before constructing the tree structure, whether there is a connectivity relationship between city nodes needs to be defined [38]. We group cities with the same geographical distribution and close spatial distance into one category, and the same kind of cities are fully connected. Two cities that are closer in space are constructed to be connected in different types of cities. The above design process not only takes into account the influence between cities in different geographical locations, but also enables the inclusion of hierarchical relationships between cities.

According to the connectivity information of the spatial distribution of city nodes, each city node goes through the process of establishing a plane tree matrix, as shown in Fig. 5. Taking the node A as an example, the A is set as the root node of the tree. The process of breadth-first traversal traces back all remaining nodes [39]. Because different root nodes will generate trees with different structures through breadth-first traversal, the size of the plane tree matrix should be able to accommodate the tree with the largest depth tree and the largest breadth tree. The size of the plane tree matrix is defined as Eq. (6). The α represents the maximum level of all trees generated by breadth-first traversal. The β represents the maximum number of children of all nodes of the tree generated by breadth-first traversal. If a generated tree cannot fill the plane tree matrix, we will add zero values to the plane tree matrix to ensure that all the plane tree matrices have the same size.

$$T' \in \mathbb{R}^{\alpha\times\beta^{\alpha-1}} \tag{6}$$

The plane tree matrix contains hierarchical information and directional information that the graph structure lacks. The plane tree matrix reflects the hierarchical information of nodes by storing nodes of the same level of the tree in the same row of the matrix. The directional information between nodes is reflected in the information stored in each column of the plane tree matrix, which is a complete sequence from the root node to the leaf node of the tree. Since the plane tree matrices generated from all nodes have the same size, we can easily concatenate all the plane tree matrices to form a spatial tree matrix representing the spatial distribution of all nodes, as shown in Fig. 6.

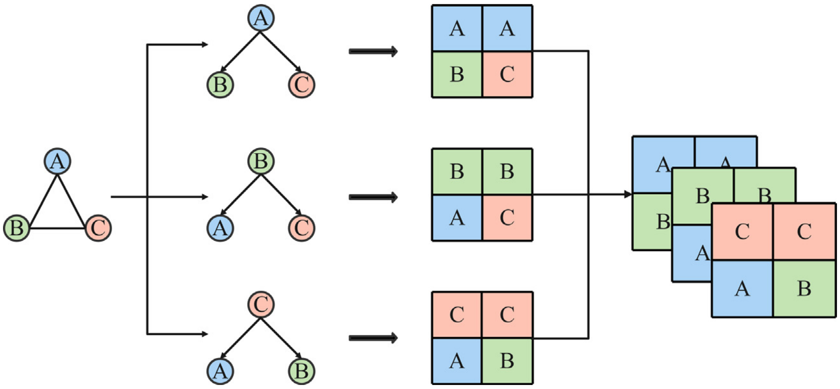


Fig. 6. Spatial tree matrix design.

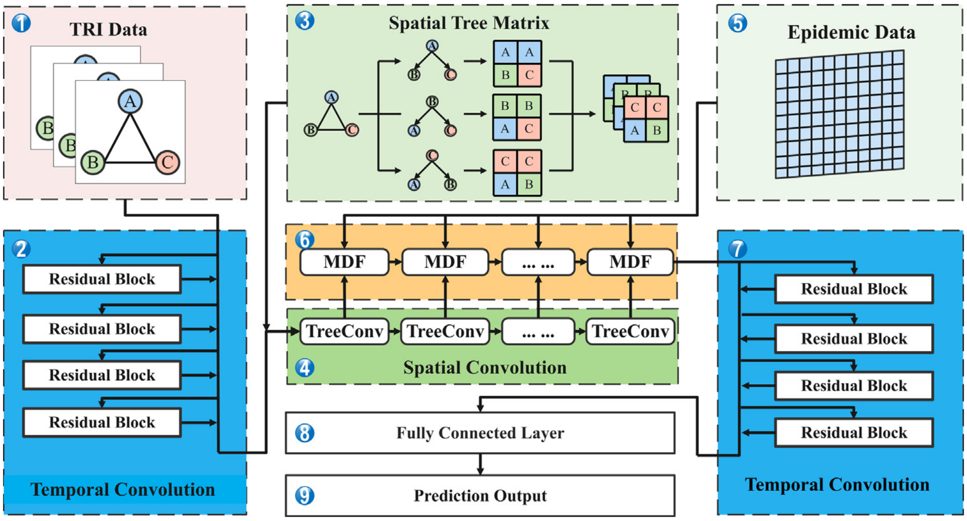


Fig. 7. The model structure of DeepTree-TRI.

4. Model design

The structure of the DeepTree-TRI model proposed in this study is shown in Fig. 7. The input data of the DeepTree-TRI includes the TRI data of cities, the spatial tree matrix representing the spatial distribution of cities and the COVID-19 epidemic data. Firstly, the historical TRI data of all cities are processed in chronological order of nodes and input to the first temporal convolution module. The function of the first temporal convolution module is to perform preliminary scale compression and dimension transformation on the TRI data so that the dimensions of the TRI data can meet the input requirements of the spatial convolution module. Secondly, the spatial features such as the positional relationship and distance relationship among city nodes are stored in the spatial tree matrix through the transformation process of the tree matrix. The temporal TRI data and the spatial tree matrix passed through the first temporal convolution module are jointly input to the spatial convolution module. The spatial convolution module uses the tree convolution process to process the spatial tree matrix to capture the deep spatial correlation features among nodes, and fuses the captured spatial correlation features with the temporal TRI data in the form of matrix multiplication. Finally, the MDF module performs multi-scale fusion of the epidemic data and the output of the spatial convolution module in the form of convolution. At this stage, the original TRI data already possesses the deep spatial correlation features of nodes and epidemic data features. The second temporal convolutional module builds with multiple layers of residual blocks captures temporal dependent features on the fused features from the preceding process. The final data features are mapped to the data sample space by the fully connected layer to obtain the predicted output of the model.

4.1. Spatial convolution module

The function of the spatial convolution layer is to capture the spatial correlation of the TRI data among the city nodes. Each plane tree matrix that makes up the spatial tree matrix is independently performed in the tree convolution process in the specific

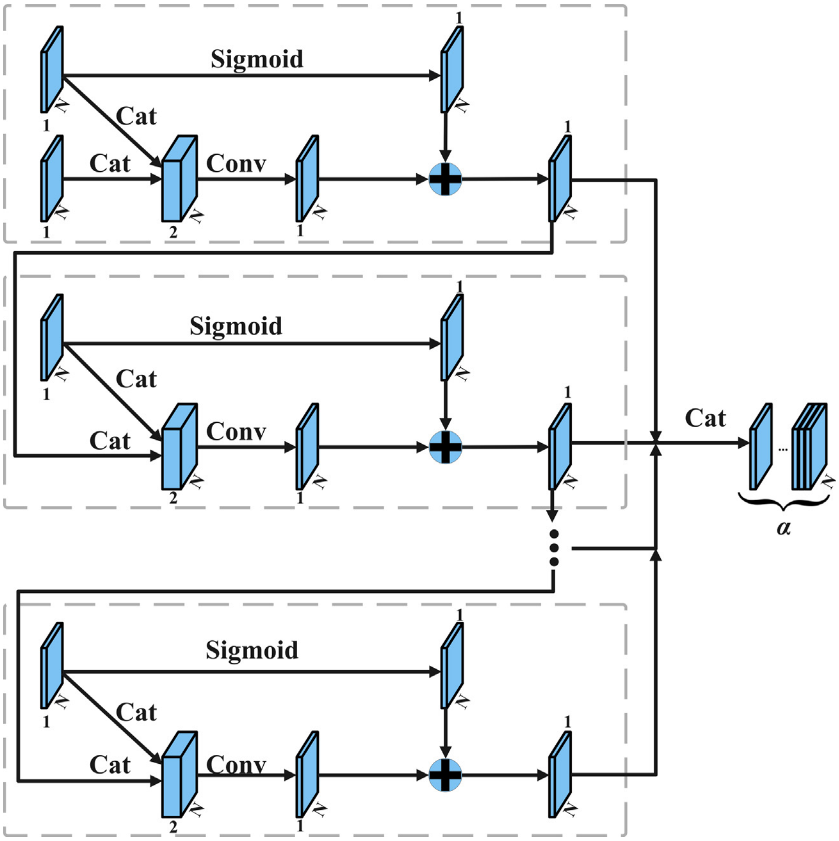


Fig. 8. The process of tree convolution.

implementation process of the spatial convolution module. The tree convolution process can convert the original features of the tree structure into preliminary features, and finally into high-level features. The convolution kernel of tree convolution always maintains the size of [2 × 1] and translates from bottom to top to capture the directional and hierarchical features among nodes. The structure of tree convolution in the spatial convolution module is shown in Fig. 8. There are multiple basic units in tree convolution to realize the layer-by-layer convolution process. Firstly, the nodes at the bottommost level and the nodes at the penultimate level in the plane tree matrix are concatenated and input into the CNN. Secondly, the original features of the nodes of the current layer are weighted with the input from the CNN to prevent the information loss of the nodes of the current layer [40]. The model repeats the above process until the nodes of all layers are fused, which are used as high-level features of a plane tree matrix. The final output of tree convolution is concatenated from the high-level features of all plane tree matrices that make up the spatial tree matrix.

4.1.1. Capturing directional features

The combined use of tree matrix and tree convolution can effectively capture the directional features between nodes [41], as shown in Fig. 9. Each column of the plane tree matrix stores the connected path from the root node to the leaf node. The nodes of the same color in Fig. 9 represent the real connectivity of city nodes. The red matrix in tree convolution represents the convolution kernel of tree convolution. The connected path of each node has an independent convolution process, which does not cross the connected paths of other nodes. The connectivity features of the connected paths of each node are transformed into high-level features. Finally, the tree convolution concatenates the high-level features of all connected paths into the directional features of the root node.

4.1.2. Capturing hierarchical features

The Fig. 10 shows the capture process of hierarchical features in tree convolution. Because the tree structures are generated by breadth-first traversal of graph, tree structures and plane tree matrices can store hierarchical relationships among nodes [42]. The hierarchical relationships are interpreted as direct or indirect relationships connected to the current node. The root node is always stored in the first row of the plane tree matrix. The nodes directly connected to the root node are stored in the second row of the plane tree matrix. Other nodes indirectly connected to the root node are stored in different rows in the plane tree matrix according to the number of indirect connections. The tree convolution obtains the hierarchical relationship of each row of the plane tree matrix through the translation calculation of the convolution kernel according to the principle of row priority. The node layer with a larger

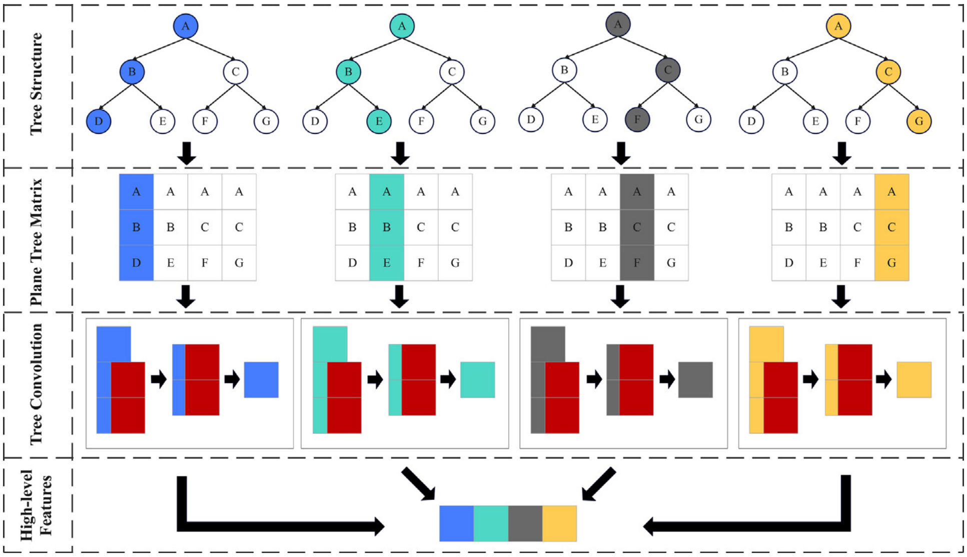


Fig. 9. The capture process of directional features in tree convolution. . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

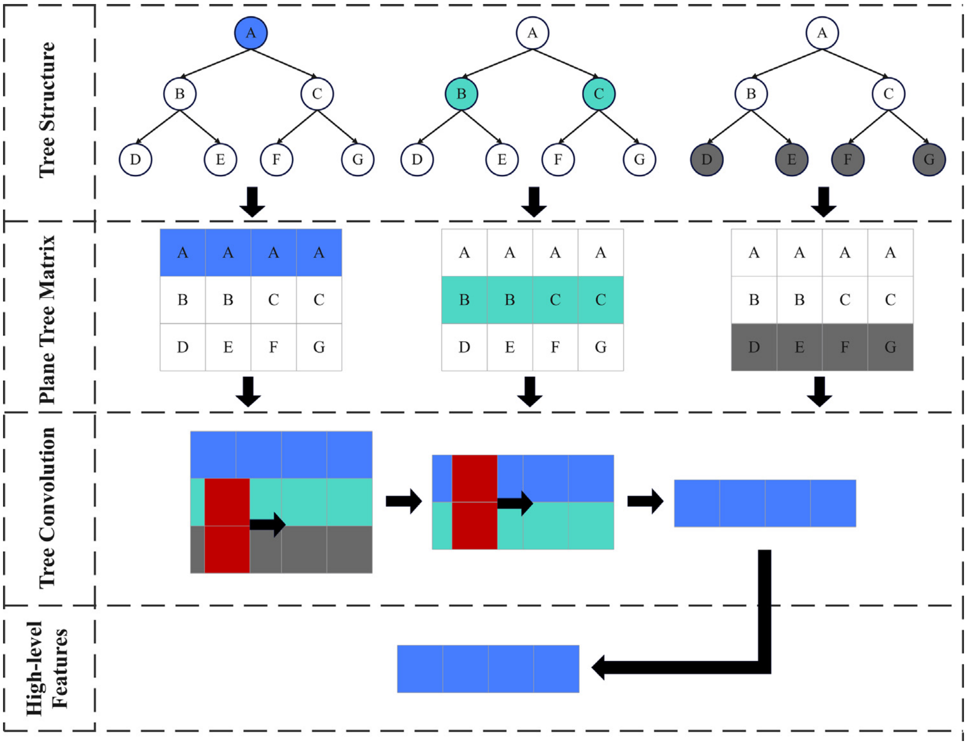


Fig. 10. The capture process of hierarchical features in tree convolution.

number of indirect connections is firstly subjected to the convolution process, which means that the indirectly connected nodes are performed more times of the convolution process. Each convolution process must have a certain degree of information loss. Therefore, the root node and directly connected nodes can retain more directly related information because they are performed less convolution process. The above process enables the high-level features generated by tree convolution to contain different degrees of hierarchical features among nodes.

$$y = X_N^t \times TreeConv\left(T\right) \tag{7}$$

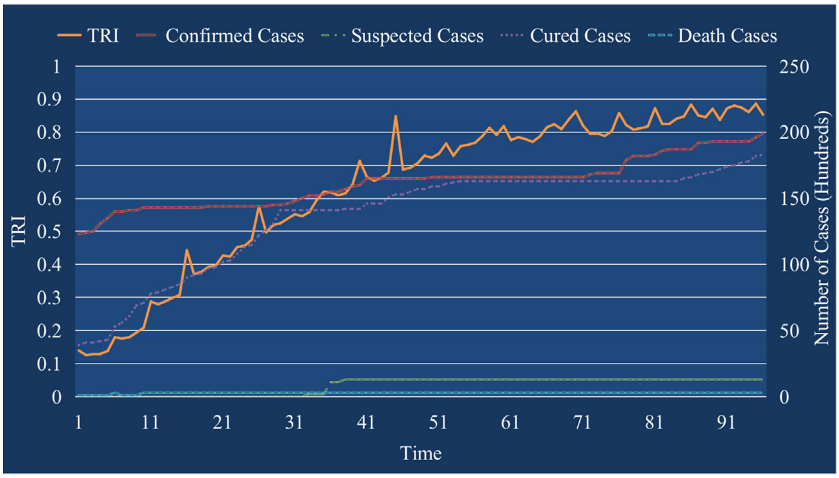


Fig. 11. The curve change between TRI data and COVID-19 epidemic data.

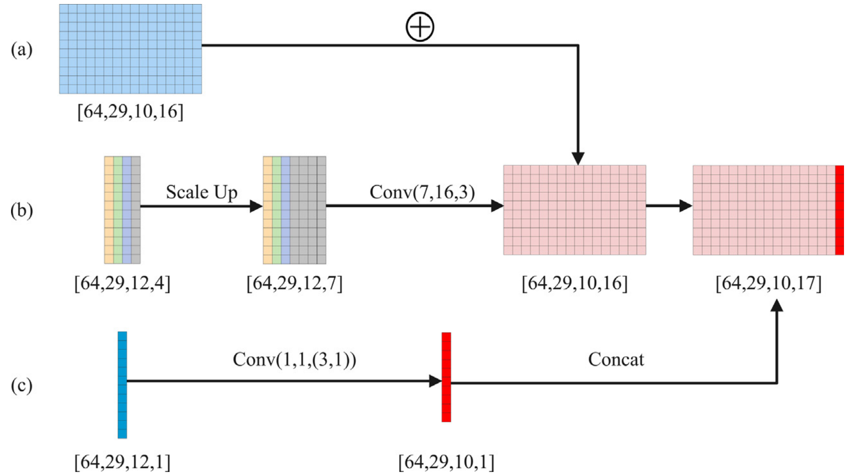


Fig. 12. The fusion process of MDF module. The (a) represents the TRI data features after processing by the spatial convolution module. The (b) represents the matrix feature of COVID-19 epidemic data for each city. The (c) represents the matrix feature of national cured cases.

The high-level features of the spatial tree matrix are composed of high-level features generated by tree convolution for each plane tree matrix. The high-level features of the spatial tree matrix contain spatial correlations among city nodes, which need to be fused with TRI data. The fusion process is shown in Eq. (7), where the T represents the original data of the spatial tree matrix and the *TreeConv* represents the tree convolution process. The matrix multiplication calculation is performed on the high-level features of the convolution of the TRI data X_N^t and the spatial tree, and the purpose is to give the TRI data the spatial correlation feature among nodes on the premise that the TRI data dimension does not change. The data dimension has not changed to avoid the problem of major information loss [43].

4.2. Matrix data fusion module

In order to capture the correlation between TRI data and COVID-19 epidemic data, this study designs a matrix data fusion module for COVID-19 epidemic data fusion. The COVID-19 epidemic data includes confirmed cases, suspected cases, cured cases, and death cases in individual cities and the entire country. Different types of data correlate differently with TRI data, as shown in Fig. 11. The Fig. 11 shows the synchronized changes of TRI and COVID-19 epidemic data in a city in China during the epidemic. Different types of data all share some degree of similarity with TRI, and the cured cases have the highest correlation with TRI. The MDF module sets different ratios for the four types of COVID-19 epidemic data in the fusion matrix according to the correlation between different data and TRI data. The ratio of the four types of data is $r_c : r_s : r_u : r_d$ = 1:1:1:4 (the cured cases occupy the largest proportion). The Fig. 12 shows the specific process of MDF. The MDF scales up the original epidemic data matrix to meet the data scale requirements. The transformed data features are passed through a CNN with 3 × 3 convolution kernel to obtain the deep features of the matrix. The

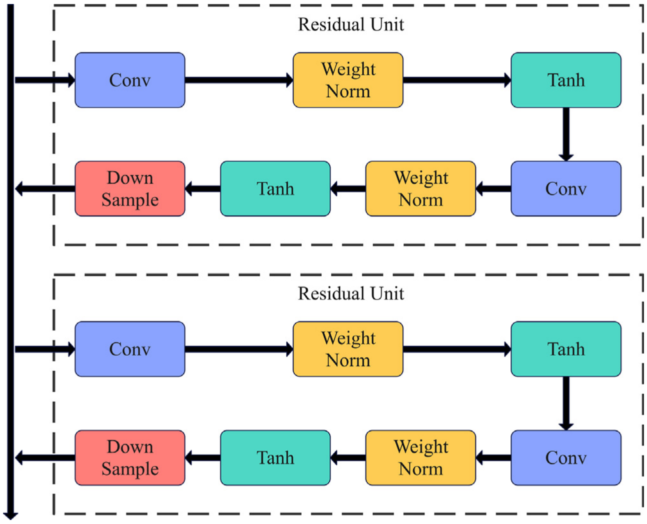


Fig. 13. The structure of the residual unit.

TRI data features processed by the spatial convolution module are weighted and fused with the COVID-19 epidemic data features. The epidemic data of the whole country is a synthesis of the epidemic data of all cities, which is related to the TRI data at the macro level. Therefore, the MDF splices the national cured case data in the final matrix feature.

4.3. Temporal convolution module

In order to fully use the historical information of the TRI data and obtain the temporal dependence from the long-term changes of the data, this study adopts a temporal convolutional module composed of a multi-layer residual structure to replace the traditional RNN to solve the problem of gradient disappearance in model training process and improve the effectiveness of the training process [44]. The structure of the residual unit is shown in Fig. 13. Each residual unit has two convolution processes that capture the features of the data. The *Weight Norm* represents the process of weight normalization [45], which can speed up the convergence of the model. The residual unit increases the activation function *Tanh* after each convolution process to optimize the output range of feature weights.

The residual structure of each layer is specifically composed of causal convolution and dilated convolution. The role of causal convolution is to enable the model to trace to longer-term historical information when capturing temporal dependent features. The calculation process of causal convolution is shown in Eq. (8), where the f_i represents the sequence value of the filter and the K represents the convolution kernel size. The causal convolution can mask future time series data and only consider historical time series data. This design enables the hidden layer to trace longer-term historical data in limited storage space. However, the ability of causal convolution to capture historical data features is also limited. This is because long-term historical information will generate a larger number of hidden layer parameters, which will cause the problem of gradient explosion during training [46]. Therefore, the dilated convolution is needed to make up for the defect of causal convolution. The role of dilated convolution is to expand the receptive field range in the convolution process and its calculation process is shown in Eq. (9). The dilated convolution uses the dilated factor d to control the range of the receptive field and the d is gradually increased according to the exponent of 2. Therefore, even if the model uses a multi-layer CNN, the model will not have the problem of gradient explosion caused by the expansion of the number of parameters.

$$y = \sum_{i=0}^{K} f_i \cdot x_{i-K+i} \tag{8}$$

$$y_t = \sum_{i=0}^{K-1} f_i \cdot x_{t-i\cdot d} \tag{9}$$

5. Experiment

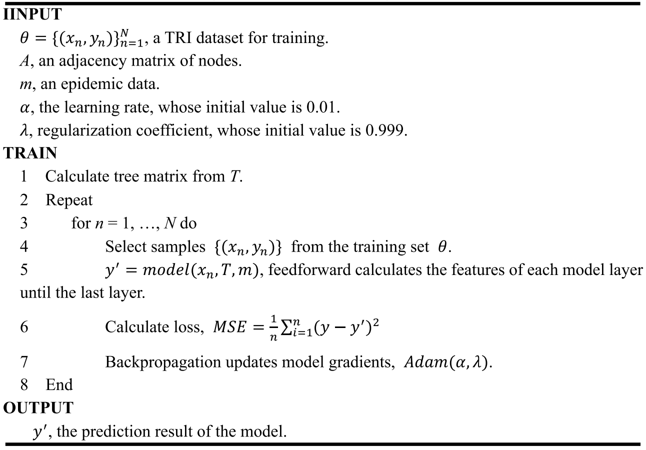
5.1. Data preparation

This experiment uses the TRI data of 29 core cities in China, whose time range is from February 10, 2020 to June 30, 2020. The data comes from the Gaia program of Didi. The TRI data are obtained by fitting, cross-validation and weighting to urban traffic trajectory data, traffic congestion data and commuter data. The COVID-19 epidemic data comes from the National Health Commission of China, which includes the number of national cured cases and the number of confirmed cases, suspected cases, cured cases and death cases in 29 cities during the same period as the TRI data.

5.2. Experiment setup

5.2.1. Model training settings

Algorithm 1. Model Training Process.



```
IINPUT
  $\theta = \{(x_n, y_n)\}_{n=1}^{N}$, a TRI dataset for training.
  $A$, an adjacency matrix of nodes.
  $m$, an epidemic data.
  $\alpha$, the learning rate, whose initial value is 0.01.
  $\lambda$, regularization coefficient, whose initial value is 0.999.
TRAIN
  Calculate tree matrix from $T$.
  Repeat
    for $n$ = 1, …, $N$ do
        Select samples $\{(x_n, y_n)\}$ from the training set $\theta$.
        $y' = model(x_n, T, m)$, feedforward calculates the features of each model layer until the last layer.
        Calculate loss, $MSE = \frac{1}{n}\sum_{i=1}^{n}(y - y')^2$
        Backpropagation updates model gradients, $Adam(\alpha, \lambda)$.
  End
OUTPUT
  $y'$, the prediction result of the model.
```

The Algorithm 1 shows the model training process in this study. This experiment utilizes TRI data from 12 historical time steps to predict data from 3 future time steps. The loss function Mean squared error (MSE) is used in the training process to calculate the gradient of the model. The optimization function used during training is the *Adam*, which uses the learning rate of 0.01 and the regularization coefficient of 0.999. The normalization method of TRI data is the Z-Score and its calculation process is shown in Eq. (10). The $\overline{x}$ represents the mean of the data sample and the s represents the standard deviation of the data sample. The Z-Score can scale the ratio of TRI data to a specific data range, which enables the TRI data to be unaffected by the magnitude of data in the comparative analysis of multiple evaluation indicators.

$$z_i = \frac{x_i - \overline{x}}{s} \tag{10}$$

5.2.2. Evaluation indicator settings

In this experiment, the Mean Square Error (MAE), Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) are used as the evaluation indicators of the model, as shown in Eqs. (11), (12) and (13). The n represents the total amount of data. The $\hat{y}_i$ represents the true value and the y_i represents the predicted value. These three indicators are usually used as metrics for time series forecasting. Both MAE and RMSE are used to measure the deviation between the predicted value and the true value. They are sensitive to extreme values, which can highlight the error values with greater influence. The MAPE is used to evaluate the offset of the predicted value relative to the true value, and it is not susceptible to extreme values. The smaller the error represented by the three indicators, the better the effect of the model.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \tag{11}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2} \tag{12}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{13}$$

5.2.3. Baselines settings

The experiments in this study compare the DeepTree-TRI with six baseline models. The basic input of all models is TRI and epidemic time series data, among which the GCN, T-GCN, ASTGCN and DeepMeta-TRI additionally use the relevant location relationship data of nodes (location and distance among nodes). The experimental results of all models are the mean of 20 evaluation results. The experimental training parameter configurations of all models are consistent, as described above. The detailed description of the six baselines follows:

- LSTM and GRU: the LSTM [47] is a special RNN with long–short term memory mechanism, which realizes long-term sequence prediction task by controlling the transmission state by gating state. The GRU [48] has the same logical structure as LSTM and its structure conforms to the design idea of RNN. However, it can solve the gradient vanishing problem of LSTM by choosing the forgetting mechanism and also solve the problem of gradient explosion caused by too many parameters in the LSTM. The time step of LSTM and GRU is 12, and the feature size of each time step is 12, and the hidden cells are 64, and the output feature size is 3.

Table 1
The statistical result of indicators.

Model	MAE			RMSE			MAPE		
	1	2	3	1	2	3	1	2	3
LSTM	4.87	4.71	4.63	7.21	6.88	6.68	9.81	8.80	8.06
GRU	4.76	4.70	4.61	7.14	6.85	6.62	9.73	8.69	7.98
GCN	4.62	4.55	4.43	6.86	6.66	6.46	9.38	8.51	7.76
T-GCN	4.51	4.54	4.32	6.80	6.77	6.40	9.21	8.51	7.58
ASTGCN	3.72	3.68	3.66	5.41	5.36	5.34	7.61	7.62	7.46
DeepMeta-TRI	2.68	2.61	2.62	3.44	3.35	3.39	4.37	4.16	4.09
DeepTree-TRI	**2.01**	**2.14**	**2.08**	**2.76**	**2.84**	**2.75**	**3.12**	**3.36**	**3.20**

- GCN: the GCN [49] uses an adjacency matrix to represent the spatial correlation information between nodes. The convolution process of GCN uses the Fourier operator to transform the Laplacian matrix, which involves the transformation of matrix data from the frequency domain to the spectral domain. We use an improved GCN as shown in Eq. (14), which introduces a degree matrix D to solve the self-propagation problem of GCN and normalizes the adjacency matrix to reduce the influence of the number of neighboring points on the current point, where the $\widetilde{w}$ is the result of the subtraction of adjacency matrix and D. The $\boldsymbol{w}'$ is the parameter weight matrix.

$$y = f\left(D^{-\frac{1}{2}}\widetilde{w}D^{-\frac{1}{2}}xw'\right) \tag{14}$$

- T-GCN: the T-GCN [34] is a temporal graph convolutional network for traffic prediction. It combines the GCN with the GRU to capture the temporal and spatial dependencies of graph topology. The optimization method of T-GCN using GCN is shown in Eq. (13), and the parameter configuration of GRU is as described above. The process of fusion of T-GCN and epidemic data occurs before GRU calculation, and matrix multiplication is used to achieve feature fusion.
- ASTGCN and DeepMeta-TRI: the ASTGCN [36] is a spatial–temporal graph convolutional network for traffic prediction. It builds a spatial–temporal attention mechanism for data dynamic spatial–temporal correlation processing on the basis of GCN and captures the effects of multi-layer time steps of time series data. The DeepMeta-TRI [27] is a deep spatial–temporal meta-learning model for TRI data prediction, which integrates meta-learning and external auxiliary information to predict the temporal changes of TRI data. Both ASTGCN and DeepMeta-TRI use GCN as the capture module of spatial features. The optimization method is shown in Eq. (13), and we use the matrix multiplication method after GCN to fuse the epidemic data. A special three-layer CNN and a fully-connected network to build the capture module of temporal features. The convolution kernel size of each layer is 3 × 3 and the input size of each layer of CNN is 128 × 128, 64 × 64 and 32 × 32 respectively. The parameter configuration of other special structures is consistent with the original research literature.

5.3. Quantitative statistics and qualitative analysis

5.3.1. Quantitative statistical results

In this experiment, the prediction indicators for the next three time steps are counted for the DeepTree-TRI and each baseline model, as shown in Table 1. The prediction effect of the RNN with LSTM and GRU is the worst compared to other models. The prediction effect of GCN is better than that of LSTM and GRU. The T-GCN and ASTGCN based on GCN add the consideration of spatial–temporal features in the model design, so they further improve the prediction effect. The DeepMeta-TRI is a model specially designed for TRI data, which has a significant improvement over the aforementioned baseline models. Compared with the DeepMeta-TRI, the DeepTree-TRI has an average improvement of 21%, 18%, and 23% in MAE, RMSE and MAPE indicators, respectively.

5.3.2. Qualitative analysis

The LSTM and GRU can effectively capture the temporal dependent features of TRI data, but they lack the process of capturing the spatial correlation features among city nodes [50], as shown in Figs. 14 and 15. We can know from the results of Fig. 14 that the prediction results of LSTM and GRU can maintain a certain degree of change trend with the TRI data. However, the LSTM and the GRU cannot achieve prediction of local peaks in TRI data. The TRI data has a continuous fluctuation interval, which has a great impact on the prediction ability of LSTM and GRU. The Fig. 14 shows the error between the predicted values and true values at different time steps, which also shows the underfitting of LSTM and GRU to the local peaks of the TRI data. The prediction results of LSTM and GRU show that the TRI data comes from various city nodes, and the correlation features among nodes are considered in the modeling process to effectively improve the prediction effect.

The prediction results of GCN shown in Figs. 14 and 15 show the importance of spatial correlation features among nodes [51]. The prediction results in Fig. 14 have been able to fit some local peaks, and the prediction error range of the GCN in Fig. 15 has been significantly narrowed compared to the prediction results of LSTM and GRU. The GCN uses the adjacency matrix to store the spatial distribution relationship among city nodes and captures the mutual influence of TRI data among nodes. However, the GCN lacks the effect of temporal dependence of data changes on the nodes themselves. Therefore, the pure GCN model also has a certain degree of underfitting, which is also reflected in Fig. 15.

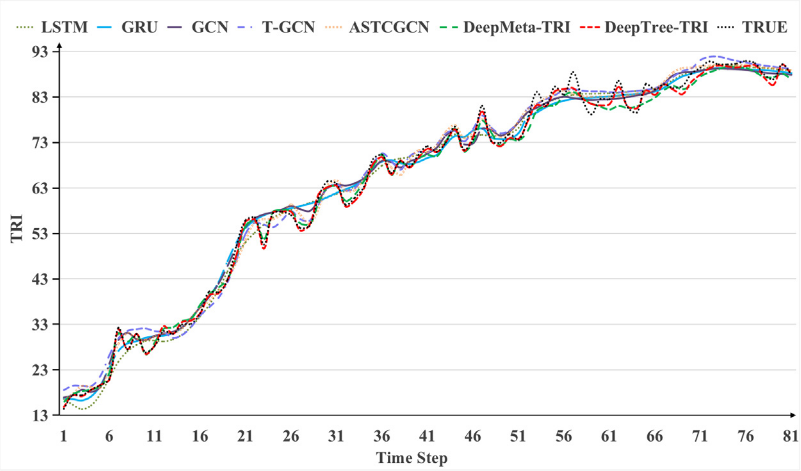


Fig. 14. The comparison of the prediction effect of DeepTree-TRI and other baselines for continuous time steps. The abscissa represents the time step, that is, the number of consecutive days of data. The ordinate represents the value of TRI data.

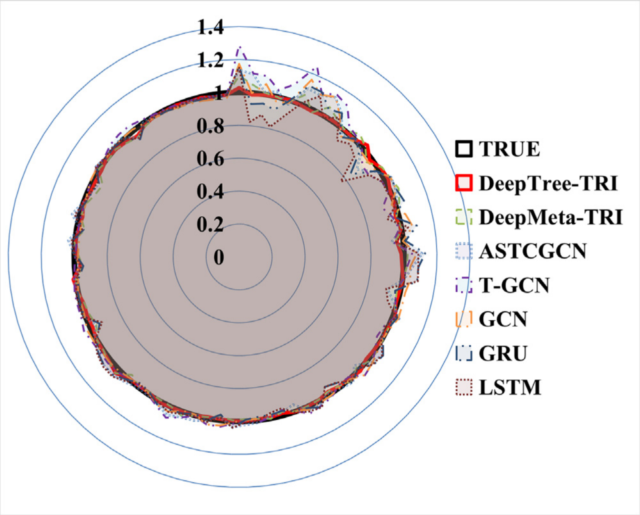


Fig. 15. The comparison of the prediction effect of DeepTree-TRI and other baselines in the radar chart.

At the same time, Figs. 14 and 15 show the prediction effects of the T-GCN, ASTGCN and DeepMeta-TRI with using the GCN as spatial capture module. The three types of baseline models use the GCN as the capture module of spatial correlation features among city nodes and add temporal feature calculation module to capture the temporal dependent features of TRI data. The consideration of spatial–temporal features makes the prediction effect of these three types of baseline models significantly better than the GCN which only considers spatial features and traditional RNN (LSTM and GRU) which only considers temporal features. Because their research is based on GCN, they cannot overcome the inherent problem of the GCN [52], which is its lack of directional features and hierarchical features among city nodes. The DeepTree-TRI proposed in this study stores the spatial distribution relationship among city nodes in a tree structure, and stores the directional and hierarchical relationships among nodes through the spatial tree matrix. The tree convolution process in DeepTree-TRI further captures the directional and hierarchical features. The temporal convolution module composed of residual structure overcomes the gradient explosion and gradient disappearance problems of traditional RNN, which enables DeepTree-TRI to obtain better prediction results compared to other baseline models within the same model training time. Meanwhile, the fusion of MDF module for COVID-19 epidemic data further reduces the prediction error of TRI data.

5.3.3. Ablation experiment

In order to verify the effect of each module of DeepTree-TRI, this experiment performed ablation experiments on each module of DeepTree-TRI, and the results are shown in Table 2. The TCM stands for temporal convolution module. The SCM stands for spatial

Table 2
The ablation experiment results of DeepTree-TRI.

Module	MAE			RMSE			MAPE		
	1	2	3	1	2	3	1	2	3
TCM	3.55	3.50	3.15	4.59	4.53	4.13	5.34	4.49	4.27
TCM+MDF	3.44	3.23	3.21	4.32	4.13	3.11	5.84	5.39	5.20
SCM	3.01	2.99	2.99	3.78	3.74	3.72	5.16	5.07	4.96
SCM+MDF	2.82	2.67	2.65	3.57	3.40	3.39	4.48	4.15	3.99
SCM+TCM	2.54	2.51	2.52	3.34	3.26	3.25	4.29	4.09	4.10
SCM+MDF+TCM	**2.01**	**2.14**	**2.08**	**2.76**	**2.84**	**2.75**	**3.12**	**3.36**	**3.20**

convolution module. The MDF stands for matrix fusion module. The SCM, MDF and TCM together constitute the whole structure of DeepTree-TRI. The results of the ablation experiments show that the MDF used for the fusion of COVID-19 epidemic data has a certain degree of improvement in both the spatial convolution module and the temporal convolution model. The combination of spatial convolution module and temporal convolution module enables the model to fully capture the spatial–temporal features of TRI data, and its effect is better than that of a single module. The results of the ablation experiments demonstrate that the spatial–temporal features of TRI data are the most important core influencing factors and their role is greater than that of the COVID-19 epidemic data as the auxiliary factors. The combination of the three modules can make DeepTree-TRI show the best prediction effect.

6. Conclusion

At present, the global COVID-19 epidemic prevention and control situation is still grim. Evaluating and predicting TRI changes in cities affected by the COVID-19 epidemic is the research goal of this study. The TRI data prediction can help relevant government departments formulate effective transportation industry policies and further promote urban economic recovery and development. This study proposes a deep spatial–temporal model based on true structure (DeepTree-TRI) for TRI data prediction. In order to overcome the inability to calculate the directional and hierarchical features of nodes using GCN in existing research, the DeepTree-TRI uses the tree structure to represent urban spatial distribution information and uses the tree convolution to capture the spatial correlation features of urban nodes. The temporal convolution module consisting of a multi-layer residual structure is able to overcome the problems of exploding and vanishing gradients while capturing the temporal dependent features of TRI data. The matrix fusion module of DeepTree-TRI integrates the COVID-19 epidemic data of each city with the TRI data to further improve the prediction effect of the model. In this study, various baseline models are set up to verify the prediction effect of DeepTree-TRI. The experimental results show that DeepTree-TRI has a good effect in multiple regression prediction indicators. Future research of this study will focus on the field of TRI data analysis and predicting on a global scale. The research focuses on the fusion calculation of large-scale TRI data and COVID-19 epidemic data.

CRediT authorship contribution statement

Zhiqiang Lv: Conceptualization, Methodology, Software, Investigation, Formal Analysis, Writing – original draft. **Xiaotong Wang:** Data Curation, Writing – original draft. **Zesheng Cheng:** Conceptualization, Funding acquisition, Resources, Supervision, Writing. **Jianbo Li:** Resources, Supervision. **Haoran Li:** Software, Validation. **Zhihao Xu:** Visualization, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the findings of this study are available in GitHub. The processed dataset and code are available at https://github.com/qdu318/DeepTree-TRI.git.

Acknowledgments

This research was supported in part by National Key Research and Development Plan Key Special Projects under Grant No. 2018YFB2100303, Shandong Province colleges and universities youth innovation technology plan innovation team project under Grant No. 2020KJN011, Program for Innovative Postdoctoral Talents in Shandong Province under Grant No. 40618030001, National Natural Science Foundation of China under Grant No. 61802216, and Postdoctoral Science Foundation of China under Grant No. 2018M642613.

References

[1] K.D. Alemdar, Kaya. Ö, A. Canale, et al., Evaluation of air quality index by spatial analysis depending on vehicle traffic during the COVID-19 outbreak in Turkey, Energies 14 (18) (2021) 5729, http://dx.doi.org/10.3390/en14185729.

[2] S. Saurabh, S. Madria, A. Mondal, et al., An analytical model for information gathering and propagation in social networks using random graphs, Data Knowl. Eng. 129 (2020) 101852, http://dx.doi.org/10.1016/j.datak.2020.101852.

[3] Z. Lv, J. Li, C. Dong, et al., Deep learning in the COVID-19 epidemic: A deep model for urban traffic revitalization index, Data Knowl. Eng. 135 (2021) 101912, http://dx.doi.org/10.1016/j.datak.2021.101912.

[4] J. Lu, A. Lin, C. Jiang, et al., Influence of transportation network on transmission heterogeneity of COVID-19 in China, Transp. Res. C 129 (2021) 103231, http://dx.doi.org/10.1016/j.trc.2021.103231.

[5] P. Zhao, Y. Gao, Public transit travel choice in the post COVID-19 pandemic era: An application of the extended theory of planned behavior, Travel Behav. Soc. 28 (2022) 181–195, http://dx.doi.org/10.1016/j.tbs.2022.04.002.

[6] Y. Zhang, L. Yang, X. Wang, Analysis and calculating of comprehensive urban vitality index by multi-source temporal-spatial big data and EW-TOPSIS, in: 2021 IEEE International Conference on Data Science and Computer Application, IEEE, 2021, pp. 196–201, http://dx.doi.org/10.1109/ICDSCA53499.2021.9650148.

[7] J. Chen, Y. Chen, The study on the influence of highway transportation on regional economic development, in: 2017 5th International Education, Economics, Social Science, Arts, Sports and Management Engineering Conference, Atlantis Press, 2018, pp. 360–364, http://dx.doi.org/10.2991/ieesasm-17.2018.74.

[8] M. Kurth, W. Kozlowski, A. Ganin, et al., Lack of resilience in transportation networks: Economic implications, Transp. Res. D 86 (2020) 102419, http://dx.doi.org/10.1016/j.trd.2020.102419.

[9] Y. Li, Y. Yang, G. Luo, et al., The economic recovery from traffic restriction policies during the COVID-19 through the perspective of regional differences and sustainable development: Based on human mobility data in China, Sustainability 14 (11) (2022) 6453, http://dx.doi.org/10.3390/su14116453.

[10] A. Sharifi, A.R. Khavarian-Garmsir, R.K.R. Kummitha, Contributions of smart city solutions and technologies to resilience against the COVID-19 pandemic: A literature review, Sustainability 13 (14) (2021) 8018, http://dx.doi.org/10.3390/su13148018.

[11] A. Abdi, C. Amrit, A review of travel and arrival-time prediction methods on road networks: classification, challenges and opportunities, PeerJ Comput. Sci. 7 (2021) e689, http://dx.doi.org/10.7717/peerj-cs.689.

[12] H. Xu, C. Jiang, Deep belief network-based support vector regression method for traffic flow forecasting, Neural Comput. Appl. 32 (7) (2020) 2027–2036, http://dx.doi.org/10.1007/s00521-019-04339-x.

[13] L. Cai, Y. Yu, S. Zhang, et al., A sample-rebalanced outlier-rejected k-nearest neighbor regression model for short-term traffic flow forecasting, IEEE Access 8 (2020) 22686–22696, http://dx.doi.org/10.1109/ACCESS.2020.2970250.

[14] L. Han, Y.S. Huang, Short-term traffic flow prediction of road network based on deep learning, IET Intell. Transp. Syst. 14 (6) (2020) 495–503, http://dx.doi.org/10.1049/iet-its.2019.0133.

[15] Y. Zhang, Short-term traffic flow prediction methods: A survey, J. Phys. Conf. Ser. 1486 (5) (2020) 052018, http://dx.doi.org/10.1088/1742-6596/1486/5/052018, IOP Publishing.

[16] K. Guo, Y. Hu, Z. Qian, et al., Optimized graph convolution recurrent neural network for traffic prediction, IEEE Trans. Intell. Transp. Syst. 22 (2) (2020) 1138–1149, http://dx.doi.org/10.1109/TITS.2019.2963722.

[17] S. Wu, Spatiotemporal dynamic forecasting and analysis of regional traffic flow in urban road networks using deep learning convolutional neural network, IEEE Trans. Intell. Transp. Syst. 23 (2) (2021) 1607–1615, http://dx.doi.org/10.1109/TITS.2021.3098461.

[18] M. Montanino, J. Monteil, V. Punzo, From homogeneous to heterogeneous traffic flows: Lp string stability under uncertain model parameters, Transp. Res. B 146 (2021) 136–154, http://dx.doi.org/10.1016/j.trb.2021.01.009.

[19] H. Yuan, G. Li, A survey of traffic prediction: from spatio-temporal data to intelligent transportation, Data Sci. Eng. 6 (1) (2021) 63–85, http://dx.doi.org/10.1007/s41019-020-00151-z.

[20] W. Li, X. Wang, Y. Zhang, et al., Traffic flow prediction over muti-sensor data correlation with graph convolution network, Neurocomputing 427 (2021) 50–63, http://dx.doi.org/10.1016/j.neucom.2020.11.032.

[21] Y. Zhao, J. Qi, Q. Liu, et al., Wgcn: graph convolutional networks with weighted structural features, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 624–633, http://dx.doi.org/10.1145/3404835.3462834.

[22] Y. Huang, X. Song, S. Zhang, et al., Transfer learning in traffic prediction with graph neural networks, in: 2021 IEEE International Intelligent Transportation Systems Conference, IEEE, 2021, pp. 3732–3737, http://dx.doi.org/10.1109/ITSC48978.2021.9564890.

[23] H. Peng, B. Du, M. Liu, et al., Dynamic graph convolutional network for long-term traffic flow prediction with reinforcement learning, Inform. Sci. 578 (2021) 401–416, http://dx.doi.org/10.1016/j.ins.2021.07.007.

[24] Y. Pang, T. Kashiyama, Y. Sekimoto, Development of a reinforcement learning based agent model and people flow data to Mega Metropolitan Area, in: 2021 IEEE International Conference on Big Data, IEEE, 2021, pp. 3755–3759, http://dx.doi.org/10.1109/BigData52589.2021.9671399.

[25] Y. Chen, Y. Kang, Y. Chen, et al., Probabilistic forecasting with temporal convolutional neural network, Neurocomputing 399 (2020) 491–501, http://dx.doi.org/10.1016/j.neucom.2020.03.011.

[26] Z. Zhao, Q. Li, Z. Zhang, et al., Combining a parallel 2D CNN with a self-attention dilated residual network for CTC-based discrete speech emotion recognition, Neural Netw. 141 (2021) 52–60, http://dx.doi.org/10.1016/j.neunet.2021.03.013.

[27] Y. Wang, Z. Lv, Z. Sheng, et al., A deep spatio-temporal meta-learning model for urban traffic revitalization index prediction in the COVID-19 pandemic, Adv. Eng. Inform. 53 (2022) 101678, http://dx.doi.org/10.1016/j.aei.2022.101678.

[28] Z. Zhang, M. Liu, W. Xu, Spatial-temporal multi-head attention networks for traffic flow forecasting, in: The 5th International Conference on Computer Science and Application Engineering, 2021, pp. 1–7, http://dx.doi.org/10.1145/3487075.3487102.

[29] K.H.N. Bui, J. Cho, H. Yi, Spatial–temporal graph neural network for traffic forecasting: An overview and open research issues, Appl. Intell. (2021) 1–12, http://dx.doi.org/10.1007/s10489-021-02587-w.

[30] Y. Zhang, T. Cheng, Y. Ren, et al., A novel residual graph convolution deep learning model for short-term network-based traffic forecasting, Int. J. Geogr. Inf. Sci. 34 (5) (2020) 969–995, http://dx.doi.org/10.1080/13658816.2019.1697879.

[31] B. Sun, D. Zhao, X. Shi, et al., Modeling global spatial–temporal graph attention network for traffic prediction, IEEE Access 9 (2021) 8581–8594, http://dx.doi.org/10.1109/ACCESS.2021.3049556.

[32] N. Hu, D. Zhang, K. Xie, et al., Graph learning-based spatial–temporal graph convolutional neural networks for traffic forecasting, Connect. Sci. 34 (1) (2022) 429–448, http://dx.doi.org/10.1080/09540091.2021.2006607.

[33] W. Fang, Y. Chen, Q. Xue, Survey on research of RNN-based spatio-temporal sequence prediction algorithms, J. Big Data 3 (3) (2021) 97, http://dx.doi.org/10.32604/jbd.2021.016993.

[34] L. Zhao, Y. Song, C. Zhang, et al., T-gcn: A temporal graph convolutional network for traffic prediction, IEEE Trans. Intell. Transp. Syst. 21 (9) (2019) 3848–3858, http://dx.doi.org/10.1109/TITS.2019.2935152.

[35] B. Yu, H. Yin, Z. Zhu, Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting, 2017, pp. 1–7, http://dx.doi.org/10.24963/ijcai.2018/505, arXiv preprint arXiv:1709.04875.

[36] S. Guo, Y. Lin, N. Feng, et al., Attention based spatial–temporal graph convolutional networks for traffic flow forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 922–929, http://dx.doi.org/10.1609/aaai.v33i01.3301922, (01).

[37] C. Chen, F. Cai, X. Hu, et al., HHGN: A hierarchical reasoning-based heterogeneous graph neural network for fact verification, Inf. Process. Manage. 58 (5) (2021) 102659, http://dx.doi.org/10.1016/j.ipm.2021.102659.

[38] J. Tang, J. Liang, F. Liu, et al., Multi-community passenger demand prediction at region level based on spatio-temporal graph convolutional network, Transp. Res. C 124 (2021) 102951, http://dx.doi.org/10.1016/j.trc.2020.102951.

[39] T. Chen, Y. Zhang, X. Qian, et al., A knowledge graph-based method for epidemic contact tracing in public transportation, Transp. Res. C 137 (2022) 103587, http://dx.doi.org/10.1016/j.trc.2022.103587.

[40] S. Qiao, N. Han, J. Huang, et al., A dynamic convolutional neural network based shared-bike demand forecasting model, ACM Trans. Intell. Syst. Technol. 12 (6) (2021) 1–24, http://dx.doi.org/10.1145/3447988.

[41] B. Wu, B. Liang, X. Zhang, Turn tree into graph: Automatic code review via simplified ast driven graph convolutional network, Knowl.-Based Syst. 252 (2022) 109450, http://dx.doi.org/10.1016/j.knosys.2022.109450.

[42] S.A. Bhavsar, V.H. Patil, A.H. Patil, Graph partitioning and visualization in graph mining: a survey, Multimedia Tools Appl. (2022) 1–42, http://dx.doi.org/10.1007/s11042-022-13017-5.

[43] S. Khan, S. Nazir, I. García-Magariño, et al., Deep learning-based urban big data fusion in smart cities: Towards traffic monitoring and flow-preserving fusion, Comput. Electr. Eng. 89 (2021) 106906, http://dx.doi.org/10.1016/j.compeleceng.2020.106906.

[44] Z. Duan, Y. Yang, W. Zhou, Multi-view spatial-temporal adaptive graph convolutional networks for traffic forecasting, in: 2021 16th International Conference on Intelligent Systems and Knowledge Engineerin, IEEE, 2021, pp. 35–41, http://dx.doi.org/10.1109/ISKE54062.2021.9755435.

[45] K. Zhang, F. He, Z. Zhang, et al., Graph attention temporal convolutional network for traffic speed forecasting on road networks, Transp. B 9 (1) (2021) 153–171, http://dx.doi.org/10.1080/21680566.2020.1822765.

[46] J. Li, Y. Lu, Z. Xu, et al., MILP: A memory improved LSTM prediction algorithm for gradient transmission time in distributed deep learning, in: ICC 2022-IEEE International Conference on Communications, IEEE, 2022, pp. 4462–4467, http://dx.doi.org/10.1109/ICC45855.2022.9839224.

[47] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[48] K. Cho, B.Van. Merriënboer, C. Gulcehre, et al., Learning phrase representations using RNN encoder–decoder for statistical machine translation, 2014, pp. 1–15, http://dx.doi.org/10.48550/arXiv.1406.1078, arXiv preprint arXiv:1406.1078.

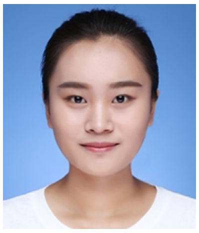
[49] M. Defferrard, X. Bresson, P. Vandergheynst, Convolutional neural networks on graphs with fast localized spectral filtering, Adv. Neural Inf. Process. Syst. 29 (2016) 1–9, http://dx.doi.org/10.48550/arXiv.1606.09375.

[50] W. Chen, C.K. Yeo, C.T. Lau, et al., Leveraging social media news to predict stock index movement using RNN-boost, Data Knowl. Eng. 118 (2018) 14–24, http://dx.doi.org/10.1016/j.datak.2018.08.003.

[51] Z. Galić, M. Baranović, K. Križanović, et al., Geospatial data streams: Formal framework and implementation, Data Knowl. Eng. 91 (2014) 1–16, http://dx.doi.org/10.1016/j.datak.2014.02.002.

[52] Y. Papanikolaou, G. Tsoumakas, I. Katakis, Hierarchical partitioning of the output space in multi-label data, Data Knowl. Eng. 116 (2018) 42–60, http://dx.doi.org/10.1016/j.datak.2018.05.003.

Zhiqiang Lv was born in Weifang city, Shandong province, China in 1995. He received the bachelor degree in software engineering from Ludong University, China, in 2019. From 2016 to 2022, he studied in the software parallel group of the State Key Laboratory of Computer Architecture, Institute of Computer Technology Chinese Academy of Sciences. He is studying for the Ph.D. degree in Qingdao University, China, majoring in software engineering. His main research directions are traffic demand research based on traffic trajectory and travel time, traffic data forecast research based on traffic flow, speed and traffic congestion, high performance parallel computing research based on deep learning and reinforcement learning.



Xiaotong Wang was born in Tai'an City, Shandong Province, China in 1999. She received a bachelor's degree in information management and information system from Qufu Normal University, China, in 2021. She is currently studying for a master's degree at Qingdao University, China, majoring in computer technology. Her main research interests are deep learning, social media big data analysis, analysis and prediction of crowd travel behavior.

Zesheng Cheng worked on his Ph.D. at University of New South Wales under supervision of Professor S.T. Waller and T.H. Rashidi. He holds the position of Assistant Professor at the College of Computer Science and technology, Qingdao University. His research currently focuses on graph theory, graph representation learning, share mobility, urban traffic planning and applying large-scale data analysis technologies on urban transport network.

Jianbo Li was born in Weifang city, Shandong province, China in 1980. He received the Ph.D. degree in computer science and technology department from the University of Science and Technology of China in 2009. From 2013 to 2014, he was a visiting scholar at Fordham University. He is currently the professor of the college of computer science & technology in Qingdao University and the director of the Institute of ubiquitous network and urban computing of the Qingdao University. He is the chairman of ACM Qingdao Branch, deputy secretary general of Qingdao Computer Society, senior member of China Computer Federation, and member of Internet of Things Professional Committee of China Computer Federation. His research interests include urban computing, mobile social networks and data offloading.

Haoran Li was born in Heze city, Shandong Province in 1992. He got a master degree in computer science and technology at Qingdao University in China. Now, he works at the Institute of Ubiquitous Networks and Urban Computing. His research focuses on urban computing, urban area classification and traffic big data mining.

Zhihao Xu was born in Binzhou city, Shandong province, China, in 1992. He received the bachelor degree in digital media technology from Qingdao University, China, in 2014. He got the master degree in Qingdao University, majoring in computer technology, in 2017. Now, he works at the Institute of Ubiquitous Networks and Urban Computing, Qingdao. His main research directions are deep learning, urban computing, Intelligent Transportation, crowd density research and crowd interest prediction.