Received 28 October 2023, accepted 13 November 2023, date of publication 28 November 2023, date of current version 4 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3336702

RESEARCH ARTICLE

Graph-Guided Neural Network for Tourism Demand Forecasting

JIJIE FAN[1,2], WEIKAI LU[3], SATYVALDIEVA BAKTYGUL ABDURAIMOVNA[4], JINLONG CHENG[2], AND HAOYI FAN[5], (Member, IEEE)

[1]Faculty of Management and Business, Kyrgyz National University named after Jusup Balasagyn, Bishkek 720033, Kyrgyzstan
[2]College of Land and Tourism, Luoyang Normal University, Luoyang 471934, China
[3]Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou 511442, China
[4]Institute of Retraining and Advanced Training, Kyrgyz National University named after Jusup Balasagyn, Bishkek 720033, Kyrgyzstan
[5]School of Computer and Artificial Intelligence, Zhengzhou University, Zhengzhou 450001, China

Corresponding authors: Satyvaldieva Baktygul Abduraimovna (bsatyvaldieva@gmail.com) and Jinlong Cheng (12763744@qq.com)

This work was supported in part by the Key Disciplines of Tourism Management in Henan Province, in part by the Collaborative Innovation Center of Smart Tourism in Henan Province, and in part by the Key Research and Development and Promotion Projects in Henan Province Key Projects of Soft Science Research under Grant 232400411024.

ABSTRACT An accurate tourism demand forecasting model is crucial for tourism decision-makers. In recent years, several deep learning-based models have been developed to predict tourist arrivals via search intensity indices. However, few methods consider the lag effect in the long-term time range and the interaction between different search intensity indices factors. To alleviate the above limitations, we propose a graph-guided tourism demand forecasting network, which can model the lag effect of historical variables on future variables. Specifically, each variable is individually encoded via a convolutional neural network in the time dimension. Then, lag effects are modeled dynamically in a bipartite graph, and mined via graph aggregation. Finally, a fully-connected network is designed for regression prediction. Experimental results on two public datasets demonstrate the superiority of the proposed method in both one-step and multi-step prediction compared with existing methods.

INDEX TERMS Tourism demand forecasting, graph neural network, lag effect.

I. INTRODUCTION

Tourism plays an important role in the global economy. An accurate and effective tourism demand forecasting model has become an urgent need of hotel operators, managers, and other stakeholders because it can help to make correct short-term decisions and avoid unnecessary market risks [1]. However, due to the complexity of the determinants behind tourist arrivals and external intervention measures, accurate tourism demand forecasting has great challenges.

In recent years, with the great success of deep learning in signal processing and time series-related tasks [2], [3], [4], [5] such as wind speed forecasting [6], [7], deep learning-based tourism demand forecasting methods have gradually become a research hotspot [8], [9]. Encouraged by the success of multivariate time series forecasting models [10], [11], [12], some methods attempt to introduce additional tourism-related variables, such as economic factors [13] and weather factors [14]. As tourism related-variables, the search intensity indices (SII) can reflect tourists' preferences and arouse extensive research interest [15], [16], [17]. In addition, the development of the Internet has allowed search intensity indices data to accumulate and be accessed openly. Therefore, this work focuses on tourism demand forecasting based on deep learning and search intensity indices factors.

One of the widely concerned issues in SII-based tourism demand forecasting is the lag effect of variables. Fig. 1 shows an example of search process for the potential Macau tourist. As we have seen, tourist arrivals may be affected by the short-term lag effect of a search a month ago or by the long-term lag effect of a search several months ago. In addition, a single search may result in multiple searches in the future, which indicates that there is the lag effect between search intensity indices factors. To address this issue, the existing works [15], [16], [17] use the attention mechanism for lag selection within the input window. On the one hand,

The associate editor coordinating the review of this manuscript and approving it for publication was Justin Zhang.

© 2023 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

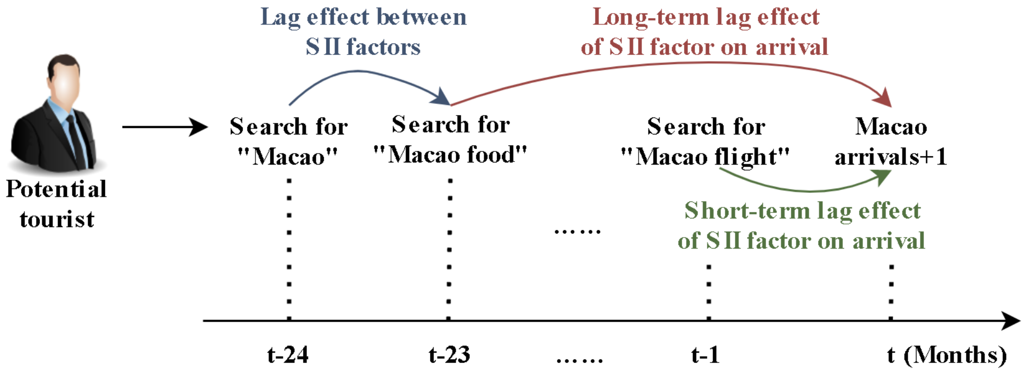


FIGURE 1. An example of online search for the potential Macao tourist.

this solution only considers the lag effect within a limited time range, and the long-term lag effect is ignored. In the real world, some tourists may make travel plans through online searches for a long time in advance, and their travel intentions will be ignored by the model. On the other hand, this solution also ignores the lag effect between SII factors. Sometimes a single search by a tourist may lead to many future searches, and these large searches do not necessarily mean an increase in potential travelers. Ignoring such lagging effects may overestimate the final tourist arrivals.

In addition, most literature on tourism demand forecasting focuses on short-term forecasting, but the effective planning and management of tourism-related departments such as hotels need longer-term forecasting [18]. With the increase of time step, the deviation between the accuracy of the predicted value and the original value gradually increases, so multi-step ahead prediction is still a big challenge [19].

To address the above limitations, we proposed a graph-guided tourism demand forecast network (GTDFN). Specifically, each variable is first encoded along the temporal dimension by a convolution neural network (CNN). Then, all the current variables and all the observable historical variables are combined to form a bipartite graph whose edges are automatically generated by the network. By one-way graph aggregation of graph neural network (GNN), GTDFN can capture the relations between any historical variable and the current variable, including the lag effect of the historical search intensity indices factors on current tourist arrivals and the lag effect of historical search intensity indices factors on the current search intensity indices factors. Finally, future tourist arrivals are predicted via a fully-connected network. By direct prediction, the proposed method can make multi-step predictions in advance.

In sum, the main contributions of this paper are as follows:

- We emphasize the importance of considering the long-term lag effect and the lag effect between SII factors in tourism demand forecasting, and construct a bipartite graph from a global perspective for modeling them.
- We propose GTDFN, which utilizes graph structure learning to automatically capture the potential lag effects of SII factors, and further simulates the impact process via unidirectional graph aggregation.
- We conduct extensive experiments on public real datasets sourced from Hong Kong and Macau, and the experimental results show the superiority of the proposed method.

II. RELATED WORK

A. TOURISM DEMAND FORECASTING

Tourism demand forecasting is important to the healthy and stable development of the tourism industry, and has been widely concerned for a long time [20]. Early tourism demand forecasting methods focus on traditional time series analysis and econometrics. Specifically, Liu et al. [21] adopted the vector autoregressive model to explore the causal relations between web searches and actual arrivals of the cultural tourism destination. Pan and Yang [22] used the ARIMA model to predict hotel occupancy via web search terms, weather, and other factors. Jiang et al. [23] proposed a tourism demand forecasting framework combining fuzzy time series (FTS) and an atom search optimization (ASO) algorithm. Li et al. [24] combined the error correction model with a varying parameter model to generate a new single equation model for tourism demand forecasting, so that the time-varying parameter error can be corrected. Although the above methods have brought progress in the domain of tourism demand forecasting, their performance is not significant because they are all traditional machine learning based algorithms whose fitting ability is often limited.

In recent years, deep learning-based methods have gradually become the most mainstream tourism demand forecasting method due to their strong nonlinear fitting ability and automatic feature extraction ability. Specifically, Li et al. [25] used graph convolution neural network to extract spatial features and LSTM to extract temporal features, so as to improve the performance of tourism demand prediction. Ni et al. [26] combine CNN and LSTM for short-term tourist flow forecasting. Bi et al. [27] encoded the tourism demand data into images, extracted features by

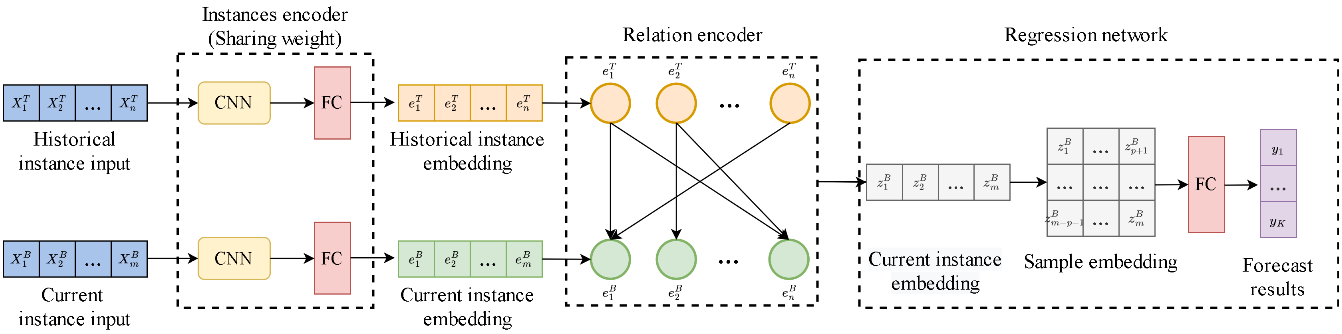


FIGURE 2. Architecture of GTDFN.

using the convolution layer and pooling layer, and further predicted the tourist arrivals by using LSTM.

In this study, we focused on the methods that use SII factors as inputs. Specifically, Law et al. [15] proposed a long short-term memory network (LSTM) for tourism arrivals forecasting, and added the attention mechanism for features selection and lag selection. The experimental results on a realistic Macao dataset show that the performance is higher than several traditional methods. Based on [15], Zhang et al. combined data from two locations and used a single model for simultaneous prediction, achieving better performance. Zhang et al. [16] proposed an STL-DADLM model, which alleviates the over-fitting problem of the LSTM model by STL decomposition and improves the accuracy of long-term and short-term predictions. The above three SII-based works all use attention mechanisms for lag selection. However, due to the limitation of input window size, this processing method can only be considered within a finite time range, ignoring the global lag effect. In addition, this method only learns an importance score for each timestamp to consider lag, ignoring the specificity of different SII factors at the same time, and ignoring the lag effect between SII factors.

B. GRAPH LEARNING ON TIME SERIES FORECASTING

Graph learning technology has been widely applied in time series forecasting due to its ability to model complex relations, such as inter-series correlations [28], spatio-temporal correlations [29], [30], [31], [32] and variables correlations [28], [33]. Wu et al. [33] uses GNN to model dependencies among variables in a timestamp and proposes a mix-hop propagation layer and a dilated inception layer to model temporal and spatial dependencies. Recently, Xu et al. [34] proposed a multivariate time series forecasting method based on the instance graph, which extends the relations reasoning between variables in a single timestamp to variables in multiple timestamps, making it easy to model the lagged effects between variables. However, this method treats all training set variables as historical variables, which makes the ground truth of some samples visible during training, resulting in the low generalization ability of the trained model. In addition, due to the use of an invariant set of historical variables in this method, the integrity of historical variables will be lost during the testing procedure. In this work, we solve the above limitations and apply them to the modeling of the lag effect in tourism demand forecasting.

III. METHODOLOGIES

In this section, we present our proposed GTDFN in detail. As shown in Fig. 2, GTDFN consists of three modules including an instance encoder, relation encoder, and regression network. First, variables including SII factors and tourist arrivals of each timestamp are encoded by a CNN encoder. Subsequently, A dynamic GNN is used to extract the lag effect between historical variables and current variables. Finally, regression forecasting is conducted using a fully-connected network.

A. PROBLEM DEFINITION

In this work, we focus on the issue of tourism demand forecast, which aims to predict future tourist arrivals according to the past tourist arrivals and search intensity indices factors. In this problem, the observed data of the i-th month can be represented as a vector $X_i = \{a, s_1, \cdots, s_p, \}$, where a denotes the tourist arrivals, $s_1 \cdots s_p$ denote search intensity indices factors. Given an observed sequence $X = \{X_{t-d}, \cdots, X_{t-1}, X_t\}$, where t is the current timestamp, d indicates past month number, our goal aims to predict the a in $\{X_{t+1}, \cdots, X_{t+h}\}$, where h is the prediction horizon.

B. INSTANCES ENCODER

To explore the lagged impact of SII factors on future variables, we first need to encode each SII factor and tourist arrivals (i.e., time series variables). To further consider time specificity, the same variables at different times are encoded separately. For convenience, we call a variable in a timestamp an instance according to [34]. A 2-layer 1D CNN with ReLU activation is designed to encode each instance. This convolution is performed on the temporal dimension to obtain the temporal dependencies information of variables. Generally, a convolution layer can capture only a short-term dependence on the size of the convolution kernel, but as the convolution layers overlap, the longer-term dependencies can be extracted. In addition, all instances share the same convolution kernel for better generalization.

Finally, a fully-connected layer is connected to CNN to transform embedding dimensions.

The instance encoder has special requirements for input, so we need to convert the original sequence into a specific form. Specifically, each original sequence X is split into an instance set $\{x_i\}_{i=1}^{p+1}$, where $x_i \in \mathbb{R}^t$ is original time series of the i-th instance. On this basis, we constructed two instance sets $D_P = \{x_1^T, x_2^T, \cdots, x_n^T\}$ and $D_C = \{x_1^B, x_2^B, \cdots, x_m^B\}$, where n and m present the total number of samples in D_P and D_C, respectively. D_P contains all instances of the historical samples, and D_C contains all instances of current mini-batch samples. In each mini-batch training procedure, D_P and D_C are fed into the instances encoder to form embedding $\{e_1^T, e_2^T, \cdots, e_n^T\}$ and $\{e_1^B, e_2^B, \cdots, e_m^B\}$, respectively. In addition, each instance batch is sampled in chronological order and added to D_P as the historical instances of the next mini-batch instances after backpropagation, so as to ensure the integrity of the historical variables in time. It is worth noting that during the inference stage, D_t still dynamically increases over time without the need for additional training.

C. RELATIONS ENCODER

Subsequently, we explicitly utilize a bipartite graph to capture the lagged impact of historical instances on current instances. In the graph, each instance is treated as a node whose node attribute is the feature embedding extracted by the instance encoder. To better simulate the process of lagged impact, all edges are only connected between historical and future instances, and all edges are directed edges, with directions pointing from historical instances to future instances. Defining edges in advance is difficult to achieve as it may require complex domain knowledge, so we use graph structure learning techniques to dynamically generate them. Specifically, The adjacency matrix A of the graph is constructed by calculating cosine distance:

$$A_{ij} = \frac{W_1 \mathbf{e}_i^B \cdot W_2 \mathbf{e}_j^T}{\left\| W_1 \mathbf{e}_i^B \right\| \cdot \left\| W_2 \mathbf{e}_j^T \right\|} (1 \leq i \leq m, 1 \leq j \leq n) \quad (1)$$

where W_1 and W_2 denote the trainable weight matrices. With the help of two parameter matrices, the network can learn the most favorable edge connection method for prediction. Subsequently, a unidirectional aggregated GNN is used to further encode the embedding of graph nodes. The process is as follows:

$$\mathbf{z}_i^B = Concat(\mathbf{e}_i^B, \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} A_{ij} \mathbf{e}_j^T) \quad (2)$$

where $\mathcal{N}_i$ is the set including top-N similar nodes for i-th mini-batch node, $Concat(\cdot, \cdot)$ represents matrix join operation, $\mathbf{z}_i^B$ denotes the embedding after aggregation of i-th mini-batch node. Intuitively speaking, the aggregation process of GNN simulates how historical SII factors affect current SII factors and tourist arrivals. The strength of the effect is determined by the edge weights in A, and the construction method of A is automatically optimized by the network. In addition, low impact edges are ignored to prevent the over-smooth embedding. It is worth noting that the first batch has no corresponding historical instance, so we assume that they are not affected by any historical instances and directly feed the embedding output of the instance encoder into the regression network.

D. REGRESSION NETWORK

For final regression forecasting, we fuse the instance embedding corresponding to each sample. Specifically, the i-th sample embedding can be expressed as $Z_i = Concat(\mathbf{z}_{1+i*(p+1)}^B, \cdots, \mathbf{z}_{(i+1)*(p+1)}^B)$. Then, a fully-connected network is used for tourist arrivals regression forecasting:

$$y_i = W_r Z_i + b_r \quad (3)$$

where y_i represents the predicted result of i-th sample, W_r and b_r are trainable weight and bias matrix, respectively. Finally, the mean square error (MSE) is used as the loss function of GTDFN:

$$\mathrm{L} = \frac{1}{K} \sum_{i=1}^{K} \left(y_i - \hat{y}_i\right)^2 \quad (4)$$

where K denotes the total number of training samples, $\hat{y}_i$ is the ground truth tourist arrivals of the i-th sample.

E. TIME COMPLEXITY ANALYSIS

The main time cost of GTDFN calculation sources from the CNN instance encoder and the relation encoder with regression network. Therefore, the total time complexity can be expressed as:

$$O_{\text{sum}} = O_{\text{cnn}} + O_{\text{graph}} \quad (5)$$

where O_{cnn} and O_{graph} are the time complexity of the instance encoder and the relationship encoder, respectively. According to the time complexity of convolutional layers, O_{cnn} can be represented as:

$$O_{\text{cnn}} = O\left(C_{in} * C_{out} * d^2 * K^2\right) \quad (6)$$

where K is the length of convolutional kernel, C_{in} and C_{out} are the number of input and output channels. For the relation encoder with regression network, the time consumption is mainly in the relation encoder. from the eq.(1), we can easily find the dimension of inputs $e_i^B \in R^{M*D}$, $e_j^T \in R^{N*D}$, and the output dimension $A_{ij} \in R^{N*M}$, thus the time complexity of relation encoder with regression networks can be represented as:

$$O_{\text{graph}} = O\left(D^2 * M\right) \quad (7)$$

To sum up, the time complexity of GTDFN is $O_{\text{sum}} = O\left(C_{\text{in}} * C_{\text{out}} * 2\, d^2 * K^2 + D^2 * M\right)$.

TABLE 1. The parameter search scope for GTDFN and baselines.

Methods	Learning rate	Weight decay	Dense layer size	Neighbors	Batch size	LSTM units
GTDFN	{0.01,0.005,0.0001}	{1e-5,1e-6,1e-7}	{32,64,128,256,512}	{10,20,30,40}	{1,2,6}	-
DLM	{0.01,0.005,0.0001}	{1e-5,1e-6,1e-7}	{32,64,128,256,512}	-	{1,2,6}	{32,64,128,256,512}
GP-DLM	{0.01,0.005,0.0001}	{1e-5,1e-6,1e-7}	{32,64,128,256,512}	-	{1,2,6}	{32,64,128,256,512}
IGMTF	{0.01,0.005,0.0001}	{1e-5,1e-6,1e-7}	{32,64,128,256,512}	{10,20,30,40}	{1,2,6}	-

IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

1) DATASETS

To evaluate the effectiveness of the proposed method, we adopted the public datasets "HK-2018" and "MO-2018" provided by Zhang et al. [17], which collected the monthly tourist arrivals and search intensity indices factors of Macao and Hong Kong. For tourist arrivals, Macau's data is collected from the Statistics and Census Services Office of the Macau government, while Hong Kong's data is sourced from the Hong Kong Tourism Board.[1] For SII factors, data was collected from Google Trends[2] and Baidu Index[3] websites. There are a total of 193 search keywords, covering seven categories related to tourism, namely dining, lodging, transportation, tour, clothing, shopping, and recreation. The time range of the two datasets is from January 2011 to August 2018, and each time step spanning one month (92 observations).

To accelerate the convergence speed and improve the accuracy of the model, the Min-max normalization was used for data preprocessing. Specifically, for each variable s in the dataset:

$$s' = \frac{s - s_{\min}}{s_{\max} - s_{\min}} \tag{8}$$

where $s_{\max}$ and $s_{\min}$ represent the maximum and minimum values of variable s in the dataset, respectively.

For the validation method, we did not follow the walk-forward validation [17] adopted by the dataset author, because it needs to retrain a model at each step of forward validation, which is expensive and not conducive to the actual deployment and application of the model. We are more concerned about the long-term performance of the model after training, so we split the dataset with a train-validation-test ratio of 60%-20%-20%, select the model that performs best in the verification set after training, and evaluate the best model in the test set.

2) BASELINES

We compare the proposed model of time series forecasting with four well-performing baselines as follows:

(1) *LSTM*: In this baseline, an LSTM layer is used to extract the long-term dependency information of the time series, and then an MLP is adopted for regression prediction.

(2) *RAE [6]*: Rough Autoencoder (RAE) is a deep autoencoder with stacked autoencoder (SAE) and stacked denoising autoencoder (SDAE) for time series forecasting, which uses unsupervised feature learning from the unlabeled data and a supervised regression layer for forecasting.

(3) *DLM [15]*: It adds an attention mechanism based on LSTM for lag order selection and feature selection.

(4) *GP-DLM [17]*: This method presents a group pooling method to fuse two datasets and simultaneously predict tourist arrivals of Hong Kong and Macao via DLM.

(5) *IGMTF [34]*: This method uses a gate recurrent unit as the instance encoder and constructs a training set instances-Mini batch instances graph to mine the correlation between instances.

3) EVALUATION METRICS

In experiments, the mean absolute percentage error (MAPE), mean absolute error (MAE), and root mean square error (RMSE) are used as evaluation metrics, and they were defined as follows:

$$MAE = \frac{1}{T}\sum_{i=1}^{T}\left|y_i - \hat{y}_i\right|, \tag{9}$$

$$RMSE = \sqrt{\frac{1}{T}\sum_{i=1}^{T}\left(y_i - \hat{y}_i\right)^2}, \tag{10}$$

$$MAPE = \frac{100\%}{T}\sum_{i=1}^{T}\left|\frac{\hat{y}_i - y_i}{y_i}\right|, \tag{11}$$

where T is the total number of the testing samples.

4) IMPLEMENTATION DETAILS

All experiments were coded using PyTorch [35] (v1.7.1) with CUDA 11.7, and performed on a workstation with an NVIDIA GeForce RTX3090 GPU and an Intel Xeon E5-2680 CPU. Adam [36] is used for all models with 100 epochs. In the two datasets, the look-back window size is set to 6, and the prediction horizon is set to 1, 2, and 4 respectively. We perform a grid search to find the optimal hyperparameters of the GTDFN and all baselines, and the scope of searched parameters is shown in Table 1. Specifically, we train the model with all possible parameter combinations and determine the optimal parameters based on the model's performance.

[1] https://www.dsec.gov.mo/Statistic.aspx?NodeGuid=251baebb-6e5b-4452-8ad1-7768eafc99ed

[2] https://trends.google.com

[3] https://zhishu.baidu.com

TABLE 2. Experimental results on the HK-2018 dataset.

Horizons	Methods	RMSE	MAE	MAPE
1	GTDFN	**276820.78**	**192967.30**	**3.78**
1	LSTM	328725.78	251190.00	5.23
1	RAE [6]	317281.27	251443.12	5.21
1	DLM [15]	305500.01	251758.53	5.20
1	GP-DLM [17]	300989.36	227858.75	4.65
1	IGMTF [34]	323761.38	242215.67	4.75
1	improving	8.02%	15.31%	18.70%
2	GTDFN	**272031.04**	**226409.20**	**4.59**
2	LSTM	320649.40	268699.78	5.46
2	RAE [6]	322413.43	269711.21	5.49
2	DLM [15]	324816.74	271230.68	5.52
2	GP-DLM [17]	344762.74	275108.5	5.52
2	IGMTF [34]	346632.22	268689.93	5.34
2	improving	15.16%	15.73%	14.04%
4	GTDFN	**327814.22**	**252605.73**	**5.05**
4	LSTM	372493.72	304746.18	6.32
4	RAE [6]	366043.66	291134.53	5.94
4	DLM [15]	356293.62	283104.71	5.80
4	GP-DLM [17]	368760.63	294461.59	6.05
4	IGMTF [34]	366893.60	299093.68	6.03
4	improving	7.99%	10.77%	12.93%

TABLE 3. Experimental results on the MO-2018 dataset.

Horizon	Methods	RMSE	MAE	MAPE
1	GTDFN	**139188.50**	**110614.03**	**4.10**
1	LSTM	178695.10	150878.70	5.57
1	RAE [6]	197865.33	171921.45	5.55
1	DLM [15]	204083.20	148240.78	5.56
1	GP-DLM [17]	190889.11	136472.51	4.92
1	IGMTF [34]	203719.68	159173.20	5.65
1	improving	22.10%	18.94%	16.66%
2	GTDFN	**158295.30**	**126771.74**	**4.64**
2	LSTM	198906.50	168488.56	6.25
2	RAE [6]	197022.32	167922.77	6.17
2	DLM [15]	199477.55	164325.18	6.15
2	GP-DLM [17]	193648.75	160511.29	5.84
2	IGMTF [34]	203892.24	161383.75	5.73
2	improving	20.41%	21.02%	19.02%
4	GTDFN	**172009.15**	**139768.6**	**5.13**
4	LSTM	215848.06	178054.43	6.78
4	RAE [6]	211955.11	174922.11	6.32
4	DLM [15]	218924.05	165060.40	6.31
4	GP-DLM [17]	194075.83	163698.37	6.01
4	IGMTF [34]	196318.78	163151.96	5.91
4	improving	11.37%	14.33%	13.19%

B. EXPERIMENTAL RESULTS

The experimental results of five methods on the HK-2018 and MO-2018 datasets are presented in Table 2 and Table 3. As we can see, the GTDFN shows the best performance in all metrics and all datasets. Particularly, on the HK-2018 dataset, the GTDFN outperforms the best baseline at the term of MAPE by 18.70% when the prediction horizon is 1. On the MO-2018 dataset, the performance of GTDFN is 22.10% better than the best baseline at the term of RMSE when the prediction horizon is 1. When the prediction horizon increases, the prediction performance of all methods decreases, but GTDFN still maintains a relatively stable performance improvement compared with other methods, which indicates that the GTDFN can effectively capture the long-term variation of tourism demand.

Overall, LSTM shows the worst performance among all baselines due to its lack of consideration of the impact between variables. Although RAE performs better than LSTM because of its uncertain factors modeling ability in the stacked denoising autoencoder, the improvement is still

TABLE 4. Ablation results on the HK-2018 dataset.

Horizons	Methods	RMSE	MAE	MAPE
1	GTDFN	**276820.78**	**192967.30**	**3.78**
1	GTDFN(-GNN)	315946.68	269706.62	5.54
1	GTDFN(-order)	320850.99	257478.75	5.09
1	improving	13.72%	25.05%	25.73%
2	GTDFN	**272031.04**	**226409.20**	**4.59**
2	GTDFN(-GNN)	352557.94	295627.94	6.02
2	GTDFN(-order)	321752.43	275516.75	5.51
2	improving	15.45%	17.82%	16.69%
4	GTDFN	**327814.22**	**252605.73**	**5.05**
4	GTDFN(-GNN)	377026.86	309458.80	6.24
4	GTDFN(-order)	335104.01	278882.00	5.65
4	improving	2.17%	9.42%	10.61%

TABLE 5. Ablation results on the MO-2018 dataset.

Horizons	Methods	RMSE	MAE	MAPE
1	GTDFN	**139188.50**	**110614.03**	**4.10**
1	GTDFN(-GNN)	205049.40	172423.52	6.31
1	GTDFN(-order)	183715.38	140398.73	5.07
1	improving	24.23%	21.21%	19.13%
2	GTDFN	**158295.30**	**126771.74**	**4.64**
2	GTDFN(-GNN)	202663.51	174859.69	6.49
2	GTDFN(-order)	184895.63	142151.64	5.10
2	improving	14.38%	10.81%	9.01%
4	GTDFN	**172009.15**	**139768.6**	**5.13**
4	GTDFN(-GNN)	229211.17	196672.95	7.45
4	GTDFN(-order)	185769.39	144578.31	5.22
4	improving	7.40%	3.32%	1.72%

slight because it neglects the potential lag effects of SII factors. In contrast, the benefits of DLM using the attention mechanism for lag order selection are obvious, and five of the six groups of experiments exceed LSTM. On the basis of DLM, the performance gain of GP-DLM using group pooling is also obvious. However, the performance of DLM and GP-DLM is far from that of GTDFN, possibly because they only consider the lag effect in a limited range, resulting in incomplete learned representation.

Compared with the IGMTF method, GTDFN has also been greatly improved. For MAPE, the performance of GTDFN is 16.75% higher than IGMTF on the HK-2018 dataset and 19.79% on the MO-2018 dataset, on average. For this phenomenon, one reason is that IGMTF is prone to information leakage during the training stage, resulting in a lack of generalization in the inference stage. On the other hand, the fixed historical instances of IGMTF lead to incomplete lag effect mining.

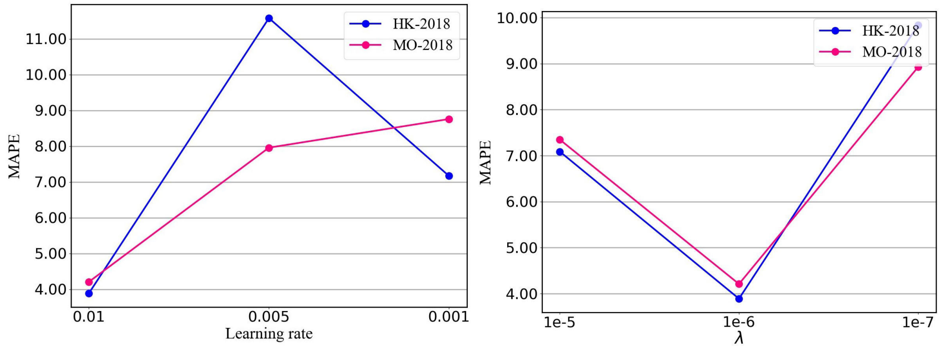
C. ABLATION ANALYSIS

To explore the effectiveness of purposeful design in the model, we designed two sub-methods for comparison: 1) GTDFN(-GNN): GTDFN without graph structure learning and GNN module, only using CNN and MLP for feature extraction and forecasting. 2) GTDFN(-order): GTDFN does not guarantee the order of historical and current variables, and uses all training instances as historical instances in the training and inference stage.

The ablation results on both datasets are shown in Table 4 and Table 5. Among the three methods, GTDFN (-GNN) showed the worst performance, while GTDFN (-order) showed further improvement on all horizons of the two datasets, indicating that using GNN to capture relationships between instances is effective for tourism demand forecasting. On the basis of GTDFN (-order), GTDFN further comprehensively improves performance, which further proves the necessity of mining complete lag effects.

D. HYPERPARAMETERS SENSITIVITY ANALYSIS

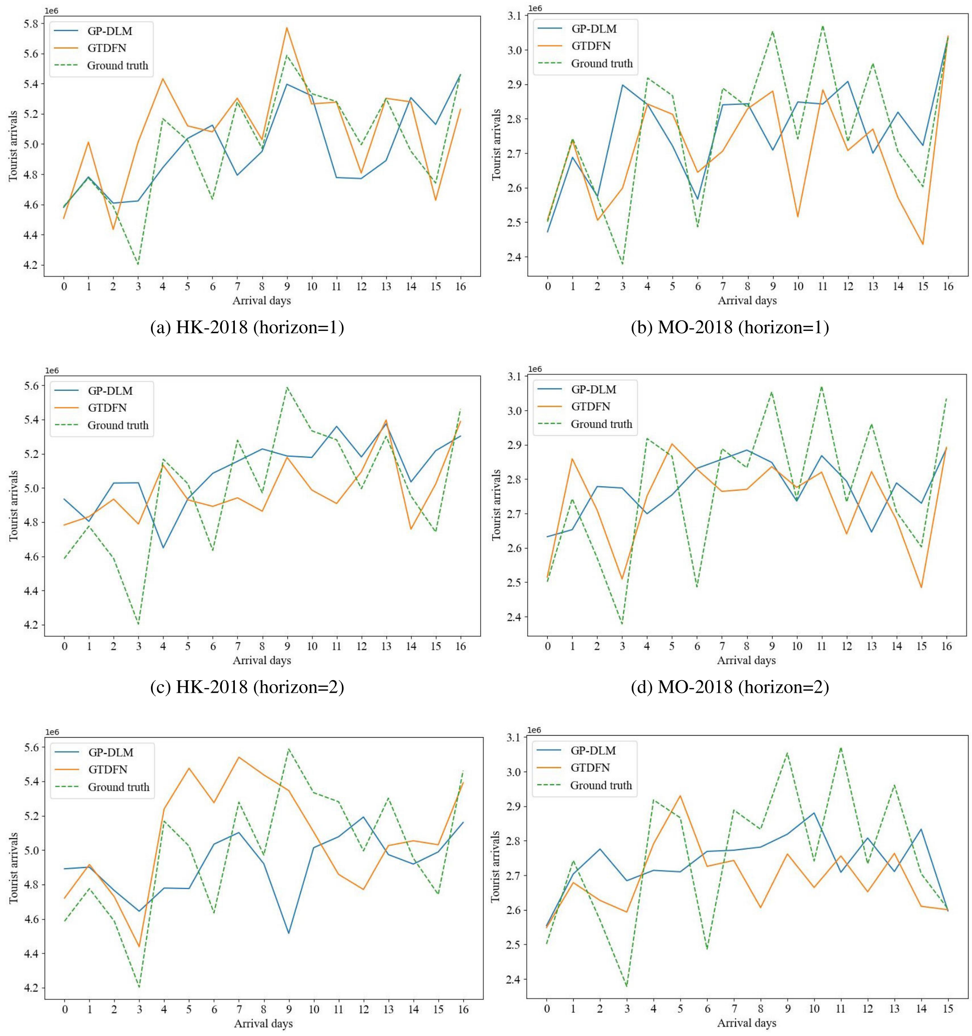
In this section, we discuss in detail the effects of learning rate and regularization factor λ as key hyperparameters. The MAPE is used as an evaluation metric to reflect the influence of hyperparameters on the experimental results. Figure 3 shows the effect of two hyperparameters on prediction results on two datasets. The performance of GTDFN improves with the increase of learning rate, which is manifested as the decrease of MAPE, showing that the proposed model has better performance and convergence ability when the learning rate is large, and often a larger learning rate is also conducive to the generalization ability of the model. The performance of



(a) Effect of the learning rate

(b) Effect of λ

FIGURE 3. **The experimental results of hyperparameters sensitivity analysis.**



(a) HK-2018 (horizon=1)

(b) MO-2018 (horizon=1)

(c) HK-2018 (horizon=2)

(d) MO-2018 (horizon=2)

(e) HK-2018 (horizon=4)

(f) MO-2018 (horizon=4)

FIGURE 4. **Visualization of the prediction results of the proposed GTDFN and GP-DLM. It includes the results of two datasets(HK-2018 and MO-2018) and three prediction horizons (1, 2, and 4).**

GTDFN decreases first and then increases with the increase of λ. A weak regularization tends to result in over-fitting, while a strong regularization tends to result in under-fitting, so a modest λ is better for our model.

E. VISUALIZATION ANALYSIS

To better understand the benefits of GTDFN, we visualized its forecasting results and those of GP-DLM, which performed best in the baseline, and presented them in Figure 4.

TABLE 6. Importance score of Macau from GTDFN.

SII factor	Importance score
HK to Macau	1.0000
Macau shopping guide	0.9907
Venetian Macau	0.9779
Ferry Macau to Hong Kong airport	0.9612
Macau shopping mall	0.9546
Macau food festival 2017	0.9504
Hong Kong to Macau ferry	0.9498
Macau food guide	0.9493
Venetian casino Macau	0.9442
Macau show bar	0.9414

TABLE 7. Importance score of Hongkong from GTDFN.

SII factor	Importance score
What to do in HongKong	1.0000
China travel	0.8695
Hong Kong airport	0.8526
Hong Kong shopping	0.8452
Bangkok	0.8150
Hotel Lisboa	0.7749
Hong Kong weather	0.7747
Agoda	0.7664
Hong Kong tourism	0.7643
Hong Kong hotel	0.7640

Compared with GP-DLM, the series predicted by GTDFN under two datasets and three prediction horizons are more similar to the ground truth, which explains why GTDFN performs better in metrics. In addition, GTDFN can better capture the changing trend of visitor arrivals, which can better help managers make decisions in practical applications. This advantage is more obvious when the prediction horizon increases, although both methods show reduced forecasting performance.

F. SEARCH INTENSITY INDICES FACTORS IMPORTANCE ANALYSIS

To further understand the proposed GTDFN method, we analyze the importance of search intensity indices (SII) factors learned by the model. Specifically, the average edge weight of each SII factor node and tourist arrival node in the learned graph is taken as the importance score of the SII factor to tourism demand. Subsequently, all importance scores were performed Min-Max normalization. The top-10 importance scores from both datasets are shown in Table 6 and Table 7, respectively. On the MO-2018 dataset, the keyword ''Macau'' is included in all 10 SII factors that the model focuses on most. Among them, ''Venetian'', ''Macau shopping'', and ''Macau food'' appeared twice, indicating what tourists want to do when they come to Macau. In addition, traffic-related searches are important. In the HK-2018 dataset, the GTDFN also notes the keyword ''Hong Kong'' in most of the top 10 SII factors. It is worth noting that both datasets contain Hong Kong and Macao search terms and that GTDFN can identify the variables most relevant to arrivals, which explains why GTDFN performs well.

V. CONCLUSION

In this paper, we investigate the tourism demand forecasting task from a novel perspective of using the bipartite graph to model the lag effect of SII factors and propose an end-to-end GTDFN method. Specifically, the embedding contains temporal dependencies of all SII factors and tourist arrivals are extracted by a CNN encoder. Then, the lag effects of all instances are generated dynamically via calculating the cosine distance of instances embedding, and the instances are further encoded by the relation encoder. Finally, a multi-layer perceptron is used for one-step to multi-step prediction. By comparing the existing methods, we demonstrate the proposed GTDFN not only shows superiority in one-step forecasting, but also can more accurately predict the long-term trend of tourism demand.

REFERENCES

[1] G. Xing, S. Sun, D. Bi, J. Guo, and S. Wang, ''Seasonal and trend forecasting of tourist arrivals: An adaptive multiscale ensemble learning approach,'' *Int. J. Tourism Res.*, vol. 24, no. 3, pp. 425–442, May 2022.

[2] M. Ćirić, B. Predić, D. Stojanović, and I. Ćirić, ''Single and multiple separate LSTM neural networks for multiple output feature purchase prediction,'' *Electronics*, vol. 12, no. 12, p. 2616, Jun. 2023.

[3] N. Sun, S. Zhang, T. Peng, J. Zhou, and X. Sun, ''A composite uncertainty forecasting model for unstable time series: Application of wind speed and streamflow forecasting,'' *IEEE Access*, vol. 8, pp. 209251–209266, 2020.

[4] E. E. Elattar, S. K. Elsayed, and T. A. Farrag, ''Hybrid local general regression neural network and harmony search algorithm for electricity price forecasting,'' *IEEE Access*, vol. 9, pp. 2044–2054, 2021.

[5] M. Khodayar, J. Wang, and Z. Wang, ''Energy disaggregation via deep temporal dictionary learning,'' *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1696–1709, May 2020.

[6] M. Khodayar, O. Kaynak, and M. E. Khodayar, ''Rough deep neural architecture for short-term wind speed forecasting,'' *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 2770–2779, Dec. 2017.

[7] M. Khodayar, J. Wang, and M. Manthouri, ''Interval deep generative neural network for wind speed forecasting,'' *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3974–3989, Jul. 2019.

[8] T. Peng, J. Chen, C. Wang, and Y. Cao, ''A forecast model of tourism demand driven by social network data,'' *IEEE Access*, vol. 9, pp. 109488–109496, 2021.

[9] H. Wang and W. Liu, ''Forecasting tourism demand by a novel multi-factors fusion approach,'' *IEEE Access*, vol. 10, pp. 125972–125991, 2022.

[10] F. Shen, J. Liu, and K. Wu, ''Multivariate time series forecasting based on elastic net and high-order fuzzy cognitive maps: A case study on human action prediction through EEG signals,'' *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 8, pp. 2336–2348, Aug. 2021.

[11] X. Zhang, Y. Lei, H. Chen, L. Zhang, and Y. Zhou, ''Multivariate time-series modeling for forecasting sintering temperature in rotary kilns using DCGNet,'' *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 4635–4645, Jul. 2021.

[12] P. Zhang, H. Jin, H. Dong, W. Song, and L. Wang, ''LA-LMRBF: Online and long-term web service QoS forecasting,'' *IEEE Trans. Services Comput.*, vol. 14, no. 6, pp. 1809–1823, Nov. 2021.

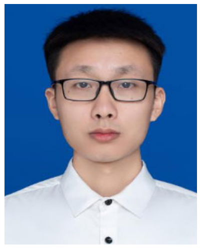
[13] C. Goh and R. Law, ''Incorporating the rough sets theory into travel demand analysis,'' *Tourism Manag.*, vol. 24, no. 5, pp. 511–517, Oct. 2003.

[14] J. Day, N. Chin, S. Sydnor, and K. Cherkauer, ''Weather, climate, and tourism performance: A quantitative analysis,'' *Tourism Manag. Perspect.*, vol. 5, pp. 51–56, Jan. 2013.

[15] R. Law, G. Li, D. K. C. Fong, and X. Han, ''Tourism demand forecasting: A deep learning approach,'' *Ann. Tourism Res.*, vol. 75, pp. 410–423, Mar. 2019.

[16] Y. Zhang, G. Li, B. Muskat, and R. Law, "Tourism demand forecasting: A decomposed deep learning approach," *J. Travel Res.*, vol. 60, no. 5, pp. 981–997, May 2021.
[17] Y. Zhang, G. Li, B. Muskat, R. Law, and Y. Yang, "Group pooling for deep tourism demand forecasting," *Ann. Tourism Res.*, vol. 82, May 2020, Art. no. 102899.
[18] A. Salamanis, G. Xanthopoulou, D. Kehagias, and D. Tzovaras, "LSTM-based deep learning models for long-term tourism demand forecasting," *Electronics*, vol. 11, no. 22, p. 3681, Nov. 2022.
[19] B. Bera, C.-L. Lin, S.-C. Huang, J.-W. Liang, and P. T. Lin, "Establishing a real-time multi-step ahead forecasting model of unbalance fault in a rotor-bearing system," *Electronics*, vol. 12, no. 2, p. 312, Jan. 2023.
[20] H. Song, R. T. R. Qiu, and J. Park, "A review of research on tourism demand forecasting: Launching the annals of tourism research curated collection on tourism demand forecasting," *Ann. Tourism Res.*, vol. 75, pp. 338–362, Mar. 2019.
[21] Y.-Y. Liu, F.-M. Tseng, and Y.-H. Tseng, "Big data analytics for forecasting tourism destination arrivals with the applied vector autoregression model," *Technol. Forecasting Social Change*, vol. 130, pp. 123–134, May 2018.
[22] B. Pan and Y. Yang, "Forecasting destination weekly hotel occupancy with big data," *J. Travel Res.*, vol. 56, no. 7, pp. 957–970, Sep. 2017.
[23] P. Jiang, H. Yang, R. Li, and C. Li, "Inbound tourism demand forecasting framework based on fuzzy time series and advanced optimization algorithm," *Appl. Soft Comput.*, vol. 92, Jul. 2020, Art. no. 106320.
[24] G. Li, K. K. F. Wong, H. Song, and S. F. Witt, "Tourism demand forecasting: A time varying parameter error correction model," *J. Travel Res.*, vol. 45, no. 2, pp. 175–185, Nov. 2006.
[25] C. Li, W. Zheng, and P. Ge, "Tourism demand forecasting with spatiotemporal features," *Ann. Tourism Res.*, vol. 94, May 2022, Art. no. 103384.
[26] T. Ni, L. Wang, P. Zhang, B. Wang, and W. Li, "Daily tourist flow forecasting using SPCA and CNN-LSTM neural network," *Concurrency Comput., Pract. Exper.*, vol. 33, no. 5, p. e5980, Mar. 2021.
[27] J.-W. Bi, H. Li, and Z.-P. Fan, "Tourism demand forecasting with time series imaging: A deep learning model," *Ann. Tourism Res.*, vol. 90, Sep. 2021, Art. no. 103255.
[28] D. Cao, Y. Wang, J. Duan, C. Zhang, X. Zhu, C. Huang, Y. Tong, B. Xu, J. Bai, and J. Tong, "Spectral temporal graph neural network for multivariate time-series forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 17766–17778.
[29] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3634–3640.
[30] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 17804–17815.
[31] J. Zhu, X. Han, H. Deng, C. Tao, L. Zhao, P. Wang, T. Lin, and H. Li, "KST-GCN: A knowledge-driven spatial–temporal graph convolutional network for traffic forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15055–15065, Sep. 2022.
[32] Y. Qin, F. Zhao, Y. Fang, H. Luo, and C. Wang, "Memory attention enhanced graph convolution long short-term memory network for traffic forecasting," *Int. J. Intell. Syst.*, vol. 37, no. 9, pp. 6555–6576, Sep. 2022.
[33] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 753–763.
[34] W. Xu, W. Liu, J. Bian, J. Yin, and T.-Y. Liu, "Instance-wise graph-based framework for multivariate time series forecasting," 2021, *arXiv:2109.06489*.
[35] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, and L. Antiga, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.
[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–12.

JIJIE FAN received the B.S. degree in management and the M.S. degree in economics from Henan University, in 2013 and 2018, respectively. She is currently pursuing the Ph.D. degree in management with Kyrgyz National University named after Jusup Balasagyn. Her current research interests include big data analysis, tourism demand forecasting, and tourism management analysis.



WEIKAI LU received the M.S. degree in electrical engineering from the Fujian University of Technology, Fuzhou, China. He is currently pursuing the Ph.D. degree with the South China University of Technology, Guangzhou, China. His current research interests include deep learning and time series analysis.

SATYVALDIEVA BAKTYGUL ABDURAIMOVNA received the Ph.D. degree in economics from the National Academy of Science of the Kyrgyz Republic and the Specialist degree in marketing management from Kyrgyz National University. Currently, she is an Associate Professor with Kyrgyz National University named after Jusup Balasagyn, Bishkek. Her current research interests include regional economics, spatial economics, and regional marketing.

JINLONG CHENG received the B.S. degree in history and applied history from Henan University, in 2003, the M.S. degree in human geography from Shanghai Normal University, in 2006, and the Ph.D. degree in human geography from Henan University, in 2011. Since 2015, he has been a Professor with Luoyang Normal University, Luoyang, China, where he is currently with the School of Land and Tourism. His current research interests include tourism management, tourism geography, and tourism economics.

HAOYI FAN (Member, IEEE) received the Ph.D. degree from the School of Computer Science and Technology, Harbin University of Science and Technology, in 2021. He is currently with the School of Computer and Artificial Intelligence, Zhengzhou University, Zhengzhou. His current research interests include pattern recognition, data mining, and deep learning.