Aerospace Science and Technology 93 (2019) 105113

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

Deep learning based short-term air traffic flow prediction considering temporal–spatial correlation

Yi Lin [a,b], Jian-wei Zhang [a], Hong Liu [a,*]

[a] National Key Laboratory of Fundamental Science on Synthetic Vision, Sichuan University, No. 24 South Section 1, Yihuan Road, Chengdu, Sichuan, 610000, China
[b] Department of Civil and Environmental Engineering, University of Wisconsin-Madison, 1217 Engineering Hall, 1415 Engineering Drive, Madison, WI, 53706, USA

ARTICLE INFO

Article history:
Received 18 July 2018
Received in revised form 7 April 2019
Accepted 12 April 2019
Available online 17 April 2019

Keywords:
Air traffic flow prediction
ConvLSTM
Deep learning
Temporal–spatial correlation
Traffic flow matrix

ABSTRACT

In order to improve the accuracy and stability of air traffic flow prediction, an end-to-end deep learning-based model is proposed in this paper. By analyzing spatial correlations of adjacent areas and temporal correlations of historical traffic on given area, we firstly apply the gridded map method (including the flight levels) to encode the whole air traffic flow situation into a new data representation, i.e., traffic flow matrix (TFM). By the proposed data representation, inherent features of air traffic flow and their transition patterns on different cells and flight levels can be represented comprehensively. Learning from the powerful ability of convolutional neural network and recurrent neural network on modeling spatial and temporal correlations, the ConvLSTM module is proposed to build a trainable model for air traffic flow prediction. Since the output of the proposed model is also the TFM and shows the overall air traffic situation of studied regions, we call the proposed model as an end-to-end one. Experimental results on real data show the superior performance over existing approaches on prediction accuracy and stability. Furthermore, the proposed model can also predict the flow distribution on different flight levels in our application, which promotes the level of air traffic management. By analyzing the distribution of prediction errors on different cells, flight levels and predicting instants, we can draw the conclusion that spatial and temporal transition patterns of flight flow in air traffic system are fully learned by the proposed model. With the proposed model, more efficient air traffic flow measures are expected to be fulfilled to improve the operation efficiency.

1. Introduction

AIR traffic flow prediction (ATFP) is an essential technique in air traffic research. The goal of ATFP is to estimate the traffic distribution for given airspaces in the future based on real-time traffic situation and historical operating data. With precise and prompt information of air traffic flow for certain airspaces, more proper decisions are expected to be made for air traffic infrastructure planning by concern departments. As the core foundation of air traffic operation, it also helps Air Traffic Controllers (ATCOs) to make more effective measures during flight control, which can ensure the operation safety, relieve the air traffic congestion, reduce the exhaust emissions, and improve the operation efficiency of Air Traffic Control (ATC). Currently, the concerned departments of Civil Aviation Administration of China (CAAC) perform air traffic control mainly by regarding single flight as the target object. However, this approach is facing enormous challenges due to the sharp growth of air traffic demands in China. Since those approaches failed to capture the global distribution of air traffic flow, controlling instructions made by sector ATCOs are usually not optimal for the local air traffic situation. Moreover, with the increasing of the air transportation, the problem of airspace resource shortage becomes detrimental and serious flight congestion and delay frequently occurred in some busy airports or terminals, such as Beijing, Shanghai, et al.

In ATC research, ATFP is usually divided into three levels based on the prediction term, i.e., long-term, mid-term and short-term [1,2]. In this paper, we mainly focus on the short-term air traffic flow prediction, in which the model predicts the air traffic flow in the near future, typically 0 to 30 minutes [3]. Since air traffic is a complicated time-varying system, any current traffic state may have significant influence on future traffic flow, that is why air traffic flow prediction is important to the ATC research. Once the traffic flow distribution in local area is estimated accurately, ATCOs can make more proper plans to ensure the traffic safety and improve the operation efficiency. Based on an accurate ATFP,

* Corresponding author.
E-mail addresses: scu_lyi@stu.scu.edu.cn, ylin353@wisc.edu (Y. Lin), Zhangjianwei@scu.edu.cn (J.-w. Zhang), liuhblue@126.com (H. Liu).

https://doi.org/10.1016/j.ast.2019.04.021
1270-9638/

all controlling instructions of ATCOs can meet safety requirements currently and avoid any potential conflict or risk in the future.

From the flight distribution of nationwide airspaces in China, we can see that the flight density is higher in hub airspaces and their bordering areas (adjacent areas) compared to any others. The fact indicates that the traffic flow distribution of given airspaces have significant spatial dependencies (higher similarity) with their neighboring areas [4]. Meanwhile, historical traffic patterns also affect the current traffic flow for certain airspaces, which reveals the temporal correlation of traffic flow. In summary, the air traffic flow in given airspace is influenced by the following two factors which should be considered in ATFP models.

a) Spatial correlation: the evolution of flight flow in its adjacent areas. Flights in neighbor regions may fly into the target airspace in the future according to their heading.

b) Temporal correlation: the traffic situation of past instants in current area. Flights may stay in current airspace or fly to any of its adjoining airspaces in the future. In addition, based on the flight schedule in China, the air traffic presents cyclical (weekly) characteristics, which also provides temporal related information for the prediction model.

Based on the fact that both spatial and temporal correlations should be taken into consideration for the air traffic flow prediction, we propose a novel deep learning-based short-term air traffic flow prediction model, in which the traffic transitions of adjacent areas and historical traffic situation are well captured by the proposed data representation. An integrated neural network called ConvLSTM is introduced to learn the complex transition patterns of air traffic system. Analogized with classic applications of the spatial correlation, i.e., image, we firstly propose a special data representation to represent the whole air traffic situation in the studied area. The gridded map method is applied to split the airspace into cells with same size [5] to illustrate spatial dependencies of traffic flow. As to the air traffic system, since aircraft flies in the three-dimensional (3-D) earth space, we also divide the earth space in altitude direction to illustrate the specificity based on flight levels designed by CAAC. Therefore, the whole air traffic situation at given instant is constructed into a 3-D cube based on the proposed data representation, in which each cell denotes a certain spatial extent with longitude, latitude and altitude. All traffic flow in those cells are formed as a new data representation, i.e., traffic flow matrix (TFM). In the TFM, spatial correlations of adjacent areas are captured by neighboring elements. In addition, past traffic situations of adjacent areas also greatly affect the current traffic flow of the studied area, which means that temporal dependencies with historical TFMs should be also considered for ATFP. Subsequently, motivated by the excellent performance of Deep Neural Network (DNN) on modeling complex and non-linear system [6], a deep neural network-based model is proposed in this paper to forecast the air traffic flow in the near future. Specifically, Convolutional Neural Network (CNN) is proposed to model the spatial dependencies among different cells within a single TFM, while Recurrent Neural Network (RNN), the standard paradigm for processing time series data, is introduced to mine temporal correlations of a TFM sequence. Based on the basic RNN block, Long Short-Term Memory (LSTM) [7] was introduced to overcome the problems of information loss and gradient vanishing and achieved excellent performance for sequential modeling. By taking advantages of the mentioned modules on modeling spatial and temporal correlations, the ConvLSTM module [8] comes into being, which serves as the fundamental component of our proposed model. To consider temporal dependencies of historical traffic situations, the 3-D TFM is extended to a 4-D tensor which serves as the input of the proposed model. The four modes of the input tensor denote the timestep of the historical TFM sequence (i.e., the look-ahead horizon), the width of the TFM, the height of the TFM and the flight levels respectively. The output of the model is a 3-D tensor (TFM) representing the whole traffic flow situation with the distribution on different flight levels at next predicting time instant. The general structure of our proposed model is sketched in Fig. 1. In this paper, real operating data stored in current Air Traffic Control Systems (ATCSs) is collected to generate training and test samples to evaluate the prediction performance. Three criteria are used to measure the prediction errors on different cells. Moreover, the distribution of traffic flow on flight levels is also estimated by our proposed model, which can enhance available measures and further improve the operating efficiency in ATC. Most importantly, based on the 3-D TFM and our proposed model, a new framework called end-to-end ATFP is obtained, in which we can predict the situational air traffic flow for whole airspaces integrally at current instant from other historical air traffic situations, not only for certain airspaces. The situational air traffic flow can illustrate the global air traffic system accurately, which enhances the supporting effectiveness of the ATFP model.

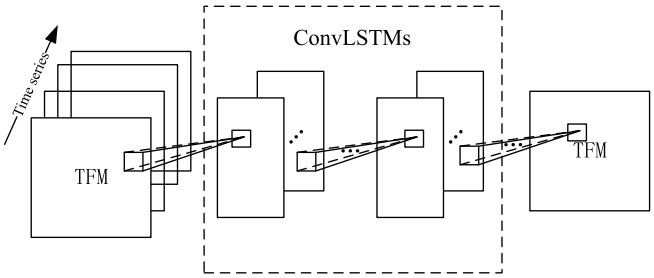

Fig. 1. General structure of the proposed model.

All in all, our original contributions in this work can be summarized below:

a) We propose an end-to-end ATFP framework to predict the air traffic flow for whole airspaces simultaneously, which can precisely depict transition patterns of air traffic flow among different adjacent areas, flight levels and time instants.

b) A special data representation is proposed to illustrate the air traffic situation, in which spatial correlations among adjacent areas and flight levels can be captured accurately.

c) The ConvLSTM block is introduced to the ATFP model to mine spatial and temporal dependencies of air traffic flow in adjacent areas and their historical traffic situations, which improves the prediction performance considerably.

d) The distribution of air traffic flow on different flight levels is also predicted in our proposed model, based on which ATCOs can make more elaborate decisions and further improve the operating efficiency of ATC.

The rest of this paper is organized as follows. The related works are reviewed in Section 2 briefly. Implementation details of our proposed model are introduced in Section 3. Section 4 lists experimental configurations, and experimental results are also reported and discussed in this section. Conclusions and future works are in Section 5.

2. Related works

In recent years, air traffic flow prediction has attracted much attention of researchers all over the world, and there are many outstanding outcomes in this field. Various techniques have been developed and applied for the ATFP task, which are mainly divided into four categories: flight plan-based algorithms, time series algorithms, probabilistic algorithms and machine learning algorithms. Brief discussions of each category are listed below:

a) Flight plan-based algorithms: The 4-D trajectory prediction estimates the fly over time of each waypoint based on the flight

plan [9], which is further used to predict the air traffic flow for concerned airspaces [10,11]. This type of method heavily depends on the accuracy of the trajectory prediction, and usually serves as the pre-takeoff prediction without considering real-time traffic situations. Although prediction results can be corrected by real-time collected flight positions, the performance also has large deviation with the ground truth since the trajectory prediction for flights cannot capture dynamic features of global traffic situation in time.

b) Time series algorithms: The historical traffic flow of a studied airspace is fed into the algorithm to predict the future traffic flow, such as ARIMA [12,13]. The main drawback of this algorithm is that it ignores the influence of traffic situation in adjacent airspaces. Similarly, the network based algorithm [14] did not pay sufficient attention to the temporal dependencies of historical traffic situations.

c) Probabilistic algorithms: Probabilistic based arriving models for flight or airspace were studied to illustrate stochastic features of the air traffic and further used to predict the traffic flow [15–17]. However, the performance of this method is affected by handcraft models, which fails to depict the real traffic situation fully and further reduces the prediction accuracy.

d) Machine learning algorithms: The support vector machine [18] and artificial neural network [19,20] based algorithms were proposed and improved to predict the air traffic flow. Comparing with our proposal, shallow neural networks were applied in existing approaches, which failed to mine high-level transition patterns of the air traffic. In addition, existing models failed to consider the spatial and temporal dependencies of the traffic situation comprehensively due to their input organization.

From above discussions, we can see that existing approaches for the ATFP are bottom-up ones, which predicts the traffic flow for certain airspace instead of the overall traffic distribution of studied areas. In this paper, we attempt to bridge this gap and provide a more intuitive framework for the ATFP research. With successful applications of deep learning-based model on other fields, it is also introduced to solve the existing problems of transportation research. The stacked auto-encoder based architecture was designed to predict the traffic flow of the next week for given road [21] and other related works can be found in [22,23]. For air traffic research, deep learning based models were developed to forecast workload and airspace configuration in [24] and predict the flight delay [25]. A similar work, i.e., state-image, was studied to the in-flight parameter identification for detection and characterization of the aircraft icing by a neural network-based model [26], in which higher prediction performance was achieved. To our limited knowledge, there is no relevant research that is proposed for the ATFP.

3. Methodologies

3.1. Traffic flow matrix

As the most important part of ATCSs, the radar network monitors and tracks flying aircraft in the air by the primary surveillance radar (PSR), second surveillance radar (SSR) and automatic dependent surveillance-broadcast (ADS-B). Various attributions of aircraft can be recognized by the surveillance equipment, such as SSR code, aircraft identification, position (longitude, latitude and altitude), velocity, heading, climb rate, et al. Different equipment keeps different updating period, typically, 4 seconds for radar and 1 second for ADS-B. All collected information is sent to a fusion center to generate trajectories for all flights with a uniform updating period, 20 seconds in our dataset.

Just as mentioned before, the traffic flow of a given region shares high similarity with its adjacent regions and current traffic flow is also affected by both temporal and spatial dependencies of traffic situation in history. Based on the characteristics that should

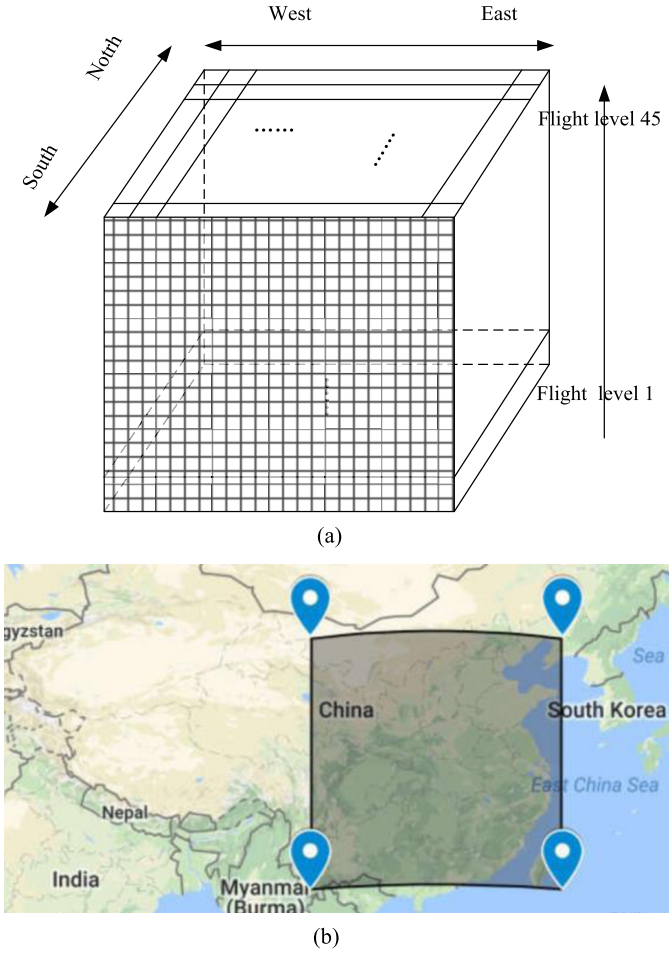

Fig. 2. (a). Gridded map scheme of the TFM. (b). The studied area in this work. This figure is built based on Google Map.

be considered for the ATFP, we present a special data representation called Traffic Flow Matrix (TFM). Gridded map method is firstly used to split the research airspace into different cells with a same size to describe spatial correlations of traffic flow in adjacent areas. Moreover, to illustrate the 3-D feature of the air traffic system, we propose to divide the altitude into flight levels based on the design of CAAC, i.e., 45 flight levels in China. Thus, the air traffic situation in the studied airspace is constructed into a 3-D cube, in which each cell can be regarded as a customized airspace with given spatial area. The customized airspaces are more refined and dedicated than existing ones, which can significantly improve the granularity of air traffic prediction and management. The spatial correlations of traffic flow in local area are captured by the relative position of elements in TFM, just as the pixel in image. Compared the TFM to image, the width and height of TFM determine the size of an image, while the flight level indicates the channel of an image. The traffic situation on different flight level is represented as a channel of an image, in which spatial correlations of traffic flow among adjacent areas are described. The 3-D cube and its cells of the proposed data representation are showed in Fig. 2 (a), whose dimension is $m \times n \times 45$. Each cell is assigned to a series number from $(1, 1, 1)$ to $(m, n, 45)$, just as the index of matrix and the total amount of flights in each cell, i.e., traffic flow, is regarded as an element of the TFM.

The research area in this work comprises of flight hotspot areas in China and is represented as an envelope whose latitude and longitude of upper-left and lower-right corner are (41.0, 100.5) and (22.0, 123.0) respectively, as shown in Fig. 2 (b). The total flights

in this area account for about 84% of China. The spatial extent of the cell (cell size) is a key parameter in this work. A small cell size makes flight transitions among cells more frequently for both adjacent regions and different predicting instants, but it may not be the optimal selection for prediction models and leads to heavy computational complexity. A bigger cell size relieves the computational burden considerably, however, slow flight transitions for both spatial and temporal dependencies may reduce the model performance. Therefore, we must make a trade-off between the model performance and computational burden when selecting the cell size of the TFM. Since the flight execution in China has seasonal period, the air traffic flow shares high similarity among days and weeks. For example, the real-time air traffic flow of this Monday is very similar with that of last Monday. Therefore, we divide the training samples into weeks and use the average flight number of each day in a week to analyze the influence of the cell size on flight transitions. The Singular Value Decomposition (SVD) [27] is a classic algorithm that is proposed to mine similar patterns of cyclical data (day of week in this work). The lower the rank of a TFM, the higher the data correlation among different modes in it. The experiment details will be listed in the later section.

3.2. ConvLSTM

ConvLSTM is a combination block of CNN and LSTM, we introduce the ConvLSTM after the basic CNN and LSTM. CNN is a widely used neural network for spatial modeling in the research of computer vision, whose improved ideas over the common DNN can be summarized as follows:

a) The convolution operation has the invariance over spatial rotation and distortion, which is very crucial for air traffic flow prediction. Since the air traffic flow shows different transition patterns with respect to different cell size in the TFM. CNN can eliminate these effects and improve the prediction performance.

b) The spatial dependency modeling of local receptive field describes transition patterns of air traffic flow among adjacent regions of the studied region.

c) The weights are shared in a same convolution kernel, which greatly reduces the amount of model parameters and improves the applicability of the CNN block.

The mathematical rules of CNN can be expressed as Eq. (1) [28].

$$h_{ij}^{k} = f((W^{k} * x)_{ij} + b_k) \tag{1}$$

where h_{ij}^{k} is the activation of the hidden unit (i, j) in k^{th} CNN layer. $f(\bullet)$ denotes the nonlinear activation function, which can be sigmoid, tanh, ReLU, et al. $W^{\bullet}$ saves the weights between two layers. x is the input of current layer, and $*$ indicates the convolutional operation. b_k is the bias of k^{th} CNN layer.

LSTM is an improved RNN block with four control gates, i.e., input gate, forget gate, cell, and output gate, [29] and hidden units. The general architecture of a single LSTM block is displayed in Fig. 3, and mathematical rules are shown in Eq. (2). The information transmission in LSTM is controlled by the four gates, in which some important inputs even last long can also be considered and used to predict future states. That is to say, LSTM shows how historical inputs determine the output in the future. LSTM has made very excellent progress in speech recognition, video analysis, sequence modeling and other fields.

$$\begin{aligned}
\boldsymbol{I}^t &= f(\boldsymbol{W}_{ix}\boldsymbol{x}^t + \boldsymbol{W}_{ih}\boldsymbol{h}^{t-1} + \boldsymbol{W}_{ic} \circ \boldsymbol{C}^{t-1} + \boldsymbol{b}_i) \\
\boldsymbol{F}^t &= f(\boldsymbol{W}_{fx}\boldsymbol{x}^t + \boldsymbol{W}_{fh}\boldsymbol{h}^{t-1} + \boldsymbol{W}_{fc} \circ \boldsymbol{C}^{t-1} + \boldsymbol{b}_f) \\
\boldsymbol{C}^t &= \boldsymbol{F}^t \circ \boldsymbol{C}^{t-1} + \boldsymbol{I}^t \circ g(\boldsymbol{W}_{cx}\boldsymbol{x}^t + \boldsymbol{W}_{ch}\boldsymbol{h}^{t-1} + \boldsymbol{b}_c) \\
\boldsymbol{O}^t &= f(\boldsymbol{W}_{ox}\boldsymbol{x}^t + \boldsymbol{W}_{oh}\boldsymbol{h}^{t-1} + \boldsymbol{W}_{oc} \circ \boldsymbol{C}^t + \boldsymbol{b}_o)
\end{aligned}$$

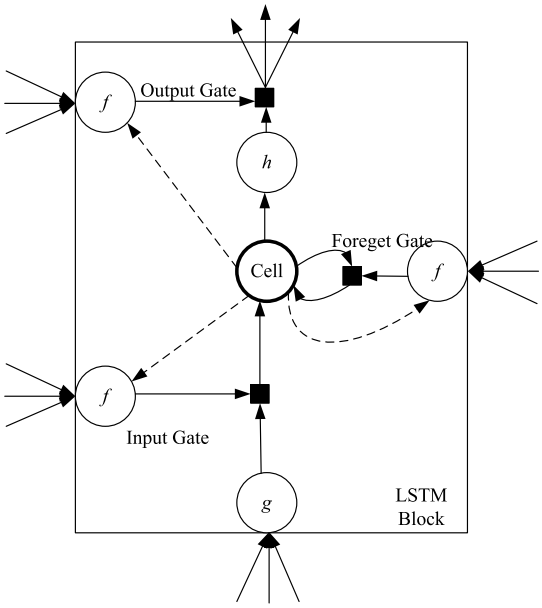

Fig. 3. Structure of the LSTM block.

$$\boldsymbol{h}^t = \boldsymbol{O}^t \circ g(\boldsymbol{C}^t) \tag{2}$$

where all superscripts t denote the predicting time instant. $\boldsymbol{I}^t$, $\boldsymbol{F}^t$, $\boldsymbol{C}^t$, $\boldsymbol{O}^t$ are the activation of input gate, forget gate, cell, and output gate, respectively. $\boldsymbol{h}^t$ denotes the hidden unit of LSTM block. $\boldsymbol{W}_{..}$ saves the weights between two blocks, whose subscripts indicate the direction of information transmission. For example, $\boldsymbol{W}_{fh}$ saves weight parameters for the information transmission between the hidden unit of last time instant and the forget gate at current time. $\boldsymbol{b}_.$ represents the bias in the corresponding gate. $\circ$ is the element-wise product of matrix. $f(\bullet)$ and $g(\bullet)$ denote the sigmoid and tanh activation function respectively.

Combine the CNN with LSTM, a ConvLSTM block is inferred by replacing the full connection operation between cells with convolutional operation to mine the spatial dependency. The inference formula of ConvLSTM is shown in Eq. (3) [30].

$$\begin{aligned}
\boldsymbol{I}^t &= f(\boldsymbol{W}_{ix} * \boldsymbol{x}^t + \boldsymbol{W}_{ih} * \boldsymbol{h}^{t-1} + \boldsymbol{W}_{ic} \circ \boldsymbol{C}^{t-1} + \boldsymbol{b}_i) \\
\boldsymbol{F}^t &= f(\boldsymbol{W}_{fx} * \boldsymbol{x}^t + \boldsymbol{W}_{fh} * \boldsymbol{h}^{t-1} + \boldsymbol{W}_{fc} \circ \boldsymbol{C}^{t-1} + \boldsymbol{b}_f) \\
\boldsymbol{C}^t &= \boldsymbol{F}^t \circ \boldsymbol{C}^{t-1} + \boldsymbol{I}^t \circ g(\boldsymbol{W}_{cx} * \boldsymbol{x}^t + \boldsymbol{W}_{ch} * \boldsymbol{h}^{t-1} + \boldsymbol{b}_c) \\
\boldsymbol{O}^t &= f(\boldsymbol{W}_{ox} * \boldsymbol{x}^t + \boldsymbol{W}_{oh} * \boldsymbol{h}^{t-1} + \boldsymbol{W}_{oc} \circ \boldsymbol{C}^t + \boldsymbol{b}_o) \\
\boldsymbol{h}^t &= \boldsymbol{O}^t \circ g(\boldsymbol{C}^t)
\end{aligned} \tag{3}$$

All notations in Eq. (3) are same with that of in Eq. (2). The extra $*$ is the convolutional operation reflecting the difference between the LSTM and ConvLSTM. In each time instant, spatial correlations of the TFM are extracted by the convolutional block. Similarly, temporal dependencies of the historical TFM sequence are further mined by the LSTM module.

3.3. Model architecture

In order to predict situational air traffic flow, the output of the proposed model in this work is also designed as a TFM to represent the overall air traffic flow distribution, which has the same dimension with the input. Therefore, all convolutional operations in the proposed model are with the same padding mode to keep the output dimension. To fully mine temporal correlations among historical instants, all hidden ConvLSTM layers return a sequence until the final output layer. In addition, to speed

up the training process, the time-distributed batch normalization layer is designed after each ConvLSTM layer. Similarly, we also apply time-distributed dropout layers (after batch normalization layers) to prevent the over-fitting problem. Let sequential layers with (ConvLSTM, batch normalization, dropout) as a group layer, the architecture of the proposed model can be summarized as (input TFM sequence, group layer, ..., group layer, output TFM). Hereafter, without specific instructions, the ConvLSTM layer denotes a group layer with (ConvLSTM, batch normalization, dropout layers).

The timestep of the input sequence, i.e., the look-ahead horizon of the prediction model, always plays a vital role in sequential models, and only most related historical TFMs should be treated as the model input. If a long TFM sequence is taken as the model input, the samples that are far from current instant may have little positive influence on the air traffic flow prediction. More seriously, a long input sequence will explode the amount of model parameters and degenerate the model performance. Conversely, if the input sequence does not take enough historical information (small timestep), the model cannot learn required temporal dependencies to support the task of air traffic flow prediction. Besides the configurations of hidden layers, the timestep of the input sequence is also regarded as a hyperparameter of the proposed model. To obtain the optimal length of the input sequence, several experiments are designed to yield the best model performance.

4. Experiments and discussions

4.1. Data

The raw data in this work is flight trajectories with 20 seconds updating period in China. All discrete positions located in the research area are considered as air traffic flow to generate the TFM at each predicting instant. The flight execution in China mainly follows the flight schedule of CAAC, which comprises of the spring and autumnal version, and all flights circulate weekly in each version of flight schedule. The data we used to train the proposed model was collected by ATCSs from 07/01/2014 to 08/31/2014. There are about 267,840 raw TFM samples generated from the raw data by our proposed data representation. However, based on the fact that all TFMs in a same minute share extreme high similarity, we only extract one sample in every minute to reduce the total data amount (89,280) and further save the computational resource and training time. The data is further divided into three parts randomly: training set (90% of training data), validation set (5% of training data) and test set (10%, remaining 5% data and 5% validation data).

4.2. Experimental design

In order to obtain desired performance, we design corresponding experiments to optimize the model architecture and other hyperparameters. Regarding the proposal in this work, the following parameters of the proposed model should be taken into consideration: the cell size of the gridded map method, the timestep of the input sequence, the number of hidden ConvLSTM layers, and the neuron of each hidden layer. We design different experiments to determine mentioned hyperparameters, in which the predicted air traffic flow is evaluated by given measurements, as shown in (4) and (5). In addition, several experiments with other existing approaches are also designed to compare their prediction performance.

$$f_k = \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=1}^{45} f_k^{ijl},\ p_k = \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=1}^{45} p_k^{ijl}$$

$$e_k = \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=1}^{45} |f_k^{ijl} - p_k^{ijl}| \tag{4}$$

where f_k^{ijl} and p_k^{ijl} are the real and predicted air traffic flow for the cell of the TFM with indices (i, j, l) of k^{th} test sample in test set, respectively. m and n denote the width and height of the TFM respectively. 45 is the number of flight levels designed by CAAC. f_k and p_k are the real and predicted air traffic flow of k^{th} test sample respectively. e_k is the prediction error of k^{th} test sample. All prediction results obtained by different approaches are evaluated by following factors:

$$\mu = \frac{1}{N}\sum_{k=1}^{N} e_k,\ \delta = \left[\frac{1}{N}\sum_{k=1}^{N}(e_k - \mu)^2\right]^{1/2} \tag{5}$$

$$\eta = 100\% \times \frac{1}{N}\sum_{k=1}^{N}\frac{e_k}{f_k} \tag{6}$$

where N saves the total number of test samples. μ and δ are the mean and standard deviation of prediction errors of air traffic flow for all test samples, respectively. η is the absolute mean percentage error of all prediction results.

4.3. Training

All implementations in this work are programmed using Python, and deep learning related codes are implemented based on the open-source framework Keras (2.0.4) with TensorFlow (1.0.0) backend. The input and output of the proposed model are the historical TFM sequence and current TFM respectively. We use the mean squared error to compute the prediction error during the forward inference, and the back-propagation through time [31] algorithm is applied to optimize model parameters to improve the final performance. The optimizer for model training is the Adaptive moment estimation (Adam) with 0.001 initial learning rate, 0.9 beta_1, 0.99 beta_2 and the decay of learning rate is assigned to 0.99. The batch size during the training is set to 32. In order to obtain optimal model parameters, we perform the early-stopping strategy on validation set to control the number of training epochs. In addition, we use the parallel computing of multi-GPU based on the Keras implementation to take full advantages of multi-GPU.

Learning ideas for selecting the filter shape of convolutional kernel from Xception [32], we use filters with different kernel size (1*1, 3*3 and 5*5) to extract spatial correlations of TFM. Generally, we use more 3*3 kernel during training process since the spatial dependency of a cell is related to its surrounding areas directly. The dropout rate is set to 0.25 for all dropout layers in this work. Besides, to reduce the training loss to a certain level as soon as possible, all training samples are sorted by the collecting time of the raw data to keep the relation among sequence in the first training epoch. From the second training epoch, all training samples are shuffled to improve the robustness and generalization of the proposed model.

4.4. Results and discussions

4.4.1. Cell size of gridded map

The cell size not only determines the computational complexity and training time, but also plays an important role on transition patterns of air traffic flow. In this section, all training samples are grouped by the day of the week from Monday to Sunday, and the time interval between two neighboring TFMs during the same day is enlarged to 5 minutes (288 TFM samples per day). To reduce the dimension of the SVD matrix, each TFM is reshaped as a vector with $m \times n$ dimension without considering the influence of

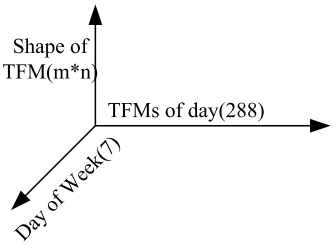

Fig. 4. Data organization for SVD analysis.

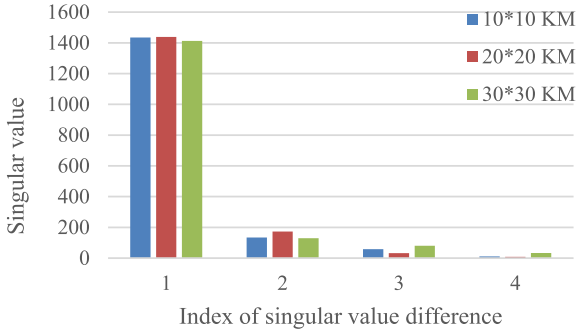

Fig. 5. SVD analysis for flight patterns with different cell size. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

flight levels. Therefore, the data organization for SVD analysis can be described as Fig. 4, based on which we use different cell size (10*10 KM, 20*20 KM, 30*30 KM) to test the optimal selection. We can obtain the shape of the TFM by the cell size and then the SVD algorithm is used to calculate its rank. The TFM shapes are (212, 212, 45), (106, 106, 45), (71, 71, 45) when the cell size is 10*10 KM, 20*20 KM, 30*30 KM respectively.

As we all know, the result of SVD indicates the feature similarity of different modes in a TFM. Sorting singular values in descending order, larger differences among first several singular values indicate that the information in the TFM is more similar and concentrated on some modes. To show the difference of singular values for each configuration of the cell size, we select first 5 singular values of SVD result for each configuration, and the difference between two neighboring singular values are measured to illustrate transition patterns of air traffic flow, as shown in the Fig. 5. From the results we find that flight transitions among days of a week is clearer and more similar when the cell size is 20*20 KM, which is the preferred value for selecting the cell size in this wok. Thus, the cell size of the gridded map is assigned to 20*20 KM in following experiments.

4.4.2. Timestep of input sequence

After the optimal cell size for the proposed data representation is determined in the last experiment, another key parameter of the model input is the timestep of the TFM sequence, i.e., the look-ahead horizon of the prediction model. We save each TFM as an image with single channel (without considering the distribution on flight level) and then replay all of them with a stable frequency. By focusing on the transition patterns of the TFM sequence, we firstly select a rough range of timestep from 6 to 14 with the step-size of 2. Then, we use two types of layer configuration to test the prediction performance with different timestep of the input TFM sequence. The layer configurations used in this section are listed in Table 1.

In this table, # denotes the number of the following item. Since we use the strategy in Xception for the kernel selection, the middle column in the table indicates that three types of convolutional filters, i.e., 32 filters with (3, 3) kernel size, 16 filters with (5, 5) kernel size and 16 filters with (1, 1) kernel size, are applied to mine the spatial dependency. In this section, we use the same kernel configuration for all ConvLSTM layers to reduce the duplication of experiments. Only the mean of prediction errors is used to evaluate the model performance with different number of ConvLSTM layer and input timestep. The experimental results for selecting optimal input timestep are displayed as Fig. 6.

Table 1
Layer configurations for time step selection.

# ConvLSTM layer	Kernel configuration of each layer	#Parameters
2	(32,3,3), (16,5,5), (16,1,1)	$4.35*10^5$
4		$9.23*10^5$

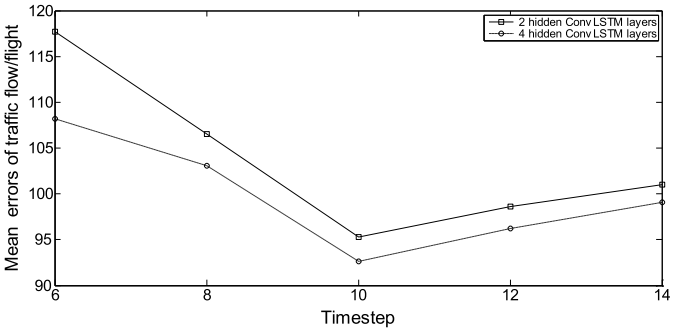

Fig. 6. Mean of prediction errors for timestep selection.

From Fig. 6, we can see that the mean errors of traffic flow prediction for both ConvLSTM layer configurations show a same trend with the increasing of the look-ahead horizon of the prediction model. Specifically, there is an inflection point in the prediction results under different configurations of hidden layer when the length of the input TFM sequence is 10. Regardless of the hidden layer configuration, the mean prediction errors go down when the timestep of the input TFM sequence increases from 6 to 10. With the increasing of the input timestep, the mean prediction errors rise slightly. Therefore, we can come to the conclusion that 10-input timestep is a preferred choice for the input TFM sequence compared to other options. We also find that standard deviations of prediction errors jump up sharply when the input timestep ranges from 8 to 6 and from 12 to 14. When the input timestep is set to 6, the model lacks enough useful information to predict future patterns of air traffic flow. However, when the input timestep comes to 14, the model has accumulated excessive historical information to predict the traffic flow, which in turn degenerates the model performance. That is why we design this experiment to select a best timestep of the input TFM sequence for the proposed model. Moreover, the mean prediction errors with 4 hidden ConvLSTM layers are lower than that of 2 hidden ConvLSTM layers, which proves the advantage of deeper architecture on modeling non-linear features in this application.

4.4.3. Layer configuration of ConvLSTM

In this section, we mainly focus on selecting the optimal network architecture to improve the overall performance of the proposed model, which concerns the number of hidden layers and kernel configurations of each hidden layer. We generally learn some existing models of the ConvLSTM application and adjust some of modules according to task requirements of air traffic flow prediction. The number of hidden ConvLSTM layers is selected from 4 to 6. The kernel size comprises of (1, 1), (3, 3) and (5, 5), just like that of the Xception model. The variation for the kernel configuration is the number of each kernel in hidden layers. Typically, the number of kernels usually is the multiple of 8, such as 8, 16, 32, etc. Different selections for the number of hidden layers

Table 2
Result of layer configuration.

#ConvLSTM Layer	Kernel configuration	μ	δ	η
4	(16, 8, 8)	98.30	17.65	9.91%
4	(32,16,16)	92.60	13.19	9.17%
4	(64,32,32)	86.10	12.49	8.82%
5	(16, 8, 8)	93.40	11.45	9.20%
5*	(32,16,16)	79.30	9.65	6.42%
5	(64,32,32)	82.15	10.93	7.81%
6	(16, 8, 8)	84.33	10.70	8.51%
6	(32,16,16)	81.04	11.30	7.98%
6	(64,32,32)	80.25	11.51	7.44%

and number of kernels are listed in Table 2, in which final measurements under different kernel configuration are also reported. We average 3 times test as final measurements to eliminate the randomness of the parameter initialization during the training of deep learning-based models. The configurations in the second column of the table indicate the number of (3, 3), (1, 1) and (5, 5) kernel. For example, the configuration of (32, 16, 16) indicates that 32 filters with (3, 3) kernel size, 16 filters with (5, 5) kernel size and 16 filters with (1, 1) kernel size are designed in each ConvLSTM layer.

From the table, we can see that the proposed model can obtain better performance with 5 hidden ConvLSTM layers whose kernel configuration is (32, 3, 3), (16, 5, 5), (16, 1, 1) among all listed configurations. With this configuration, the proposed model has a total amount of $1.19^{*}10^{6}$ parameters needs to be optimized. The model architecture is showed in Fig. 7. To simplify the figure, we only describe one of the five ConvLSTM layers (Group Layer), in which the layer type, input and output shape are also displayed. The training process is terminated after 12 training epochs based on the default early-stopping implementation in Keras. The mean prediction errors on validation dataset during model training are drawn as a curve and shown in Fig. 8. Just as mentioned before, the training samples are sorted by their collected time to keep the temporal relations in the first training epoch, which explains the local minimum near the end of first epoch. After this period, the mean of prediction errors on validation dataset jumps to a higher level temporarily and decreases along the training process gradually.

According to experimental results predicted by the proposed model with selected parameters, we further make following explanations on the distribution of prediction errors from the spatial and temporal view to analyze the advantages and disadvantages of the proposed model.

a) By accumulating prediction errors on different cells of the TFM, we find that the mean of prediction errors in cells intersected with trunk routes is lower than that of other cells. In general, the flight density in these areas is higher compared to other areas, and transition patterns of traffic flow are smoother and easy to be captured by the prediction model. The results indicate that the transition patterns of air traffic flow with spatial and temporal correlation are mined by the proposed model successfully.

b) Following a), we also find that the mean prediction errors in cells located by busy airports on the margin of the research area are slightly higher than that of any other cells. This is because the end effect of the internal system (the research area) and there are many external activations affecting transition patterns of the traffic flow in these areas. In this work, flights outside of the research area fly into those cells (busy airports) and impact the prediction performance, such as a large number of international flights fly in and out at Shanghai (airports).

c) By accumulating prediction errors at different time instant in the same day, we find that the mean prediction errors from 10:30 AM to 6:30 PM are more stable than that of other time instants since flights executed regularly during this period in China.

d) Following c), we also find that the standard deviations of prediction errors near 10:00 AM fluctuate in a short term. We attribute this phenomenon to the peak of the flight departure in China.

f) By accumulating prediction errors on different flight levels, we find that prediction errors in low altitude airspaces (<3000 meters) are higher than that of high-altitude airspaces. It is easy to explain that the aircraft is usually in the climb or descent stage when its altitude is lower than 3000 meters. During

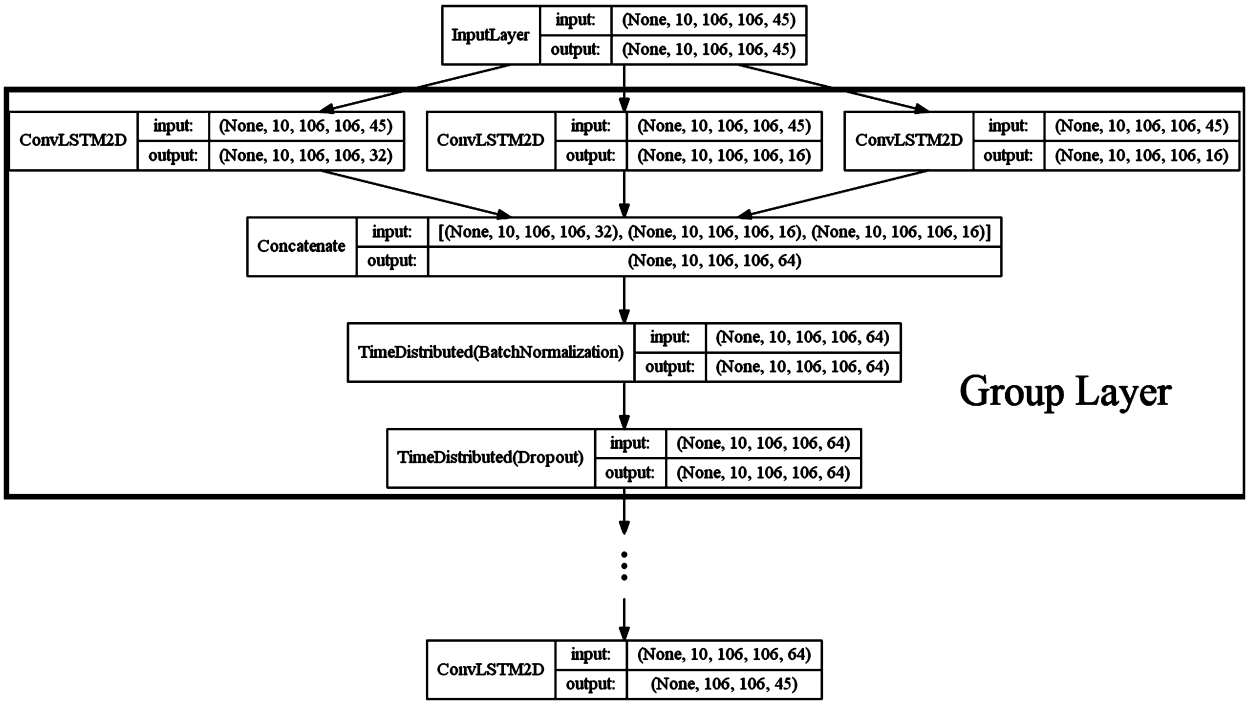

Fig. 7. Architecture of the proposed model.

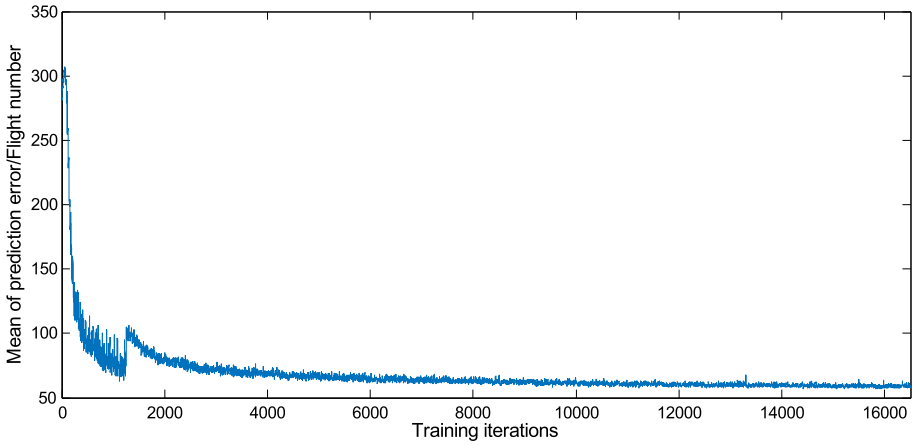

Fig. 8. Validation loss of training process.

Table 3
Comparative results.

Approaches	μ	δ	η
Flight plan based	119.36	21.35	13.24%
Regression model	102.52	13.68	10.15%
DNN	93.21	14.70	9.79%
The proposed model	79.30	9.65	6.42%

this stage, the aircraft has larger maneuverability, which reduces the predictability and results in higher prediction errors.

From the analysis of the prediction error distribution, common intuitions of air traffic flow are proved by experimental results. Therefore, we can see that the spatial and temporal transition patterns of flight flow in air traffic system are indeed learned by the proposed model. The key features of air traffic flow are captured to predict the future air traffic flow. However, for some special situations, such as the traffic flow outside the research area, the transition patterns of flight flow in the low altitude need to be further studied based on other independent features.

4.4.4. Comparative results with existing approaches

In order to show the performance superiority over other existing approaches, we conduct several experiments with following air traffic flow prediction approaches: flight plan-based [10], regression model [12] and shallow NN based model [18]. In these experiments, the mentioned three measurements are applied to evaluate the model performance. Experimental results of comparative methods are reported in Table 3, from which we can see that the proposed model shows the desired superiority over other approaches on both the prediction accuracy and stability.

In addition, the proposed model is a more intuitive and high-level (situational) one, which predicts the traffic flow of the whole research area by taking the spatial and temporal dependencies into consideration. Since the input and output of the proposed model are TFMs, which show the global flight distributions of air traffic situation, we call it as end-to-end model for air traffic flow prediction. The proposed data representation illustrates the global traffic situation comprehensively with the spatial dependencies among adjacent cells and flight levels, which is modeled by the inherent convolutional operation of ConvLSTM block. The temporal dependencies of historical TFMs are mined by LSTM blocks in the proposed model. By taking advantages of the two blocks on modeling spatial and temporal dependencies, the proposed model can obtain a desired prediction performance. Conversely, the flight plan-based approach predicts the pre-takeoff air traffic flow without considering the real-time flight operation, and its correction heavily depends on the accuracy of trajectory prediction. It is a time consuming and bottom-up operation and very sensitive to change of flight patterns, which explains the standard deviation of prediction errors predicted by this method is larger than that of others. The regression model only introduces the influence of historical data (temporal without spatial correlations) which makes the model heavily dependent on data amount and the look-ahead horizon. Besides, the regression model usually serves as the long-term ATFP and is sensitive to sudden changes of air traffic flow, such as the adverse weather, traffic flow control, et al. The Shallow NN based model is a more advanced approach compared to the regression model. It also can process multiple research areas with very limited spatial dependency among adjacent areas. However, the shallow NN based model is not enough to predict the air traffic flow because of the complex management of civil aviation control in China. Only by learning advantages of existing approaches and improving their weakness can we propose this model to achieve the goal of ATFP.

5. Conclusions and future works

For the research of air traffic flow prediction, we apply the machine learning approach to present an end-to-end model in this paper. The proposed model is a deep neural network based one, which was proved to be effective for modeling spatial and temporal dependencies of air traffic. By analyzing special characteristics of air traffic system, we formulate the air traffic flow prediction as a spatiotemporal sequence forecasting problem which is solved based on a CNN and LSTM combined model, i.e., ConvLSTM. A special data representation called TFM is proposed to encode the air traffic situation into the model input, in which spatial dependencies of air traffic flow among adjacent areas and flight levels are represented. The ConvLSTM module not only preserves the advantages of the LSTM block on sequence modeling but is also very suitable for processing spatiotemporal data due to its inherent convolutional structure. Furthermore, the distribution of the traffic flow on different flight levels can greatly enhance available measures for ATC. By integrating the ConvLSTM with Batch Normalization and dropout layers, we build an end-to-end trainable model for air traffic flow prediction which can predict the air traffic in the situational level. Finally, the superiorities of accuracy and stability over existing approaches are proved by our designed experiments on real operating data.

In the future, we will continue to optimize the model architecture to improve the prediction performance. Another work may be concentrated on the modeling of out-system activations, such as the peaking hour in the morning or the influence of international flights.

Conflict of interest statement

The authors declare no conflict of interest.

Acknowledgements

This work was supported by the National Key Scientific Instrument and Equipment Development Project of China (No. 2013YQ490879) and the National Natural Science Foundation of China (Grant No: 71573184).

References

[1] C. Wanke, M. Callaham, D. Greenbaum, A. Masalonis, Measuring uncertainty in airspace demand predictions for traffic flow management applications, in: AIAA Guidance, Navigation, and Control Conference and Exhibit, American Institute of Aeronautics and Astronautics, Reston, Virigina, 2003, pp. 1–11.
[2] C.-L. Wu, R.E. Caves, Research review of air traffic management, Transp. Rev. 22 (2002) 115–132, https://doi.org/10.1080/01441640110074773.
[3] S.H. Hosseini, B. Moshiri, A. Rahimi-Kian, B.N. Araabi, Traffic flow prediction using MI algorithm and considering noisy and data loss conditions: an application to minnesota traffic flow prediction, PROMET – Traffic Transp. 26 (2014) 393–403, https://doi.org/10.7307/ptt.v26i5.1429.
[4] L. Yang, M. Hu, S. Yin, H. Zhang, Characteristics analysis of departure traffic flow congestion in mega-airport surface, Hangkong Xuebao/Acta Aeronaut. Astronaut. Sin. 37 (2016) 1921–1930, https://doi.org/10.7527/S1000-6893.2015.0280.
[5] H. Jeung, H.T. Shen, X. Zhou, Mining trajectory patterns using hidden Markov models, in: Data Warehousing and Knowledge Discovery, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 470–480.
[6] J. Schmidhuber, Deep learning in neural networks: an overview, Neural Netw. 61 (2015) 85–117, https://doi.org/10.1016/j.neunet.2014.09.003.
[7] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997) 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735.
[8] F. Karim, S. Majumdar, H. Darabi, S. Chen, LSTM fully convolutional networks for time series classification, IEEE Access 6 (2018) 1662–1669, https://doi.org/10.1109/ACCESS.2017.2779939.
[9] Y. Lin, J. Zhang, H. Liu, An algorithm for trajectory prediction of flight plan based on relative motion between positions, Front. Inf. Technol. Electron. Eng. 19 (2018) 905–916, https://doi.org/10.1631/FITEE.1700224.
[10] S. Chen, Short term air traffic flow prediction based on real time data of Eurocat-X system, Inf. Commun. 05 (2013) 42–44.
[11] M. Hu, Theory and Method of Air Traffic Flow Management, 1st ed., Science Press, Beijing, 2010.
[12] E. Cadenas, W. Rivera, R. Campos-Amezcua, C. Heard, Wind speed prediction using a univariate ARIMA model and a multivariate NARX model, Energies 9 (2016) 109, https://doi.org/10.3390/en9020109.
[13] S. Mehrmolaei, M.R. Keyvanpour, Time series forecasting using improved ARIMA, in: 2016 Artificial Intelligence and Robotics (IRANOPEN), IEEE, 2016, pp. 92–97.
[14] D. Chen, M. Hu, Y. Ma, J. Yin, A network-based dynamic air traffic flow model for short-term en-route traffic prediction, J. Adv. Transp. 50 (2016) 2174–2192, https://doi.org/10.1002/atr.1453.
[15] Y. Bie, H. Zhang, Z. Zhang, Method of short-term probability prediction for air traffic network flow, Aeronaut. Comput. Tech. 43 (2013) 29–32.
[16] X. Li, Prediction model to air traffic flow of Kunming airspace title, J. Civ. Aviat. Univ. China 30 (2012) 5–8.
[17] C. Wang, L. Yang, Probabilistic methods for airspace sector flow and congestion prediction, J. Southwest Jiaotong Univ. 46 (2011) 162–166.
[18] R. Geng, D. Cui, B. Xu, Support vector machine based combinational model for air traffic flow prediction, J. Tsinghua Univ. 48 (2008) 1205–1208.
[19] D. Cui, S. Wu, B. Xu, Air traffic flow forecasts based on artificial neural networks combined with regression methods, Qinghua Daxue Xuebao/J. Tsinghua Univ. 45 (2005) 96–99.
[20] F. Qiu, Y. Li, Air traffic flow of genetic algorithm to optimize wavelet neural network prediction, in: 2014 IEEE 5th International Conference on Software Engineering and Service Science, IEEE, Beijing, China, 2014, pp. 1162–1165.
[21] Y. Lv, Y. Duan, W. Kang, Z. Li, F.Y. Wang, Traffic flow prediction with big data: a deep learning approach, IEEE Trans. Intell. Transp. Syst. 16 (2014) 865–873, https://doi.org/10.1109/TITS.2014.2345663.
[22] N.G. Polson, V.O. Sokolov, Deep learning for short-term traffic flow prediction, Transp. Res., Part C, Emerg. Technol. 79 (2017) 1–17, https://doi.org/10.1016/j.trc.2017.02.024.
[23] H. Yi, H. Jung, S. Bae, Deep neural networks for traffic flow prediction, in: 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), IEEE, 2017, pp. 328–331.
[24] D. Gianazza, Forecasting workload and airspace configuration with neural networks and tree search methods, Artif. Intell. 174 (2010) 530–549, https://doi.org/10.1016/j.artint.2010.03.001.
[25] Y.J. Kim, S. Choi, S. Briceno, D. Mavris, A deep learning approach to flight delay prediction, in: 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), IEEE, 2016, pp. 1–6.
[26] Y. Dong, An application of deep neural networks to the in-flight parameter identification for detection and characterization of aircraft icing, Aerosp. Sci. Technol. 77 (2018) 34–49, https://doi.org/10.1016/j.ast.2018.02.026.
[27] D. Kalman, A singularly valuable decomposition: the SVD of a matrix, Coll. Math. J. 27 (1996) 2, https://doi.org/10.2307/2687269.
[28] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Backpropagation applied to handwritten zip code recognition, Neural Comput. 1 (1989) 541–551, https://doi.org/10.1162/neco.1989.1.4.541.
[29] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM networks, in: Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005, IEEE, 2005, pp. 2047–2052.
[30] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, Wang-chun Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting, in: Advances in Neural Information Processing Systems, Montreal, Canada, 2015, pp. 1–9.
[31] P.J. Werbos, Backpropagation through time: what it does and how to do it, Proc. IEEE 78 (1990) 1550–1560, https://doi.org/10.1109/5.58337.
[32] F. Chollet, Xception: deep learning with depthwise separable convolutions, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017, pp. 1800–1807.