Engineering Applications of Artificial Intelligence 126 (2023) 107012

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Deep learning-powered vessel traffic flow prediction with spatial-temporal attributes and similarity grouping

Yan Li [a,1], Maohan Liang [b,1], Huanhuan Li [c,*], Zaili Yang [c,**], Liang Du [d], Zhongshuo Chen [e]

[a] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China
[b] Department of Civil and Environmental Engineering, National University of Singapore, Singapore
[c] Liverpool Logistics, Offshore and Marine (LOOM) Research Institute, Liverpool John Moores University, Liverpool, UK
[d] School of Civil and Environmental Engineering, Nanyang Technological University, Singapore
[e] School of Intelligent Finance and Business, Xi'an Jiaotong-Liverpool University, Suzhou, China

ARTICLE INFO

Keywords:
Vessel traffic flow prediction
Automatic identification system (AIS)
Deep learning
Spatial view
Temporal view
Similarity grouping view

ABSTRACT

Perceiving the future trend of Vessel Traffic Flow (VTF) in advance has great application values in the maritime industry. However, using such big data from the Automatic Identification System (AIS) for accurate VTF prediction remains challenging. Deep training networks can learn valuable features from extensive historical data. This paper proposes a new learning-based prediction network, improved Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) with similarity grouping, including three views. To effectively enable the training network to capture the temporal and periodic (i.e. a spatial attribute) change characteristics of VTF, the CNN and LSTM are employed to compose spatial and temporal views, respectively. Hence, the original one-dimensional data is transformed into a matrix (hour of the day × day) to adapt the input of the proposed methodology. In practical applications, VTF of multiple adjacent target regions need to be predicted simultaneously, and the changes of VTF in different areas may influence each other. To explore their hidden relationships, the similarity grouping view aims to find the target area that exhibits the most similarity with the VTF change trend of the current research area. Furthermore, similar information is combined with the features generated from the other two views to obtain the prediction results. In summary, the new advantage lies in mining the spatiotemporal attributes of data and fusing the similarity information of adjacent regions. Comparative experiments with eleven other methods on realistic VTF datasets show that the proposed method demonstrates superior prediction accuracy and stability performance.

1. Introduction

The ongoing progress of economic globalisation has led to an increase in the frequency of import and export trade between countries (Li and Yang, 2023; Xin et al., 2023a). Currently, shipping, air, railway, and pipeline transportation are the main modes of transportation for such transactions (Xiao et al., 2022). However, railway transportation is only feasible between a few countries (Mlepo, 2022). Meanwhile, air and pipeline transportation incur high costs and have limited capacity for carrying certain types of goods (Hummels, 2007). Facilitated by the low transportation cost and a wide variety of goods, the maritime industry has received the favour of all nations and accounts for about 80% of the total international trade (Li et al., 2022; Li and Lam, 2017; Millefiori et al., 2016). Following the frequent maritime trade, a sharp increase in the number of vessels in the channel and port waters has exposed potential navigation and surveillance risks (Xin et al., 2023b). Hence, how to effectively predict risks and ensure the navigation safety of vessels is an imperative issue in water transportation research.

Vessel Traffic Flow (VTF) provides a quantitative assessment of the level of activity in a targeted maritime area, such as a channel or port waters. It is defined as the total number of vessels passing through a specific location within a given time unit (Xiao et al., 2023). The high density of vessels indicates the VTF data of a target area increases suddenly or is higher than the traditional historical value, increasing the collision risk of vessels or causing channel congestion. Hence, it is urgent to carry out research on VTF in the field of water transportation. As an increasing number of scholars delve deeper into VTF research and data

* Corresponding author.
** Corresponding author.
E-mail addresses: H.Li2@ljmu.ac.uk (H. Li), Z.Yang@ljmu.ac.uk (Z. Yang).
[1] Equal contribution.

https://doi.org/10.1016/j.engappai.2023.107012
Received 20 May 2023; Received in revised form 11 August 2023; Accepted 17 August 2023
Available online 22 August 2023

Nomenclature roman letters

Variable	*Definition*
AIS	Automatic Identification System
ARIMA	Autoregressive Integrated Moving Average Model
ARIMA-GARCH	ARIMA with Generalized Autoregressive Conditional Heteroscedasticity
AFSA-SVM	SVM with Artificial Fish Swarm Algorithm
AOA-SVM	SVM with Arithmetic Optimisation Algorithm
AE	Auto Encoder
BFM	Bayesian forecasting model
BPNN	Back Propagation Neural Network
Bi-LSTM	Bidirectional LSTM
Bi-GRU	Bidirectional GRU
CNN	Convolutional Neural Network
CcatCSAGA	Chaotic Cloud-Simulated Annealing Genetic Algorithm
CJP	Chengshan Jiao Promontory
DL	Deep Learning
DBN	Deep Belief Network
DTW	Dynamic Time Warping
ENN	Elman Neural Network
EMD	Empirical Mode Decomposition
EEMD	Ensemble EMD
FNN	Fuzzy Neural Network
GM	Grey Theory-based Models
GWO-SVM	SVM with Grey Wolf Algorithm
GRNN	Generalized Regression Neural Network
GAN	Generative Adversarial Network
GRU	Gate Recurrent Unit
GCNN	Graph Convolution Neural Network
GANet	Graph Attention Network
GAE	Graph Auto-Encoder
HMM	Hidden Markov Model
ICLSGNet	Improved CNN-LSTM Network with A Similarity Grouping
IBCM-DL	Improved Bayesian Combination Model with Deep Learning
KF	Kalman Filtering
KARIMA	Kohonen-ARIMA
KPCA	Kernel Principal Component Analysis
LSTM	Long Short-Term Memory
ML	Machine Learning
MM	Markov Model
MSP-STTN	MultiSize Patched Spatial-Temporal Transformer Network
MSE	Mean Square Error
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
NN	Neural Networks
OCPB	Online Change-Point-Based
PSO-BP	Particle Swarm Optimisation-Back Propagation
RF	Random Forest
RNNs	Recurrent Neural Networks
RBM	Restricted Boltzmann Machine
RSVR	Robust V-Support Vector Regression Model
RBFNN	Radial Basis Function Neural Network
ReLu	Rectified Linear Unit
REMean	Mean Relative Error
REStd	Standard Deviation of The Relative Error
RMSE	Root Mean Square Error
SVM	Support Vector Machine
SMA-SVM	SVM with Slime Mold Algorithm
Seq2Seq	Sequence to Sequence
TF	Traffic Flow
VTF	Vessel Traffic Flow
WNN	Wavelet Neural Network
WOA-SVM	SVM with Whale Optimisation Algorithm

mining technology continues to advance, the research directions of VTF are expanding, mainly including VTF prediction (Liang et al., 2022; Zhao et al., 2022a, 2022b), simulation (Rahimikelarijani et al., 2018; H. Zhang et al., 2019), time series analysis (i.e. a study of VTF characteristics) (Yu et al., 2020; Du et al., 2022; Zhang et al., 2019), data feature extraction (Gao and Shi, 2019; Rong et al., 2022), etc. VTF prediction research can forecast the changes in vessel density within a particular water area to judge whether congestion may occur and increase the probability of collision risk. Hence, this study will concentrate on VTF prediction and establish a methodology with high prediction accuracy and stability. The future VTF data is closely linked to the changing characteristics of historical data, highlighting the prevalence of data-driven prediction methods as the current mainstream research direction. As the automatic identification system (AIS) continues to develop and become mandatory, large-scale AIS data from terrestrial, satellite, and ad hoc networks can be collected (Li et al., 2023). As illustrated in Fig. 1, AIS data (i.e. time stamp, vessel trajectory, course of ground, speed of ground, etc.) is transmitted to the ground servers. It can aid in gathering statistical VTF data that provides crucial data for prediction research (Kim, 2021). According to the relevant definitions of VTF, when a vessel's trajectory crosses a research area within a certain period, the count of VTF in that area increases by one.

Data-driven VTF prediction methods are mainly divided into two categories: modelling-based and learning-based prediction methods. Modelling-based prediction methods typically employ traditional machine learning (ML) techniques (Zhao et al., 2022a, 2022b), while learning-based prediction methods basically utilise neural networks (NN) (Do et al., 2019) and deep learning (DL) methods (Gao et al., 2023a,b; Zhou et al., 2020). VTF data is usually non-stationary and irregular over time, which can pose challenges for prediction research, especially during the period of abnormal change (e.g. COVID-19 pandemic) (Zhao et al., 2022a, 2022b). Modelling-based methods (i.e. traditional ML methods) and traditional learning-based methods (i.e. NN methods) are challenging to accurately learn these irregular transformation features. DL methods have shown promise in effectively capturing these changes (Chai et al., 2021; Weerakody et al., 2021). VTF is a type of time series data, which is why Recurrent Neural Networks (RNNs) are commonly used for prediction tasks (Yao et al., 2018). However, VTF data also has a time attribute and a period attribute that changes over time, meaning that the VTF information in a certain period of a day is related to the VTF in the adjacent period and the same period of the past day or the coming day. Meanwhile, multiple channels are distributed in real areas, which can mutually influence each other, as depicted in Fig. 2. The interactive nature of VTF trend variation across eight channels is significant. For instance, variations in the VTF of the fourth and sixth channels presented in Fig. 2, whether they increase or decrease, can have an impact on the VTF dynamics of the eighth one. This is primarily due to the fact that the VTF from these two channels is directed towards the eighth channel. Similarly, sudden alterations in the VTF of the seventh channel can influence the third and fifth ones. This is attributed to the fact that vessels departing from the seventh channel often need to navigate into either the third or fifth one, highlighting the interconnectedness among these three channels. To enhance prediction accuracy, the training network's design should take into account the

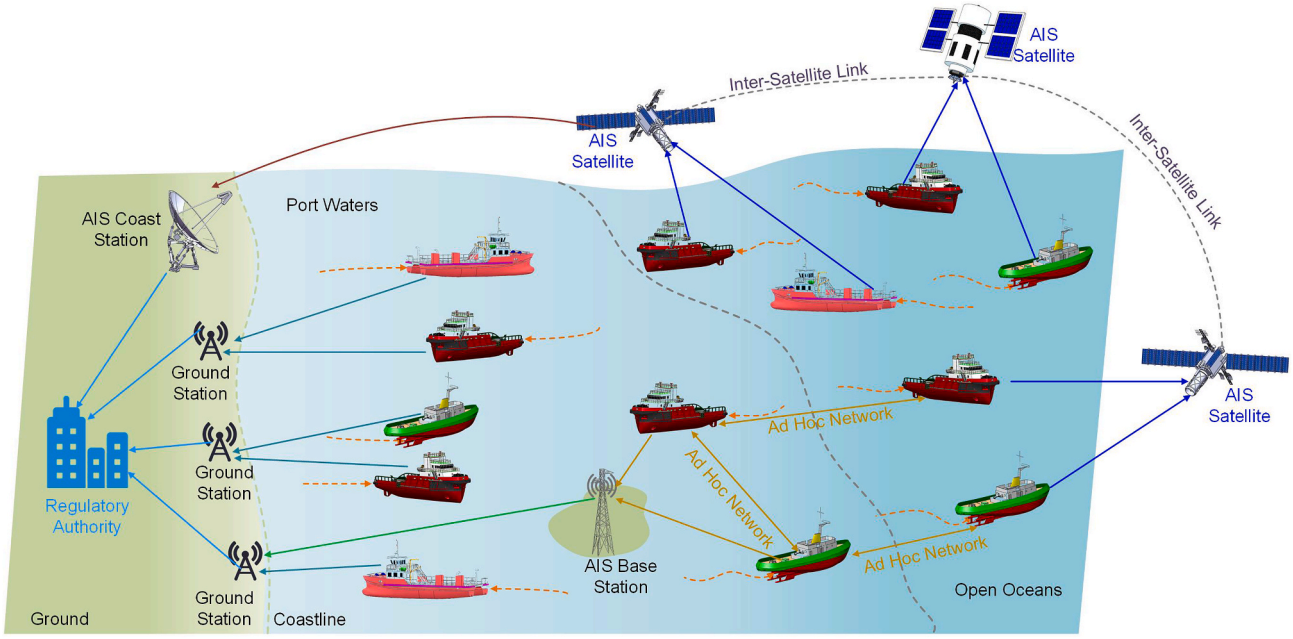

Fig. 1. The illustration of terrestrial, satellite, and ad hoc AIS networks.

interplay of VTF information across various channels. To address this problem, a Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) based interactive spatiotemporal prediction approach, named Improved CNN-LSTM network with a Similarity Grouping (ICLSGNet), is proposed for VTF prediction in multiple channels. It can not only solve the VTF prediction problem with multiple channels in the target area, but also measure the information exchange among the channels with high similarity.

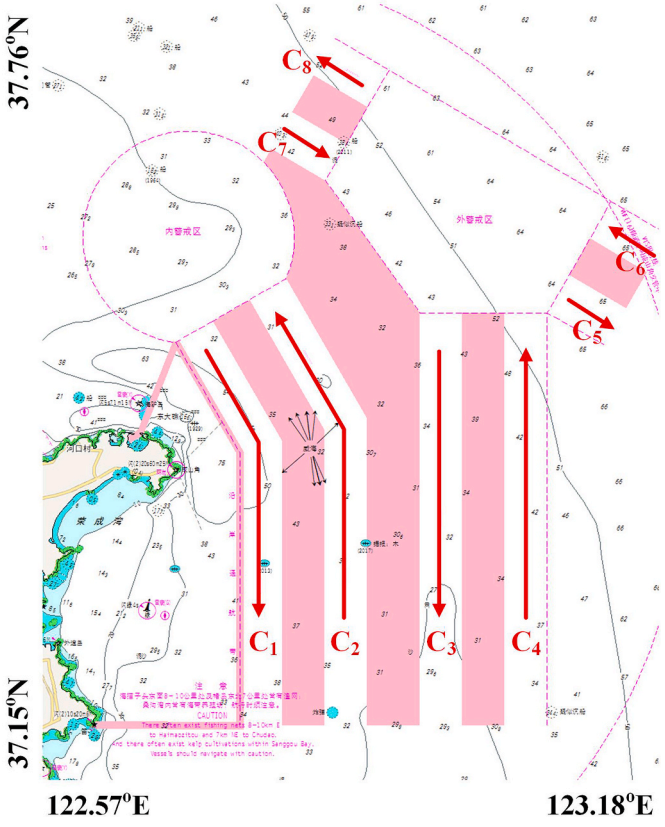

Fig. 2. Schematic diagram of channel distribution (i.e. eight channels C_1 to C_8) in Chengshan Jiao Promontory.

Since VTF prediction methods are primarily derived from road traffic flow (TF) prediction, section 2 summarises the advanced prediction methods of TF, outlines the prediction methods of VTF, and presents the relevant contributions of this study. Section 3 details the framework and implementation process of the proposed ICLSGNet prediction method. Section 4 mainly focuses on verifying and analysing comparative experiments in two parts. The first part examines the impact of different network parameters on the prediction performance of ICLSGNet, while the second one involves a comparative analysis with other commonly used prediction methods. Section 5 concludes this paper with a summary of the relevant research and the future sustainable work direction.

2. Literature review

Numerous techniques for predicting VTF have been adapted from the conventional road TF research methods. This section begins by outlining the prediction methods employed in road TF (i.e. Section 2.1) and then expands to introduce methods used in VTF prediction research (i.e. Section 2.2). The future development trend of TF is closely related to the evolving features of historical data, and as a result, the literature review is mainly data-driven prediction methods. Section 2.3 highlights the research contributions of this paper.

2.1. Overview of traffic flow prediction methods

This section mainly divides traditional TF prediction methods into modelling-based and learning-based methods. Modelling-based prediction methods refer to the traditional ML methods, while learning-based prediction methods are dominated by NN and DL methods.

2.1.1. Modelling-based traffic flow prediction

The primary approach taken by most modelling-based methods in TF prediction research is to assume that future data change trends are similar to historical data and to predict future TF data by fitting the characteristics of historical data. Simple linear regression models (Rath et al., 2020), such as Least Square Method (Zhang et al., 2020), Ridge Regression (Hazarika et al., 2021), and Quantile Regression (K. Wang et al., 2022), are the earliest model-based prediction method applied to TF. These models assume that the change characteristics of historical data can be represented by linear laws. However, TF data is often subject

to external disturbances and irregular changes, making it challenging to fit the change characteristics of historical data by simple linear regression models accurately. To improve the accuracy of historical data, many other modelling-based methods have been studied in the literature, including Autoregressive Integrated Moving Average Model (ARIMA) (Rubi et al., 2022), Kalman Filtering (KF) (Gu et al., 2021), Support Vector Machine (SVM) (Yu et al., 2023), Markov Model (MM) (Fu et al., 2022), Grey Theory-based Models (GM) (Zhu et al., 2022), Bayesian Forecasting Model (BFM) (Chen and Sun, 2022), and Random Forest (RF) (Evans et al., 2019). These methods can effectively identify periodic changes in TF data and have better accuracy than simple linear regression models. SVM is a common and typical ML method that adopts the structural risk minimisation principle and performs well in solving the learning problem of small samples. ARIMA combines the sliding average and the autoregressive models to fit data accurately with periodic solid attributes. KF can continuously update the estimation of state variables, thus improving data prediction accuracy (Okutani and Stephanedes, 1984). Hidden Markov Model (HMM) (Vuković et al., 2015) is the most basic and classic extended version of MM, a probability model about time series, whose model parameters are elementary. The essence of GM is to treat the discrete data scattered on the time axis as a set of continuously changing sequences. By accumulating and subtracting data over time, GM constructs a continuous differential equation that can be used for prediction purposes. GM can achieve accurate short-term predictions with relatively few data samples. However, it is not suitable for processing irregular data. In particular, GM (1,1) (Liu et al., 2014) is the most commonly used method in GM. BFM is a prediction method based on Bayesian statistics. Its biggest feature is that it not only uses data and model information, but also can effectively mine prior knowledge in the implementation prediction tasks. In recent years, some scholars have combined Bayesian statistics with deep learning to solve the problem of data prediction and obtain considerable results. For example, Gu et al. (2020) proposed an improved Bayesian combination model with deep learning (IBCM-DL) to predict short-term TF. The core of this method is divided into two steps. Firstly, it involves constructing the IBCM framework. Secondly, it utilises correlation analysis to investigate the relationship between historical and current TF changes. Similarly, Pang et al. (2021) constructed a Bayesian deep learning framework to predict aircraft trajectory data, considering the influence of convective weather in the training network. RF entails the creation of a forest in a random manner. The forest consists of multiple independent decision trees. For the sample data to be predicted, it is inputted into all decision trees to generate the prediction results for each decision tree. The final prediction result is determined by a majority vote. While this method boasts high predictive accuracy and robustness, it may be susceptible to overfitting if the data exhibits complex patterns.

Numerous scholars have explored extensional versions of modelling-based methods to enhance the accuracy of predicting future TF data. Taking the SVM as a typical example, the primary strategy of the improved method is to optimise SVM parameters. Various developed algorithms, such as SVM with Whale Optimization Algorithm (WOA-SVM) (D. Kong et al., 2020), SVM with Grey Wolf Algorithm (GWO-SVM) (Li et al., 2020), SVM with Artificial Fish Swarm Algorithm (AFSA-SVM) (Kou et al., 2020), SVM with Slime Mold Algorithm (SMA-SVM) (Zhao et al., 2023), SVM with Arithmetic Optimization Algorithm (AOA-SVM) (Chen et al., 2022), have been proposed to limit the range of parameters to be optimised and set the fitness function to find the best parameters. Furthermore, there are also some improved versions of the ARIMA method to solve complex data prediction problems, such as Kohonen-ARIMA (KARIMA) (Van Der Voort et al., 1996), subset ARIMA (Lee and Fambro, 1999), seasonal ARIMA (Williams and Hoel, 2003), ARIMA with Generalized Autoregressive Conditional Heteroscedasticity (ARIMA-GARCH) (Chen et al., 2011), and Online Change-Point-Based model (OCPB) (Comert and Bezuglov, 2013). Through continuous improvements to the modelling-based method, many scholars have successfully enhanced prediction accuracy.

2.1.2. Learning-based traffic flow prediction

Modelling-based prediction methods face challenges in accurately fitting historical data due to the significant volatility of TF. Consequently, the prediction accuracy is reduced. The emergence of NN techniques offers a partial solution by constructing a training network that learns the data change characteristics of historical TF. It involves iterating through the optimiser to obtain better weight and bias values, thereby completing the data prediction task. Back Propagation Neural Network (BPNN) (Wu et al., 2023) is a typical representative of NN, which comprises three parts (i.e. input, hidden, and output layers). Many other NN methods are based on BPNN extension and optimisation, such as Wavelet Neural Network (WNN) (Su et al., 2023), Generalized Regression Neural Network (GRNN) (Safari, 2019), Fuzzy Neural Network (Xu et al., 2021) (FNN) (Fei and Liu, 2022), and Elman Neural Network (ENN) (Xu et al., 2021). In particular, the network structure of WNN is similar to that of BPNN, which only employs the wavelet basis function as the activation function in the hidden unit.

TF data is susceptible to external factors, making it challenging for NN methods to accurately learn historical data changes due to sudden increase or decrease in TF. The emergence and development of DL technology can excavate the change characteristics of irregular historical data by building a deep training network, which was first proposed by Hinton et al. (2006). Currently, there are three categories of commonly used DL methods for time series (i.e. TF and VTF) prediction: discriminative, generative, and hybrids-based DL (Han et al., 2021a,b). Discriminative methods are centred around feeding a set of sample data into a network and obtaining the output value through repeated iteration and optimisation of weight and bias values. The output value is then compared to the actual target value to determine the degree of consistency or difference between them. RNN (Zhang et al., 2023) and Auto Encoder (AE) (Fu et al., 2021), as two leading representatives of discriminative methods, can be used to solve TF prediction problems. The generative methods mainly take into account the joint probability distribution of observation and target data, including the Restricted Boltzmann Machine (RBM) (Hranisavljevic et al., 2020), Deep Belief Network (DBN) (Yang et al., 2021), and Generative Adversarial Network (GAN) (Zhong et al., 2023). Hybrids-based methods are essentially a combination of two or more DLs, such as CNN with LSTM (called CNN-LSTM) (Vidya and Sasikumar, 2022) and LSTM-GAN-AE (Liu et al., 2022). TF is a type of time series data that utilises RNN for prediction tasks (Lee and Ku, 2022). However, RNN is susceptible to gradient disappearance and explosion during training, which has been addressed through the emergence of LSTM (Vatsa and Hati, 2022) and Gate Recurrent Unit (GRU) (Yao and Ge, 2023). The network unit of LSTM consists of three gate structures: forget, input, and output gates. GRU is simplified compared to LSTM in network structure, which only includes two gate structures (i.e. reset and update gates). Bidirectional LSTM (Bi-LSTM) (Bi et al., 2023) and bidirectional GRU (Bi-GRU) (W. Wang et al., 2022) are optimized networks based on LSTM and GRU, respectively. Their networks fuse the forward and reverse information of input data based on two hidden states. However, RNN and its variants have a drawback: all datasets input into the layer must be of uniform length. Fortunately, the advent of Sequence to Sequence (Seq2Seq) offers a solution to addressing this issue, leveraging the Encoder-Decoder framework. The encoder converts input data into a fixed-length vector, and then the decoder generates output data from this vector (Sutskever et al., 2014). In particular, both encoder and decoder can be implemented by using RNN and its advanced versions (i.e. LSTM, GRU, Bi-LSTM, and Bi-GRU). Cao et al. (2022) proposed a spatiotemporal Seq2Seq method capable of mining heterogeneous and time-varying spatial attributes in data and capturing periodic temporal attributes, thereby improving TF prediction accuracy. Hao et al. (2019) proposed a Seq2Seq method based on an attention mechanism to predict short-term subway passenger flow. This method can effectively solve the long dependency problem in network training and can be extended to other fields or application scenarios.

In recent years, scholars have deeply integrated Seq2Seq and attention mechanism, giving rise to a novel network architecture known as Transformer (Han et al., 2021a,b), which is extensively utilised in TF prediction research. For instance, Wen et al. (2023) put forth a new prediction method by the Transformer framework called RPConvformer. The encoder and decoder in this method are respectively responsible for extracting the variation characteristics in historical data and predicting future states, mining the temporal features of traffic data through multiple attention mechanisms. Xie et al. (2022) put forward the MultiSize Patched Spatial-Temporal Transformer Network (MSP-STTN) capable of simultaneously solving short-term and long-term personnel flow prediction problems.

In practical application scenarios, many studies go beyond predicting TF in a single region, requiring concurrent prediction of TF in multiple regions. There may exist certain relationships between the TFs in each region, and these internal connections can be reflected based on the graph theory. Hence, some scholars combine graph and DL methods to uncover hidden relationships between TF in various regions and perform prediction, such as Graph Convolution Neural Network (GCNN) (Pope et al., 2019), Graph Attention Network (GANet) (Veličković et al., 2018), Graph Auto-Encoder (GAE) (Do et al., 2020). These classic graph-related networks have been widely used to address TF prediction issues. Djenouri et al. (2023) utilised scalable GCNN to predict TF and achieved accurate and stable results. Ali et al. (2022) proposed a dynamic deep spatio-temporal neural network abbreviated as DHSTNet for predicting personnel flow. Moreover, they combined LSTM and GCNN with DHSTNet to obtain a new architecture called GCN-DHSTNet, which can effectively capture the spatial and short-term temporal characteristics of data. Y. Wang et al. (2022) put forward an attention-based spatiotemporal GANet, abbreviated as ASTGAT. This new framework can effectively tackle issues of excessive smoothing and network degradation, and can deeply explore the spatiotemporal attributes of data. X. Kong et al. (2020) explored an end-to-end DL dual path method called spatial-temporal GANet. The advantage of this method is that it can handle any graph structure, making input data more flexible. It can also effectively process long time series and solve long-term dependency problems.

2.2. Overview of vessel traffic flow prediction methods

VTF prediction involves applying the general concept of TF prediction, commonly used in road traffic, to maritime transportation. VTF prediction methods mostly draw from road traffic and can be categorised as either modelling-based or learning-based methods. For example, He et al. (2019) proposed a new short-term Kalman model combining regression analysis and KF to predict VTF in China's inland rivers (i.e. Wuhan Yangtze River Bridge and the Second Yangtze River Bridge in Wuhan). Tang et al. (2019) proposed a combined method based on RF and Bayesian networks to predict the level of vessel collision accidents. The core idea of this method is to use RF to identify factors that affect the prediction of vessel collision accident levels. The identified results serve as nodes in the Bayesian network. Yu et al. (2018) discussed the prediction effect of vessel arrivals based on three standard data mining methods (i.e. BPNN, classification and regression tree, and random forest). Liu et al. (2017) restructured the original one-dimensional VTF data into a two-dimensional matrix (month × year) and applied the non-convex low-rank plus sparse decomposition method to the matrix, resulting in low-rank and sparse matrices. Specifically, a low-rank matrix is defined as a trend term reflecting consistent data changes, while a sparse matrix represents a volatility term, which frequently undergoes significant fluctuations due to sudden impacts on VTF. These two matrices were then converted into one-dimensional sequence data, with ARIMA and WNN used for their respective predictions. The final prediction was produced by summing the predicted low-rank and sparse components. Li et al. (2015) optimised the parameters of a robust v-support vector regression model (RSVR) based on their proposed chaotic cloud-simulated annealing genetic algorithm (CcatCSAGA). Then, the kernel principal component analysis (KPCA) method determines the final input vector based on the candidate input variables. Haiyan and Youzhen (2015) combined Radial Basis Function Neural Network (RBFNN), grey prediction model, and autoregressive model with SVM to predict VTF data in the Yangtze River basin. The advantage of this combined prediction method is to avoid the uncertainty of a single method that affects prediction accuracy and stability. With the emergence of NN and DL, scholars have begun to investigate learning-based prediction methods. Xiao et al. (2023) proposed a hierarchical prediction method that considers weather factors, and then utilised GRU and Seq2Seq for prediction. Li et al. (2023) introduced a spatiotemporal GNN strategy. This approach excels in extracting the spatiotemporal fluctuations of VTF across distinct port areas by utilising the graph attention network along with an extended causal convolution framework, thereby enhancing prediction accuracy. Li and Ren (2022) proposed an Encoder-Decoder multi-step prediction method, abbreviated as LSTM-ED, based on the LSTM framework. Liang et al. (2022) extracted the traffic network based on AIS data and used a spatiotemporal multigraph convolutional network to predict the VTF of essential nodes. Zhao et al., (2022a, 2022b) put forward a spatiotemporal dynamic graph neural network method that can effectively capture the spatiotemporal attributes of VTF data. Xu and Zhang (2022) input a constructed spatiotemporal correlation feature matrix into GRU for the VTF prediction task. Zhou et al. (2020) employed CNN, LSTM, and a hybrid of Bi-LSTM and CNN for VTF data prediction. By conducting comparative experiments based on real-world data from Singapore waters, they demonstrated that the combined CNN and Bi-LSTM network exhibited superior prediction performance. Z. Zhang et al. (2019) proposed an improved Particle Swarm Optimisation-Back Propagation (PSO-BP) method to predict VTF data in port areas. Li et al. (2019) explored the predictive performance of VTF using five popular and effective NN methods, namely WNN, ENN, FNN, BPNN, and GRNN. They also leveraged sequence data decomposition techniques, such as Empirical Mode Decomposition (EMD) and Ensemble EMD (EEMD), and similarity measurement algorithms in combination with these NNs to create a three-step hierarchical prediction method. The primary strength of this new approach is its ability to identify the self-similarity within VTF series data, leading to high accuracy of the prediction results. However, existing learning-based prediction methods have limitations in addressing collaborative prediction of VTF in multiple target regions and incorporating time development and period change attributes of VTF data into the training network simultaneously. Therefore, this paper proposes ICLSGNet, a solution to these problems, which is suitable for completing the VTF collaborative prediction task of multi-target waters (referring to the channel in this paper).

2.3. Contributions of our study

The literature review of VTF prediction research has revealed that DL methods suffer from practical deficiencies, highlighting the need for new research to address two key weaknesses. (1) The time-varying characteristics and periodic attributes of VTF data in network training can not fully capture. (2) The realization of information exchange between two channels with high similarity in data changes is challenging when multiple channels need to complete VTF prediction tasks simultaneously.

To tackle these weaknesses, this paper proposes a new learning-based prediction methodology, called ICLSGNet, consisting of three views: spatial view (using CNN), temporal view (using LSTM), and similarity grouping view (using LSTM). In particular, spatial and temporal views can perfectly address the first weakness mentioned above, while the similarity grouping view is used to solve the second disadvantage by facilitating information exchange between highly similar data changes across multiple channels. The main contributions are described below.

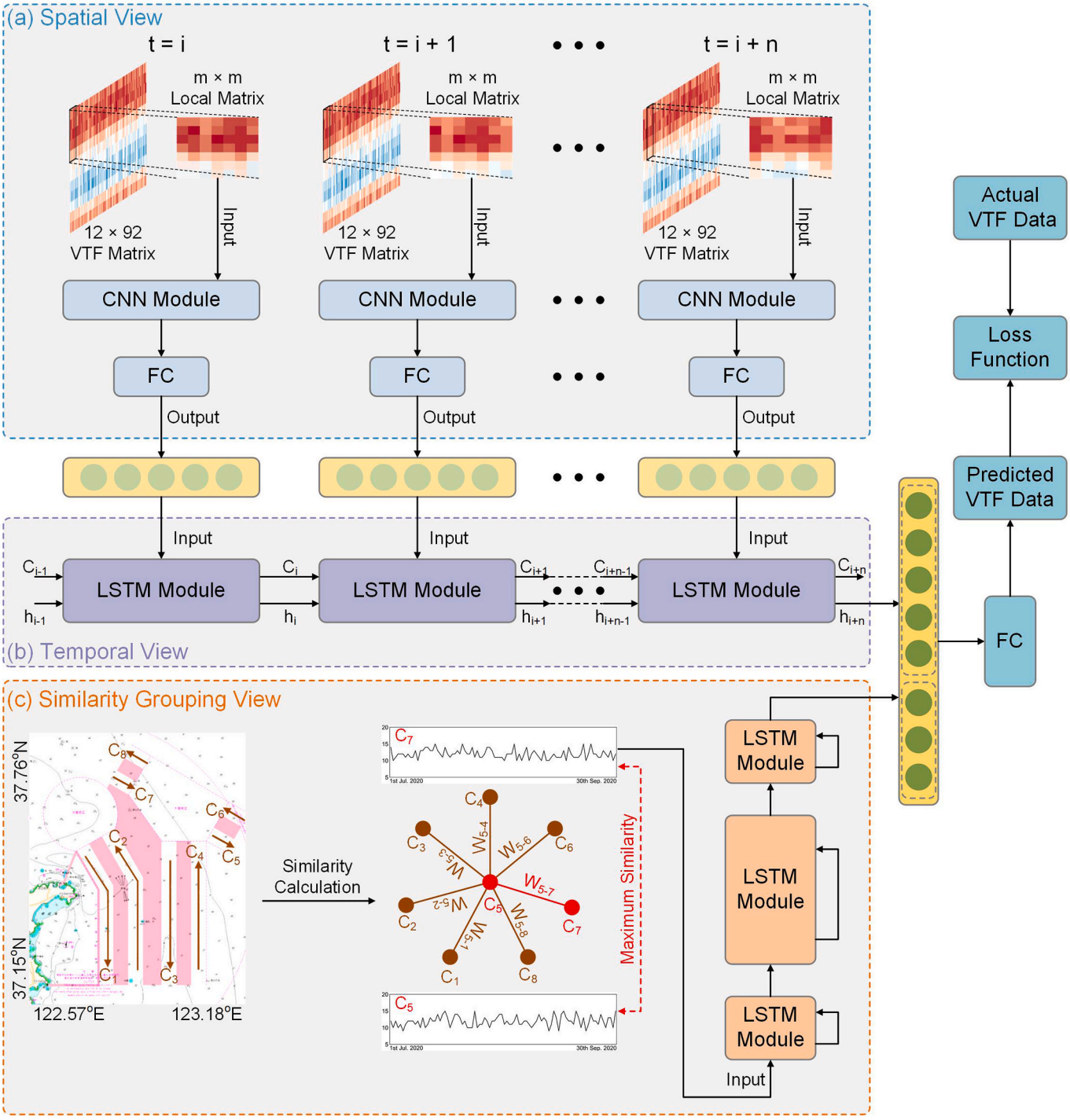

Fig. 3. The flowchart of the ICLSGNet prediction method. **Spatial** and **Temporal views** use CNN and LSTM to mine the spatial (essentially a periodic change) and temporal attributes of VTF, respectively. **Similarity grouping view** calculates the similarity between the historical VTF change trend of the current predicted channel C_5 and other channels, finds the channel C_7 with the greatest similarity with C_5, and then learns the VTF historical change information of the channel based on LSTM. Finally, the knowledge of the three views is fused and input into a fully connected network to get the prediction result.

(1) *Spatial and Temporal Views.* CNN is used to learn periodic change characteristics (i.e. spatial features). Subsequency, LSTM is utilised to mine VTF changing information over time (i.e. temporal relationship) through the received feature vectors by CNN from each neighbourhood matrix of multiple continuous time nodes.

(2) *Similarity Grouping View.* The similarity grouping view is introduced to identify channels with similar VTF characteristics based on Dynamic Time Warping (DTW) method and learn their information through LSTM for collaborative prediction tasks.

(3) *Multi-views Information Fusion.* The outputs from the three views (i.e. spatial, temporal, and similarity grouping views) are fused into a vector and then input into a fully connected network to obtain the prediction results of the next time node.

(4) *Comparative Experiments with Realistic VTF Datasets.* The paper uses realistic AIS data to count VTF data of all channels and conducts a comparative analysis with eleven other advanced prediction methods in the study area. The comparative experiments are conducted on realistic VTF datasets, evaluating the

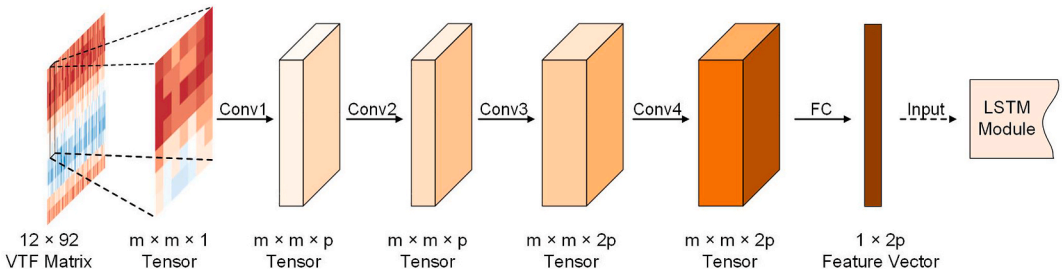

Fig. 4. The working mechanism of the CNN module. Especially, m is the size of each neighbourhood matrix, and p represents the number of convolution kernels.

ICLSGNet method from fine-grained and coarse-grained perspectives.

3. Methodology

A new learning-based prediction method is developed based on the spatiotemporal characteristics of VTF and the interaction between multiple channels. This section introduces the training principle of the proposed ICLSGNet method in detail. Section 3.1 briefly describes the technical framework of the training network. The ICLSGNet method consists of three views, including spatial, temporal, and similarity grouping views. Hence, section 3.2 explains the role of the above three views and the training process. Section 3.3 summarises a suitable loss function for ICLSGNet to complete VTF prediction tasks.

3.1. Overview of the whole framework

This section outlines a new learning-based VTF prediction method, including three views. The spatial view takes the VTF data of a certain time node as the centre and forms a neighbourhood matrix with its surrounding data (i.e. data of adjacent time nodes, data of the same period in the past or future days) as input to CNN to learn periodic attributes (i.e. spatial features). Similarly, the local VTF matrices of multiple continuous time nodes are input to CNN to obtain different eigenvectors. Then the outputs are entered into the temporal view (using LSTM) to learn the time change attribute of VTF data. To realise the interaction of VTF information between two channels with high similarity in the study area, the similarity grouping view selects a channel with high similarity with the target channel based on similarity weight and uses LSTM to learn its VTF information over time. The output results of the spatiotemporal view and the similarity grouping view are fused into a new vector, which is input into a fully connected network to obtain the prediction data of the next time node. The framework of our proposed ICLSGNet is shown in Fig. 3.

In this paper, a day is divided into 12 time periods, and the VTF data of 92 days are calculated. Then the original one-dimensional VTF data in 92 days is converted into a two-dimensional matrix with a size of 12 × 92 in actual network training. The ordinate of the matrix (i.e. 1-12) represents the data in each period of the day. The abscissa (i.e. 1-92) indicates how many days of data have been counted. The data of several consecutive days under a certain period is taken as the training set to predict the VTF data of the next day.

3.2. Details of our proposed methodology

This section describes the implementation principle of the three views in the proposed ICLSGNet method in detail. It is important to first introduce a data preprocessing step: normalizing the VTF data prior to input into the training network. Normalization involves limiting the preprocessed data to a specific range, which helps eliminates any negative effects caused by singular sample data and accelerates the speed of gradient descent to find the optimal solution. Additionally, normalizing the data can improve the accuracy of network training. The method used in this paper is Min-Max Normalization (Islam et al., 2022), which maps the raw data into an interval of 0–1. The specific function expression is shown below.

$$lv_j^{'} = \frac{lv_j - \min(lv)}{\max(lv) - \min(lv)} \tag{1}$$

where lv represents the original VTF sequence data, lv_j denotes the jth

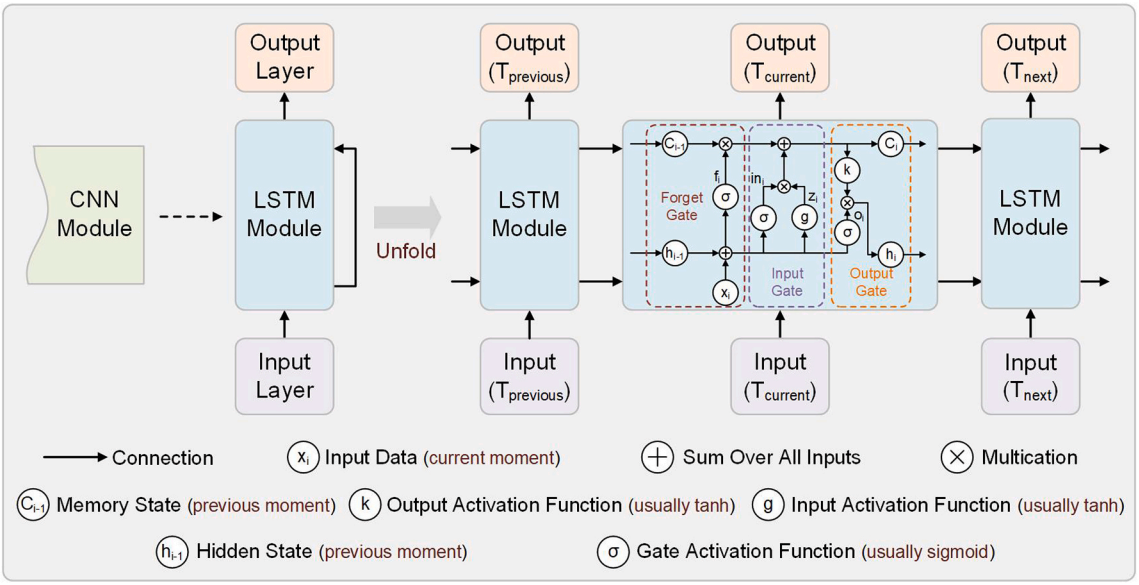

Fig. 5. The working mechanism of the LSTM module.

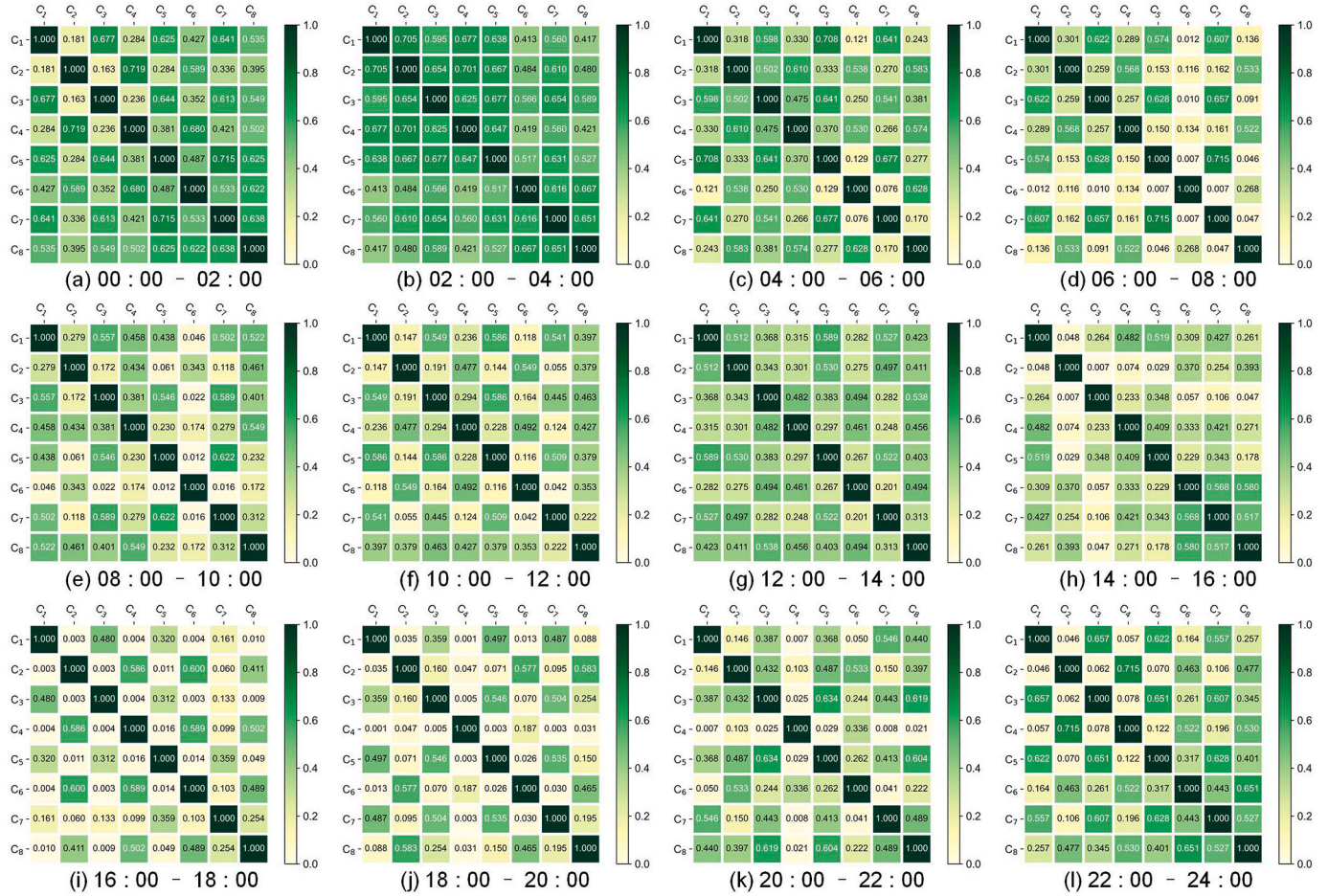

Fig. 6. Visual illustration of similarity matrix of 8 channels in different time periods. From (a) to (l) represent 12 time periods.

value in the sequence data, and $lv_j^{'}$ is the result of normalization. max (lv) and min (lv) are the maximum and minimum values of the original sequence, respectively. Despite its effectiveness, the normalization method mentioned above has a drawback. Specifically, the predicted result may exceed the maximum value or fall below the minimum value of the original sequence data. In other words, while Eq. (1) normalises the VTF data, it also limits the predicted result to some extent within the range of the maximum and minimum values of the original series data. It can ultimately decrease prediction accuracy. In summary, this paper improves Eq. (1), defined as follows,

$$lv_j^{'} = \frac{lv_j - \frac{1}{2} \times \min(lv)}{2 \times \max(lv) - \frac{1}{2} \times \min(lv)} \tag{2}$$

The new function Eq. (2) expands the range between the maximum and minimum values in VTF series data, avoiding potential interference with the predicted values due to normalization operations.

3.2.1. Spatial view

The spatial view uses CNN to mine the spatial characteristics of VTF data at a particular time node. As shown in Fig. 3 (a), a training set $Y = \{y_i, y_{i+1}, \cdots, y_{i+n}\}$ is formed by selecting the VTF data for several consecutive days within a certain period. Each VTF data in the training set and its surrounding data are then used to create a neighbourhood matrix, resulting in a local matrix training set $LM = \{lm_i, lm_{i+1}, \cdots, lm_{i+n}\}$. Finally, each local matrix lm is input to CNN for training to obtain corresponding eigenvectors. Taking the local matrix lm_i (m × m local matrix in Fig. 3 (a)) of the ith time node as an example, the specific process of CNN training is explored. If the central data of the local matrix is located at the edge of the entire VTF matrix, this paper uses 0 to fill the local matrix. Since CNN requires input data in the form of a three-dimensional tensor, the two-dimensional neighbourhood matrix needs to be converted into a tensor $lt_i \in \mathbb{R}^{m \times m \times 1}$ for each time node. The tensor lt_i is then input into CNN to extract the spatial features of VTF through a multi-layer convolution operation. The specific operation process is as follows,

$$lt_i^q = f_q\left(lt_i^{q-1} * W_i^q + b_i^q\right) \tag{3}$$

where W_i^q and b_i^q express the weight and bias tensors of the qth convolution layer, respectively. $f_q(\cdot)$ denotes the activation function (using Rectified Linear Unit (ReLu) (Hara et al., 2015)) of the qth convolution layer, * is the convolutional operation. lt_i^{q-1} and lt_i^{q+1} represent the input and output tensors of the convolution layer, respectively. q represents the qth layer of convolution. When the value of q is 1, lt_i^0 (which essentially is lt_i) indicates the local VTF tensor of the original input. This paper uses four convolution layers to construct CNN in Fig. 4, and then the value of q is from 1 to 4. Meanwhile, the size of the convolution core used for each layer is 3 × 3.

After calculating four convolution layers, the output tensor $lt_i^{Out} \in \mathbb{R}^{m \times m \times 2p}$ is transformed into a two-dimensional matrix $M_i^{Out} \in \mathbb{R}^{(m \times m) \times 2p}$ that is input into a fully connected network to obtain the final eigenvector $lv_i^{Out} \in \mathbb{R}^{1 \times 2p}$. The mathematical expression is as follows,

$$lv_i^{Out} = f\left(M_i^{Out} W_i + b_i\right) \tag{4}$$

where $f(\cdot)$ represents a linear function in the fully connected network. W_i and b_i are the weight and bias values in the network, respectively. The process of CNN extracting the characteristic vector of the neigh-

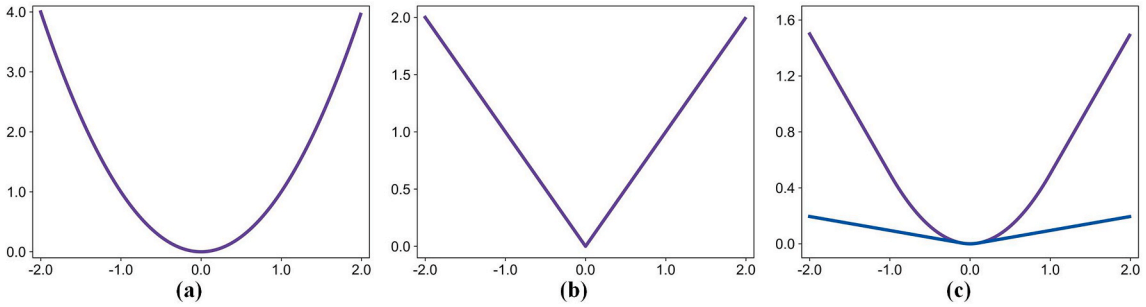

Fig. 7. The visual illustration of three different loss functions, (a) MSE, (b) MAE, and (c) Huber loss.

bourhood matrix is illustrated in Fig. 4. In particular, m and p are hyperparameters in network training and will be further discussed in Section 4.3.

3.2.2. Temporal view

Based on the spatial view, the eigenvectors of each neighbourhood matrix are obtained, which can form a sequence $LV = \{lv_i, lv_{i+1}, \cdots, lv_{i+n}\}$. In the temporal view, the above series is input into LSTM to extract the temporal variation characteristics of VTF. The core network structure of LSTM is similar to the original RNN, which adds a memory state (C) based on a hidden state (h). LSTM has three gate structures: forget, input, and output gates. The forget gate decides how much information from the previous time $C_{i\text{-}1}$ can be discarded and how much information can be retained when it is transferred to the current time C_i. The input gate determines how much information from the input data (lv_i) at the current time step can be saved to C_i. The input and forget gates in the network jointly determine the memory state. The output gate controls how much information from C_i can be transmitted to h_i. The structure of LSTM and the data chain transmission mode of data are illustrated in Fig. 5.

To better illustrate the training process of data in the LSTM unit, this paper provides an example using the eigenvector lv_i of the ith time node. Firstly, the $h_{i\text{-}1}$ at time i-1 and the input sequence lv_i at time i undergo linear transformations by a linear layer. Meanwhile, the results from linear transformations are mapped to the interval [0,1] using an activation function. This process can be expressed as follow,

$$f_i = \sigma\left(W_f[h_{i-1}, lv_i] + b_f\right) \tag{5}$$

where σ is an activation function (using Sigmoid). W_f and b_f represent the weight matrix and bias value, respectively. When the output result of f_i is 0, all information of $C_{i\text{-}1}$ is discarded. If the output result of f_t is 1, all information of $C_{t\text{-}1}$ will be retained and transferred to C_t.

Secondly, the $h_{i\text{-}1}$ at time i-1 and the input sequence lv_i at the time i undergo linear transformations, and their results are passed through different activation functions to obtain in_i and z_i. The function expressions are as follows,

$$in_i = \sigma(W_{in}[h_{i-1}, lv_i] + b_{in}) \tag{6}$$

$$z_i = \tanh(W_z[h_{i-1}, lv_i] + b_z) \tag{7}$$

where W_{in} and W_z represent weight matrices, b_{in} and b_z denote bias values.

Thirdly, the C_i at the moment i can be obtained based on the f_i, $C_{i\text{-}1}$, in_i, and z_i, whose function expression is as follows,

$$C_i = f_i \times C_{i-1} + in_i \times z_i \tag{8}$$

The final output state of h_i at the current time node is determined by not only the $h_{i\text{-}1}$ at the previous time node and the input data lv_i at the current time node but also the C_i at the current time node. The functional expression for calculating h_i is as follows,

$$o_i = \sigma(W_o[h_{i-1}, lv_i] + b_o) \tag{9}$$

$$h_i = o_i \times \tanh(C_i) \tag{10}$$

where W_o and b_o are the weight matrix and bias value, respectively. According to Fig. 3 (b), the output vector of the temporal view (i.e. h_i) and similarity grouping view are fused into a new vector that is input into a fully connected network to get the prediction results.

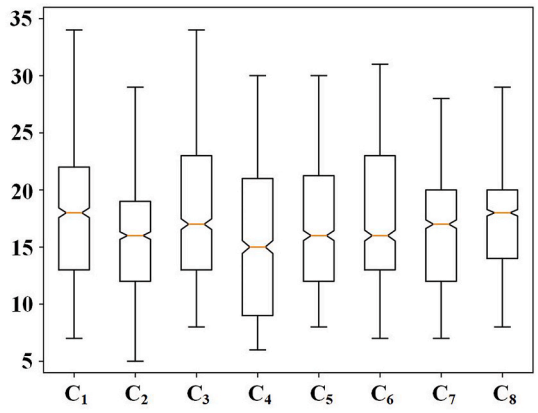

Fig. 8. The visualizing boxplots of VTF data in 8 different experimental datasets.

3.2.3. Similarity grouping view

To address VTF prediction tasks for multiple channels in real-world scenarios, it is often necessary to take into account the correlation between VTF variation patterns across different channels. For instance, as illustrated in Fig. 3 (c), the VTF changes in channel C_7 during a specific period show high similarity with those in channel C_5, which performs the prediction task. Hence, the similarity grouping view searches for a channel with high similarity weights to the target channel based on VTF variation rules, and learns the VTF information of this channel over time through LSTM.

As described in Section 3.1, this paper uses each time period as a reference and input data from consecutive days within that time period to train the model for predicting VTF data for the next day. To assess the similarity of VTF series data between two channels in different periods, this paper employs the DTW method (Li et al., 2019, 2020) and exponential function. The expression is represented in Eq. (11).

$$\omega_{AB} = e^{-DTW(A,B)} \tag{11}$$

where $A = \{a_1, a_2, \ldots, a_g\}$ and $B = \{b_1, b_2, \ldots, b_h\}$ are the VTF series data of the target channel and the other channel, respectively.

The DTW method often yields a relatively large similarity measure between two sequences. It could potentially lead to issues such as gradient vanishing or exploding if the similarity results are applied directly to network training. Hence, this paper maps DTW measures and weight values one-to-one using the exponential function. Given the

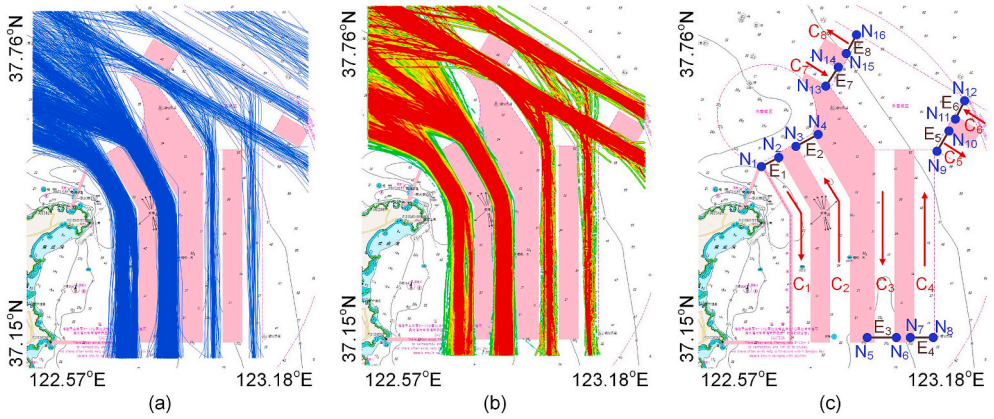

Fig. 9. Visualisation of the CJP water area, (a) visual illustration of vessel trajectories, (b) visual display of vessel trajectories density, and (c) schematic diagram of 8 VTF datasets from C_1 to C_8 in VTF intercepting surface.

characteristics of the exponential function, the result lies between 0 and 1 when the independent variable is negative. Ultimately, the mathematical meaning of Eq. (11) is to express the similarity between two VTF sequences quantitatively, with transformations applied to suit the training network. If the VTF variations of the two channels are identical, the weight value is 1. Conversely, if the VTF variation characteristics largely differ, the weight value tends towards 0.

As a simple and effective measurement method, DTW is often applied to calculate the similarity between two sequence data. The length of two sequence data can be the same or different. In this paper, sequences A and B are equal in length ($g = h$). Before conducting similarity calculations, a $g \times g$ patch matrix is constructed. In this matrix, the element position $p, q = 1, \cdots, g$ represents the weighted Euclidean distance d (a_p, b_q) between the two values, a_p and b_q. The best match between sequences A and B is the shortest path distance after aligning these two sequences. Hence, the optimum warping patch is given by

$$DTW(A,B) = \chi(p,q), \tag{12}$$

where the minimum cumulative distance d (a_p, b_q) is represented in Eq. (13).

$$\begin{aligned}\chi(p,q) &= d(a_p,b_q) + \\ &\min\{\chi(p-1,q-1),\chi(p-1,q),\chi(p,q-1)\}\end{aligned} \tag{13}$$

This paper calculates the VTF similarity weight matrix among the eight channels in 12 different time periods in a day, as shown in Fig. 6.

Suppose that the current VTF prediction task is undertaken on channel C_1 during the first time period (00:00–02:00). According to Fig. 6 (a), the similarity weight of channels C_1 and C_3 is the largest, with a weight value is 0.677. The similarity grouping view leverages LSTM to capture the VTF variation characteristics of channel C_3, and calculated as

$$h_i^{Sim} = \mathrm{LSTM}\left(h_{i-1}^{Sim}, C_{i-1}^{Sim}, v_i^{Sim}; W^{Sim}, b^{Sim}\right), \tag{14}$$

where v_i^{Sim} is the VTF data of the ith time node. h_{i-1}^{Sim} and C_{i-1}^{Sim} represent the hidden state and memory state of the previous time node $i-1$, respectively. W^{Sim} and b^{Sim} indicate the weight and bias values of LSTM during network training, respectively. LSTM$(\cdot)$ contains the contents of Eqs. (5)–(10). h_i^{Sim} is the output vector of the network, which is fused with the output vector of the spatial and temporal views.

3.2.4. Information fusion

The output vector h_i of the spatial and temporal views accurately captures the spatial and temporal attributes of VTF data. Meanwhile, h_i^{Sim} is a feature vector that has a high similarity in the VTF variation pattern between a certain channel and the target channel currently undergoing the prediction task, calculated using the similarity grouping view. The two vectors h_i and h_i^{Sim} are merged into a vector and then input into a fully connected network to obtain the prediction results of the next time node. The functional expressions are represented in Eqs. (15) and (16).

Table 1
The statistical and geometric information of 8 channels in the CJP water area. In particular, each node's longitude and latitude data are obtained under World Geodetic System-1984 Coordinate System.

Channel	Node	Longitude(°)	Latitude(°)
C_1	N_1	122.7042	37.4955
	N_2	122.7393	37.5111
C_2	N_3	122.7746	37.5277
	N_4	122.8107	37.5444
C_3	N_5	122.9429	37.1942
	N_6	122.9851	37.1945
C_4	N_7	123.0266	37.1948
	N_8	123.0689	37.1942
C_5	N_9	123.0692	37.5176
	N_{10}	123.0905	37.5474
C_6	N_{11}	123.1111	37.5758
	N_{12}	123.1320	37.6053
C_7	N_{13}	122.8498	37.6212
	N_{14}	122.8708	37.6501
C_8	N_{15}	122.8910	37.6788
	N_{16}	122.9123	37.7088

$$h_i^{New} = h_i \oplus h_i^{Sim}, \tag{15}$$

$$v_{pre} = f\left(W_{fc} h_i^{New} + b_{fc}\right). \tag{16}$$

where $\oplus$ indicates tandem operation, which can merge two vectors (h_i and h_i^{Sim}) into a new vector (h_i^{New}). W_{fc} and b_{fc} are the weights and bias values for fully connected networks, respectively. $f(\cdot)$ denotes a linear function in the fully connected network. v_{pre} expresses the predicted value of the final output, which is the result based on Min-Max Normalization (Eq. (2)). Therefore, the final prediction result needs to be subjected to inverse normalization processing, defined as follows,

$$v_{pre}^{final} = v_{pre} \times \left(2 \times \max(lv) - \frac{1}{2} \times \min(lv)\right) + \frac{1}{2} \times \min(lv) \tag{17}$$

where max (lv) and min (lv) represent the maximum and minimum values of the original sequence, respectively. lv is the original VTF sequence data. v_{pre} and v_{pre}^{final} denote the predicted values before and after normalization, respectively.

3.3. Loss function

The loss function serves as a crucial indicator for evaluating the training quality of a network model. It essentially measures the error

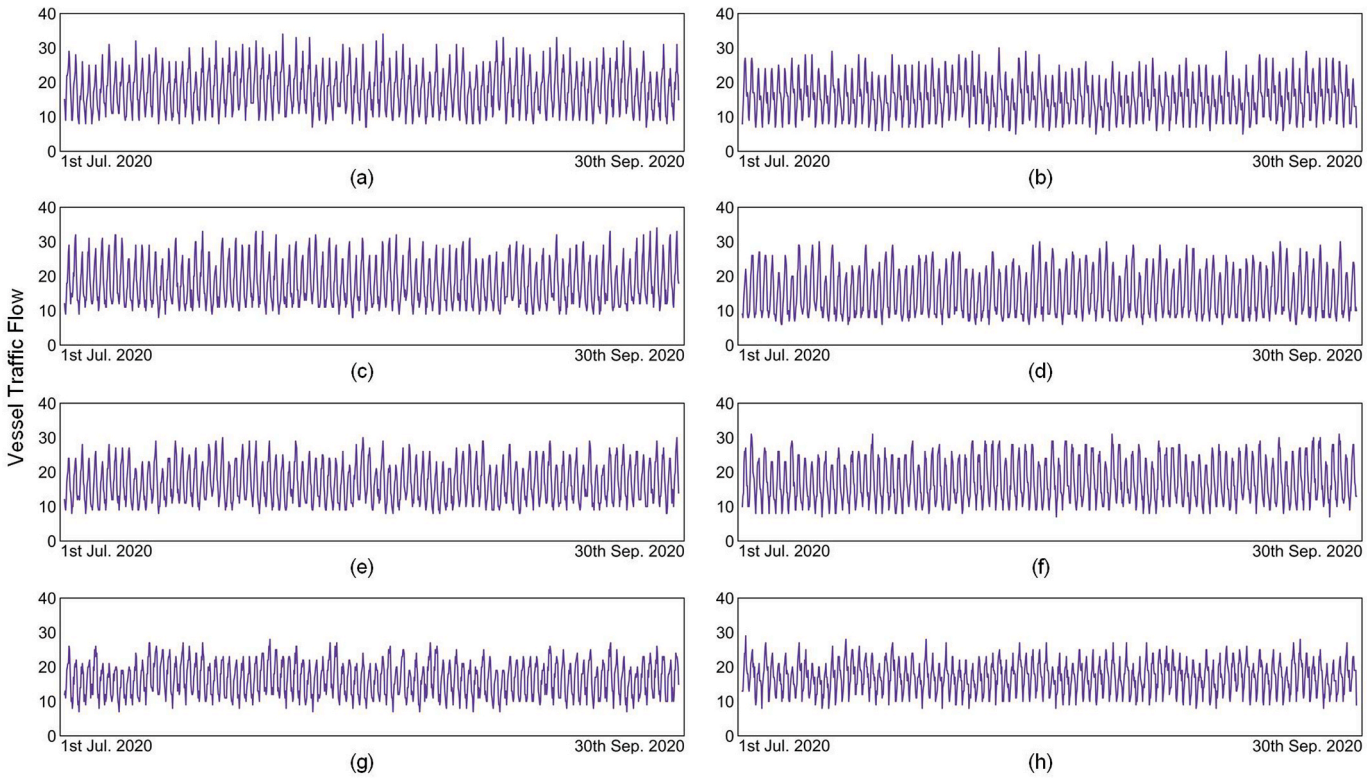

Fig. 10. The change trends of VTF data in 8 different datasets from July 1, 2020 to September 30, 2020, (a) dataset C_1, (b) dataset C_2, (c) dataset C_3, (d) dataset C_4, (e) dataset C_5, (f) dataset C_6, (g) dataset C_7, and (h) dataset C_8.

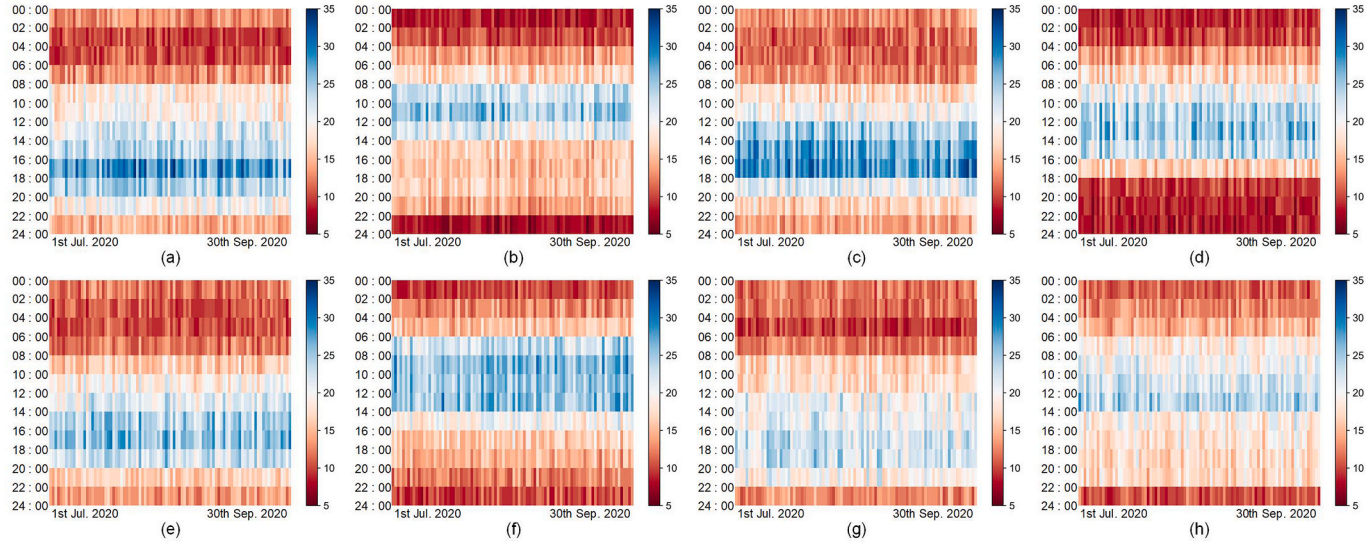

Fig. 11. Visual illustration of VTF matrix (i.e. the size is 12 × 92) in 8 different datasets, (a) dataset C_1, (b) dataset C_2, (c) dataset C_3, (d) dataset C_4, (e) dataset C_5, (f) dataset C_6, (g) dataset C_7, and (h) dataset C_8.

between the observed value and the predicted value. During network training, the network parameters (i.e. weight and bias values) rely on the loss function value to be back-propagated, allowing for iterative optimisation. There are generally two categories of loss functions: regression and classification. The loss function utilised in this paper falls under regression and primarily contains two common functions: Mean Square Error (MSE) (Allen, 1971), which has a smooth and continuous curve, and Mean Absolute Error (MAE), whose curve is V-shaped. Their function images are shown in Fig. 7 (a) and (b), respectively. Meanwhile, their function expressions are as follows,

$$MSE = \frac{1}{k}\sum_{i=1}^{k}(\widetilde{y}_i - y_i)^2, \tag{18}$$

$$MAE = \frac{1}{k}\sum_{i=1}^{k}|\widetilde{y}_i - y_i|. \tag{19}$$

where k indicates the total number of time nodes. $\widetilde{y}_i$ and y_i represent the predicted and actual values of the ith time node, respectively. In real network training, this paper predicts the data of the next time node by VTF data over a period of time, meaning that the value of k is 1. As the

Table 2
Network parameter settings.

	Parameter Name	Parameter Value	Parameter Name	Parameter Value
Spatial View	Size of Each Neighbourhood Matrix	*	–	–
Temporal View	Input Size	*	Hidden Size	2 × Input Size
	Number of Hidden Layers	1	Sequence Length	6
Similarity Grouping View	Input Size	1	Hidden Size	6
	Number of Hidden Layers	1	Sequence Length	6
General Parameters	Learning Rate	*	Iteration	*
	Optimiser	Adamax	–	–

error decreases, MSE results in a decrease in the gradient, promoting the convergence of the function. In contrast, the gradient of MAE remains constant in most cases, even for small loss values. This can hinder the convergence of the function. While MAE has more disadvantages than MSE as a loss function, it is less sensitive to outliers in sample data due to the absence of a square term in its function. This means that the processing power of any sample data is the same. The Huber loss (Meyer, 2021) integrates the advantages of MSE and MAE and is the chosen loss function in this paper, as shown in Eq. (20).

$$L(\theta)=\begin{cases}\frac{1}{2}(\widetilde{y}-y)^2, |\widetilde{y}-y|\le\sigma\\ \sigma|\widetilde{y}-y|-\frac{1}{2}\sigma^2, |\widetilde{y}-y|>\sigma\end{cases} \tag{20}$$

where θ represents learnable parameters based on the training network and σ denotes a hyperparameter. $\widetilde{y}$ and y are predicted and actual values, respectively. As σ is close to 0, the loss function tends to MAE, while σ approaches infinity, it tends towards MSE. Its function image is visually illustrated in Fig. 7 (c). The purple and blue curves represent the function with parameter σ values of 1 and 0.1, respectively. The VTF data used in this paper has almost no outliers and exhibits excellent volatility. As shown in Fig. 8, the VTF data in each dataset used in the comparison experiment are between the lower and upper whiskers in the boxplot, indicating no outlier in the data. Thus, σ is set to 1 in this paper, making the effect of the loss function similar to MSE.

4. Experimental results and discussion

To thoroughly verify the prediction performance of the proposed ICLSGNet method, the experiments in this paper are carried out from two perspectives. One is to demonstrate the prediction performance of the ICLSGNet method under different network parameters. The other is a comparative experiment with eleven other classical prediction methods.

4.1. Dataset description

The proposed ICLSGNet method is particularly designed to handle multi-objective regional VTF collaborative predictions. Consequently, it's essential to have an experimental water area that comprises multiple channels, such as the CJP water area, to evaluate this new method's ability to predict VTF data across all channels simultaneously. This paper calculates the VTF data of 8 channels in Chengshan Jiao Promontory (CJP) based on realistic AIS data, as shown in Fig. 9. The distribution of vessel trajectories and the density of vessel trajectories in each channel in CJP waters are presented in Fig. 9 (a) and (b), respectively. The red arrows in Fig. 9 (c) indicate the vessels' sailing direction in the channel.

The VTF data refers to the cumulative count of all vessels passing through a specific place per unit of time. Thus, the VTF count is determined by the formation of a brown edge between two blue nodes, as depicted in Fig. 9 (c). For instance, if a vessel's trajectory crosses edge E_1, which is formed by points N_1 and N_2, within a specific period, then the VTF count for channel C_1 is incremented by one. Similarly, the VTF data of all channels (i.e. channels C_1–C_8) is computed from the longitude and latitude information available in the AIS data. The specific longitude and latitude of all nodes (i.e. N_1–N_{16}) are provided in Table 1 to aid in the generation of VTF data.

Traditional learning-based prediction methods, such as BPNN, WNN, RNN, LSTM, GRU, Seq2Seq, and Transformer, input the original one-dimensional sequence data directly into the training network. In contrast, the ICLSGNet method proposed in this study utilises CNN and LSTM for spatial and temporal views, respectively, thereby mining temporal evolution characteristics and periodic change attributes of VTF. Consequently, the original one-dimensional VTF sequence data is transformed into a two-dimensional matrix (hour of the day × day). The continuous local VTF matrix periods are then captured and input into the spatial view's CNN. Feature vectors, derived from each local VTF matrix via CNN, are subsequently input into the LSTM within the temporal view. Moreover, the similarity grouping view involves inputting the one-dimensional VTF sequence data from a channel with similarity to the current prediction task. To ensure enough data for network training, this paper has utilised the VTF data in a 3-month period from July 1, 2020 to September 30, 2020. The day has been divided into 12

Table 4
The ConvNet configurations and the input size for different neighbourhood matrix sizes from the temporal view.

Size of Each Neighbourhood Matrix	3 × 3	5 × 5	7 × 7
conv1	conv3-16	conv3-32	conv3-64
conv2	conv3-16	conv3-32	conv3-64
conv3	conv3-32	conv3-64	conv3-128
conv4	conv3-32	conv3-64	conv3-128
Input Size	32	64	128

Table 5
The experimental environment configuration.

Hardware	Model	Software	Version
CPU	i7-12700KF Dodeca Core	Python	3.8.3
Host Memory	32 GB	Pytorch	1.12.1

Table 3
Different parameter settings for the size of each neighbourhood matrix, learning rates, and iterations.

Parameter Name	Parameter Value				
Size of Each Neighbourhood Matrix	3 × 3		5 × 5		7 × 7
Learning Rate	0.0001	0.0005	0.001	0.005	0.01
Iteration	200	400	600	800	1000

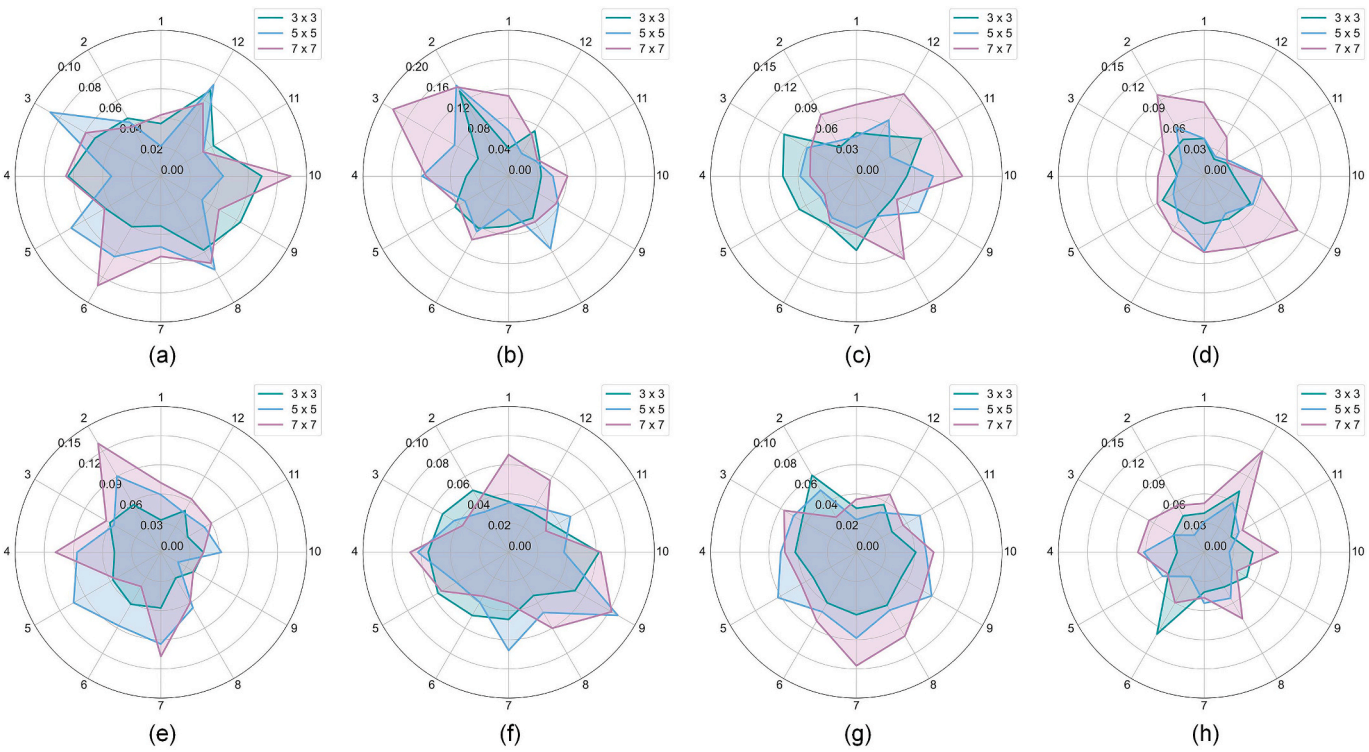

Fig. 12. Visual illustration of REMean for three different sizes of neighbourhood matrix (i.e. 3 × 3, 5 × 5, and 7 × 7) based on the proposed ICLSGNet at 12 periods in 8 different VTF datasets. From (a) to (h) represent datasets from C_1 to C_8, respectively.

periods with a 2-h interval, resulting in 12 VTF data nodes per day. Each dataset comprises 1104 VTF datasets, which show their growing trends as displayed in Fig. 10. Since the proposed ICLSGNet needs input data in the form of a matrix, this paper converts the original one-dimensional data into a VTF matrix based on the hour of the day and the day itself, visually illustrated in Fig. 11.

4.2. Evaluation metrics

To quantitatively evaluate the accuracy and stability of prediction results of ICLSGNet and eleven other methods in predicting VTF in each period, this paper employs two metrics: the mean relative error (REMean) and the standard deviation of the relative error (REStd). The

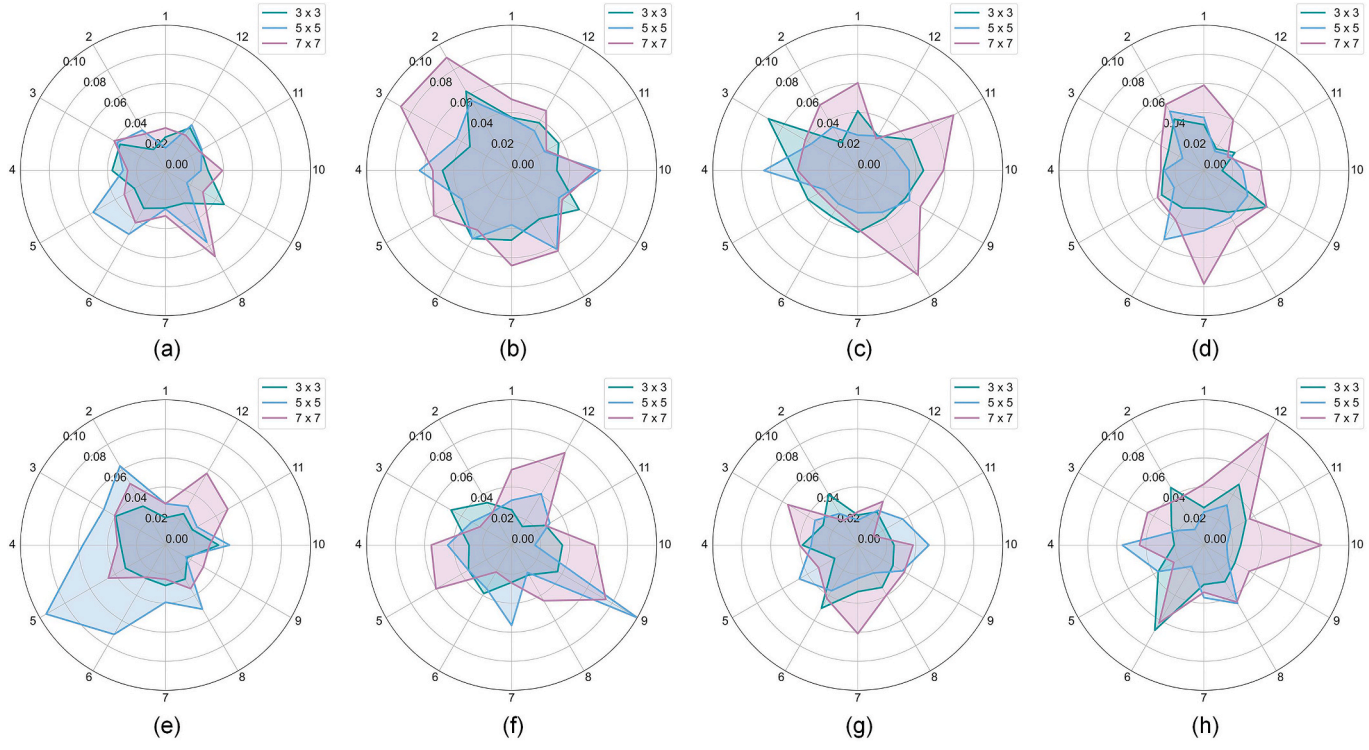

Fig. 13. The display of the REStd for three different sizes of neighbourhood matrix (i.e. 3 × 3, 5 × 5, and 7 × 7) based on the proposed ICLSGNet at 12 periods in 8 different VTF datasets. From (a) to (h) represent datasets from C_1 to C_8, respectively.

Table 6
The prediction results (i.e. RMSE and MAPE) on September 30, 2020 for three different sizes of neighbourhood matrix (i.e. 3 × 3, 5 × 5, and 7 × 7) based on the ICLSGNet in 8 VTF datasets.

Dataset	Evaluation Metrics	Size of Each Neighbourhood Matrix		
		3 × 3	5 × 5	7 × 7
C_1	RMSE	**0.4517**	0.6309	0.9465
	MAPE	**0.0200**	0.0309	0.0384
C_2	RMSE	**0.4753**	0.9516	1.2619
	MAPE	**0.0309**	0.0644	0.0849
C_3	RMSE	**0.9264**	1.1680	1.2121
	MAPE	**0.0355**	0.0461	0.0474
C_4	RMSE	**0.4066**	0.6791	0.9240
	MAPE	**0.0198**	0.0262	0.0394
C_5	RMSE	**0.7611**	1.0795	1.3145
	MAPE	**0.0305**	0.0394	0.0602
C_6	RMSE	**0.4959**	0.5233	0.5138
	MAPE	**0.0190**	0.0245	0.0205
C_7	RMSE	**0.4631**	0.6740	0.7479
	MAPE	**0.0185**	0.0291	0.0314
C_8	RMSE	**0.4111**	0.4741	0.6865
	MAPE	**0.0185**	0.0213	0.0348

two indexes can be expressed as follows,

$$REMean = \frac{1}{k}\sum_{i=1}^{k}\frac{|\widetilde{y}_i - y|}{y}, \tag{21}$$

$$REStd = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(\frac{|\widetilde{y}_i - y|}{y} - REMean\right)^2}. \tag{22}$$

with k indicates each method's running times when performing the prediction task. To mitigate the possibility of a particular operational result, each prediction method is executed ten times under different conditions in this paper. Hence, the value of k in the experiment is 10. $\widetilde{y}_i$ and y represent ith predicted and actual value. The minimum *REMean* and *REStd*, the better the prediction performance.

REMean and *REStd* are utilised to assess the prediction performance based on fine granularity. However, this study selects the Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) as quantitative evaluation indicators to further gauge the performance of each method to predict VTF data in all periods of the day. The quantitative evaluation indexes of coarse granularity are defined as follows,

$$RMSE = \sqrt{\frac{1}{l}\sum_{j=1}^{l}\left(\overline{x}_j - x_j\right)^2}, \tag{23}$$

$$MAPE = \frac{1}{l}\sum_{j=1}^{l}\left|\frac{\overline{x}_j - x_j}{x_j}\right|. \tag{24}$$

where $\overline{x}_j$ and x_j represent the jth average predicted value and the factual VTF data, respectively. j refers to the jth period of the day. l is the number of periods in a day, indicating the overall number of VTF prediction data. In this paper, the day is divided into 12 time periods, each with an interval of 2 h. Hence, the value of l in the experiment is 12. In summary, REMean and REStd are used to evaluate the accuracy and stability of each prediction method in predicting VTF data at different time nodes. RMSE and MAPE, as two comprehensive indicators, can quantitatively assess the accuracy of each method in predicting VTF data for the upcoming day. Essentially, the prediction results of 12 time nodes are collectively assessed for effectiveness.

4.3. Network parameter settings

The proposed ICLSGNet prediction method is optimised based on CNN and LSTM. The network needs to set relevant parameters when training historical data. The proposed ICLSGNet method integrates the spatial, temporal, and similarity grouping view by CNN and LSTM. Hence, parameters are selected based on three different perspectives. The types of parameters are primarily from CNN and LSTM, such as learning rate, iteration, and optimiser, as shown in Table 2.

The parameter 'sequence length' in Table 2 refers to the number of

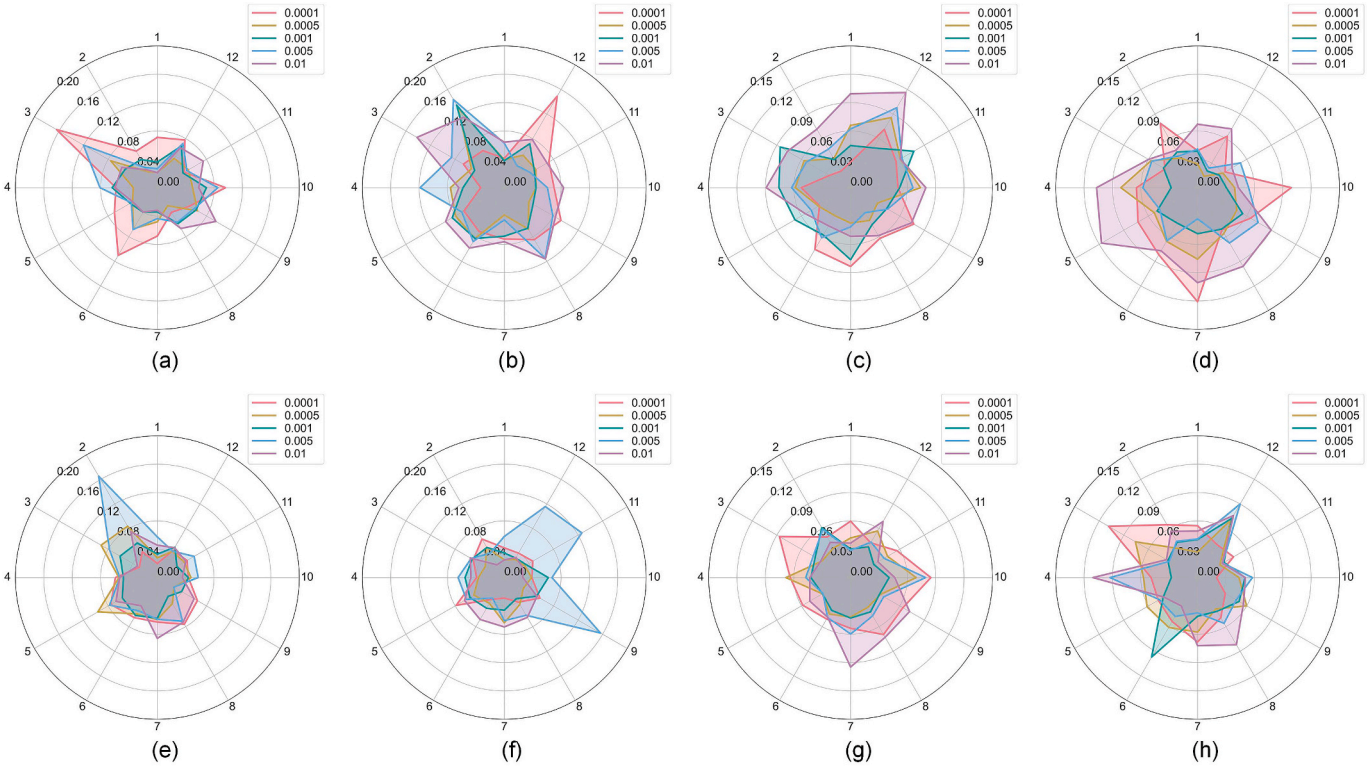

Fig. 14. Visual illustration of mean relative error (REMean) for five different learning rates (i.e. 0.0001, 0.0005, 0.001, 0.005, 0.01) based on our proposed ICLSGNet at 12 periods in 8 different VTF datasets. From (a) to (h) represent datasets from C_1 to C_8, respectively.

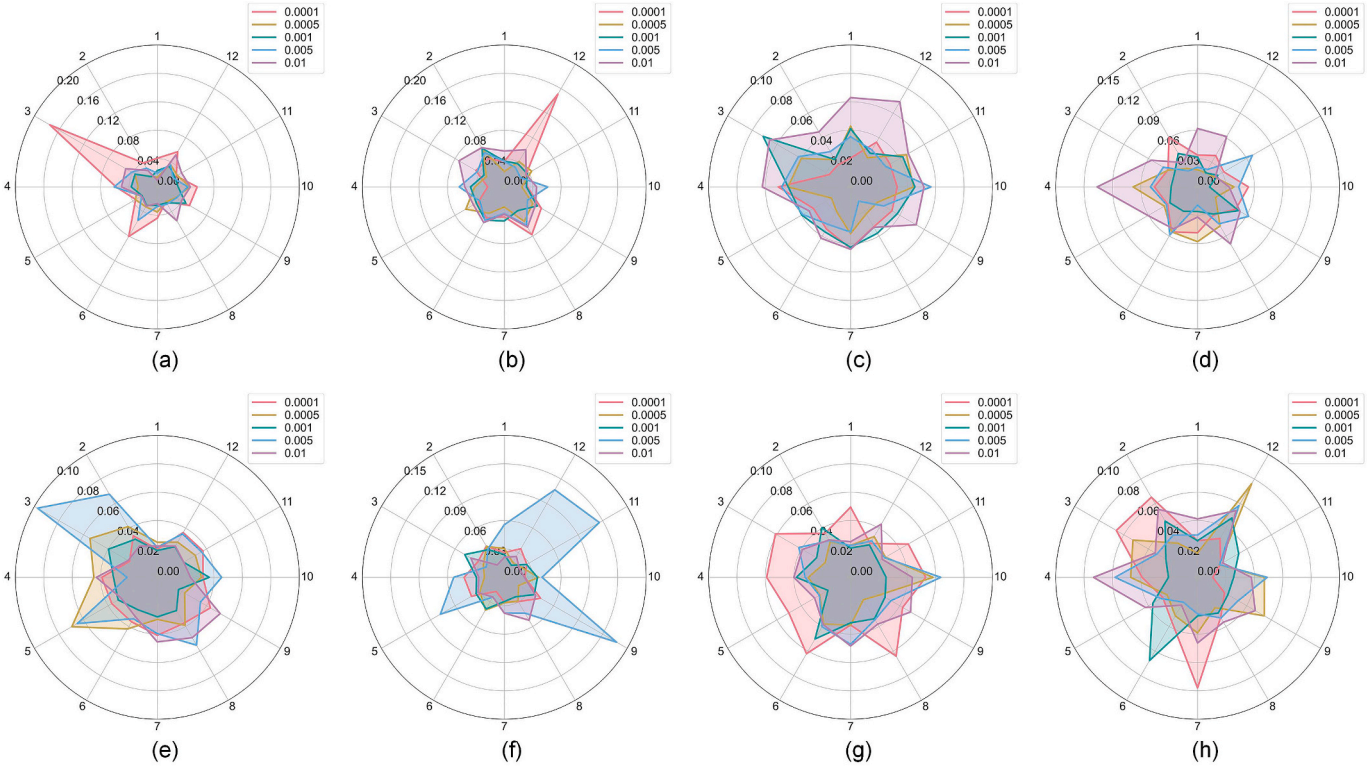

Fig. 15. The standard deviation of relative error (REStd) display for five different learning rates (i.e. 0.0001, 0.0005, 0.001, 0.005, 0.01) based on our proposed ICLSGNet at 12 periods in 8 different VTF datasets. From (a) to (h) represent datasets from C_1 to C_8, respectively.

Table 7
The prediction results (i.e. RMSE and MAPE) on September 30, 2020 for five different learning rates (i.e. 0.0001, 0.0005, 0.001, 0.005, 0.01) based on our proposed ICLSGNet in 8 VTF datasets.

Dataset	Evaluation Metrics	Learning Rate				
		0.0001	0.0005	0.001	0.005	0.01
C_1	RMSE	0.8699	0.5598	**0.4517**	0.8363	1.0821
	MAPE	0.0396	0.0220	**0.0200**	0.0335	0.0460
C_2	RMSE	0.7008	0.7305	**0.4753**	0.9516	1.1555
	MAPE	0.0409	0.0440	**0.0309**	0.0644	0.0818
C_3	RMSE	1.3134	**0.9059**	0.9264	0.9571	1.2659
	MAPE	0.0515	**0.0347**	0.0355	0.0400	0.0485
C_4	RMSE	1.2267	0.6167	**0.4066**	0.7155	1.1894
	MAPE	0.0535	0.0276	**0.0198**	0.0373	0.0459
C_5	RMSE	1.0816	**0.7343**	0.7611	0.9555	1.2171
	MAPE	0.0423	**0.0291**	0.0305	0.0376	0.0489
C_6	RMSE	0.7278	0.5812	**0.4959**	0.7521	0.9131
	MAPE	0.0292	0.0212	**0.0190**	0.0296	0.0341
C_7	RMSE	0.7723	0.5604	**0.4631**	0.7856	0.9608
	MAPE	0.0310	0.0282	**0.0185**	0.0351	0.0352
C_8	RMSE	0.7671	0.6829	**0.4111**	0.7321	0.8853
	MAPE	0.0371	0.0307	**0.0185**	0.0333	0.0398

nodes' data that are input into the training network simultaneously, which also determines the number of LSTM units connected in the network. Different parameter values may affect the accuracy of the network in learning historical data characteristics, such as the parameters marked with asterisks in Table 2. Therefore, this paper conducts a comparative experiment under different network parameters in section 4.5.1, including the size of each neighbourhood matrix, learning rate, and iteration, as presented in Table 3.

The size of the local matrix input in a CNN network is closely linked to the input sizes in the spatial view, as it determines the size of each neighbourhood matrix. Through four-layer convolution, the CNN network learns the features of the neighbourhood matrix in the spatial view. When the neighbourhood matrix size changes, the number of convolution cores in each layer also changes, resulting in a different size of the eigenvector output in the fully connected network of the CNN. Thus, the parameter 'input size' in spatial view is also marked with an asterisk. The convolution layer setting parameters and input size of the temporal view under the local VTF input matrix of different sizes are shown in Table 4. Notably, the term "con*v*a-*b*" indicates that the size of the convolutional kernel is $a \times a$ and b refers to the depth of the convolutional layer.

4.4. Comparison with other prediction methods

This paper compares eleven standard and classic VTF prediction methods to verify the excellent performance of the proposed ICLSGNet method. The details of the comparative methods are as follows.

(a) HMM (Vuković et al., 2015): It is a probability model for time series that utilises elementary model parameters. Despite extensive training data, it is more operationally efficient than NN and DL models. However, it is difficult for HMM to learn the irregular and non-stationary data characteristics, thus significantly reducing prediction accuracy.
(b) ARIMA (Li and Hu, 2012): This model is a hybrid of the sliding average and the autoregressive models. Its concept involves learning from historical data that changes with time and using this knowledge to predict the future.
(c) SVM (Jose et al., 2021): As a common and typical ML method, it adopts the structural risk minimisation principle, solves the learning problem of small samples well, and has a globally unique optimal solution. In the VTF prediction task, future data development can be estimated by fitting the change characteristics of historical data.
(d) BPNN (Hecht-nielsen, 1992): It is a multilayer feedforward NN trained according to the error backpropagation algorithm. Its network structure mainly comprises input, hidden, and output layers.

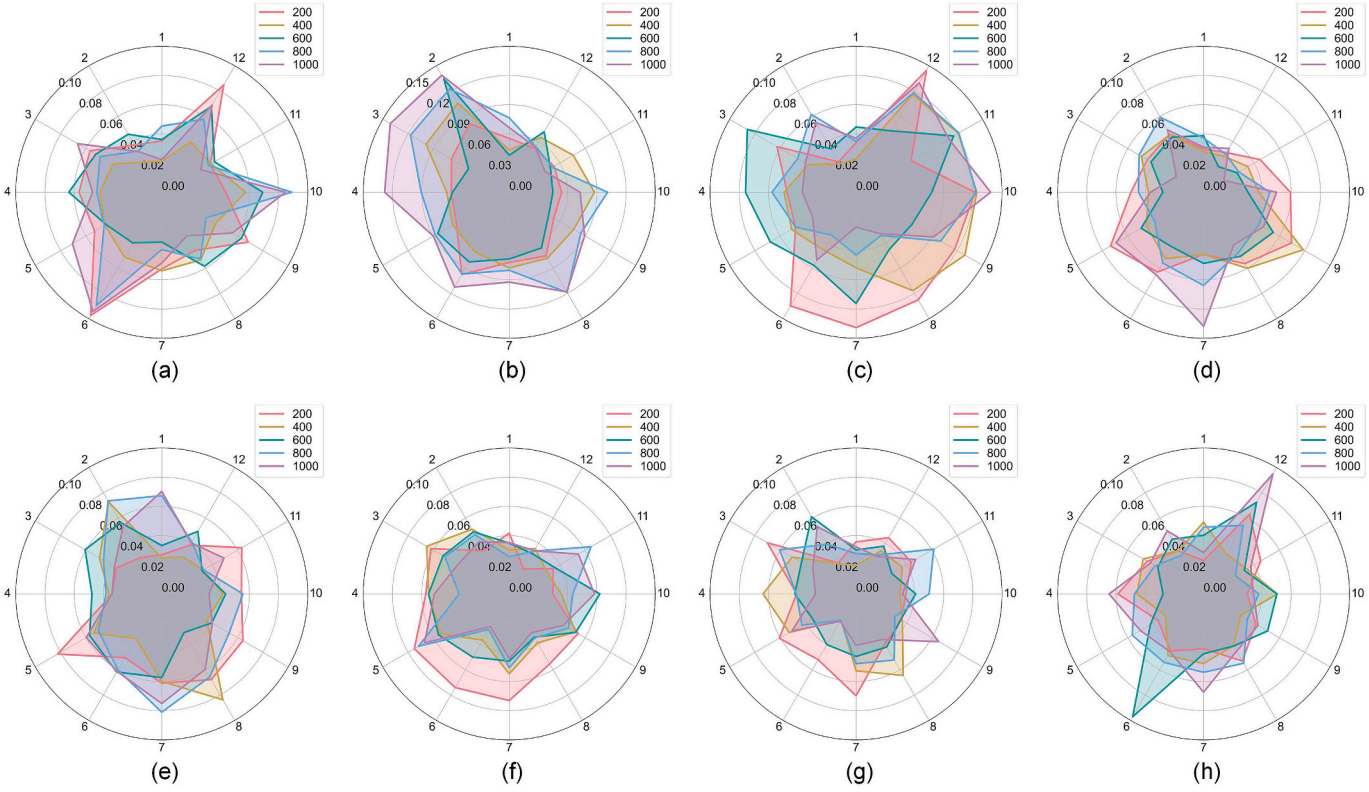

Fig. 16. Visual illustration of REMean for five different iterations (i.e. 200, 400, 600, 800, and 1000) based on our proposed ICLSGNet at 12 periods in 8 different VTF datasets. From (a) to (h) represent datasets from C_1 to C_8, respectively.

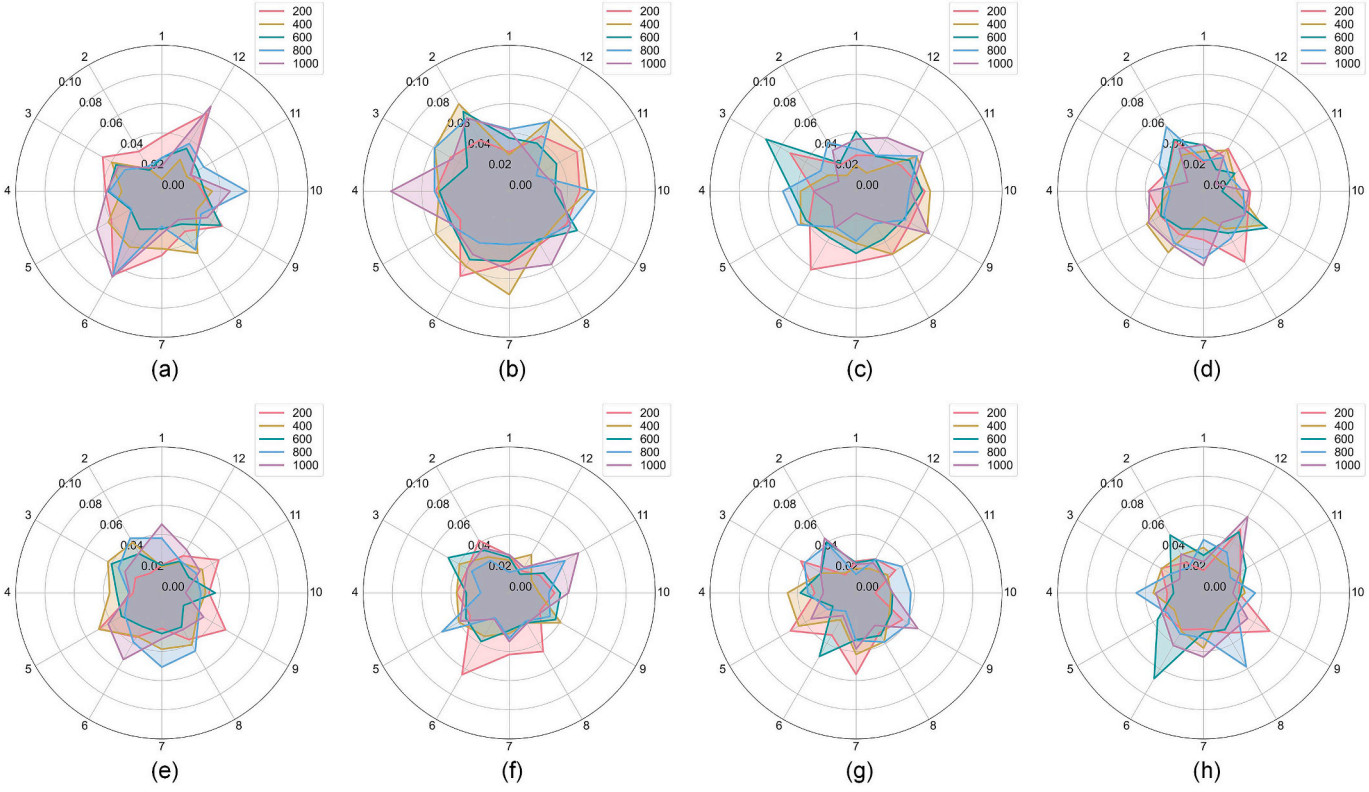

Fig. 17. The REStd result display for five different iterations (i.e. 200, 400, 600, 800, and 1000) based on our proposed ICLSGNet at 12 periods in 8 different VTF datasets. From (a) to (h) represent datasets from C_1 to C_8, respectively.

Table 8
The prediction results (i.e. RMSE and MAPE) of VTF on September 30, 2020 for five different iterations (i.e. 200, 400, 600, 800, and 1000) based on our proposed ICLSGNet in 8 VTF datasets.

Dataset	Evaluation Metrics	Iteration				
		200	400	600	800	1000
C_1	RMSE	0.7776	0.5214	**0.4517**	0.8475	0.9178
	MAPE	0.0282	0.0212	**0.0200**	0.0332	0.0410
C_2	RMSE	0.7687	0.6260	**0.4753**	1.2300	1.3075
	MAPE	0.0433	0.0431	**0.0309**	0.0809	0.0815
C_3	RMSE	1.4787	1.3640	**0.9264**	1.0780	1.0666
	MAPE	0.0520	0.0481	**0.0355**	0.0421	0.0381
C_4	RMSE	0.6481	**0.3872**	0.4066	0.6537	0.8750
	MAPE	0.0281	**0.0197**	0.0198	0.0298	0.0323
C_5	RMSE	1.0251	0.8691	**0.7611**	1.0627	1.0370
	MAPE	0.0370	0.0362	**0.0305**	0.0473	0.0438
C_6	RMSE	0.7558	0.6031	**0.4959**	0.5509	0.5507
	MAPE	0.0266	0.0242	**0.0190**	0.0240	0.0208
C_7	RMSE	0.6215	**0.4434**	0.4631	0.4701	0.5892
	MAPE	0.0274	**0.0181**	0.0185	0.0225	0.0241
C_8	RMSE	0.5027	0.4186	**0.4111**	0.5403	0.6987
	MAPE	0.0232	0.0201	**0.0185**	0.0252	0.0348

(e) WNN (Zhang et al., 1995): It is improved based on BPNN, replacing the original sigmoid activation function of the hidden layer with a wavelet function.
(f) RNN (Sherstinsky, 2020): It is a feedforward model with a memory function whose network structure can transmit historical information to the current moment and integrate it. Hence, it is widely used to predict time series data (i.e. TF and VTF).
(g) LSTM (Graves, 2012): The model optimises the network structure of RNN to address the issue of exploding or vanishing gradients during training.
(h) GRU (Dey and Salem, 2017): This model is improved based on LSTM with the aim of reducing the model complexity by reducing the construction of LSTM units. Unlike LSTMs, which feature three gates (i.e. forget, input, and output gates), GRU employs gated recurrent units (GRUs), which comprise only two gates (i.e. update and reset gates).
(i) Seq2Seq (Sutskever et al., 2014): It is based on the framework of Encoder-Decoder, which can solve the problem that the length of each group data in the input layer of RNNs must be consistent. In particular, the encoder and decoder can be implemented based on RNN and its upgraded versions, such as LSTM and GRU.
(j) Transformer (Han et al., 2021a,b): It is similar to Seq2Seq and based on the Encoder-Decoder framework to construct a training network. Compared to Seq2Seq, its most significant advantage is the effective utilisation of attention mechanisms to improve the ability of training networks to capture data changes and features. Moreover, this method is suitable for parallel computing, improving training efficiency to a certain extent.
(k) CNN-LSTM (Kim and Cho, 2019): The network can effectively learn the spatiotemporal characteristics of time series data (i.e. TF and VTF). The proposed prediction method's spatial and temporal views are created based on CNN and LSTM.

In particular, HMM, ARIMA and SVM are traditional ML methods. BPNN and WNN belong to NN methods. RNN, LSTM, GRU, CNN-LSTM, Seq2Seq, and Transformer are among the DL methods.

4.5. Prediction results analysis

In this paper, all comparative experiments are conducted under the same hardware and software platforms to ensure fairness, whose information is listed in Table 5.

4.5.1. Prediction results of different network parameters

This section focuses on the impact of ICLSGNet on the accuracy and stability of VTF prediction under different values of three important network parameters, including the size of each neighbourhood matrix, learning rate, and iteration. Comparative experiments are conducted to evaluate the influence of different neighbourhood matrix sizes (i.e. 3 ×3, 5 × 5, and 7 × 7) on the prediction results. Meanwhile, the learning rate and iteration parameters are set to 0.001 and 600, respectively.

Quantitative evaluations show that of fine granularity, the proportions of getting the minimum REMean value are 54.16%, 32.29%, and 13.55% for neighbourhood matrix sizes 3 × 3, 5 × 5, 7 × 7, respectively, as shown in Fig. 12. Additionally, the proportions of obtaining the minimum *REStd* value are 41.67%, 40.63%, and 17.70% for sizes 3 × 3, 5 × 5, 7 × 7, respectively, as shown in Fig. 13. The results in Figs. 12 and 13 show that the ICLSGNet prediction method performs better in most cases when the size of the neighbourhood matrix is 3 × 3.

According to the comparative analysis of coarse granularity, the size of 3 × 3 yields the minimum RMSE and MAPE in all experimental datasets, as shown in Table 6. Hence, using a neighbourhood matrix of size 3 × 3 enables ICLSGNet to effectively learn the spatiotemporal

Table 9
Network parameter settings for different other learning-based comparison methods.

Method		Parameter Name	Parameter Value	Parameter Name	Parameter Value
BPNN, WNN		Input Size	6	Hidden Size	2 × Input Size
		Learning Rate	0.001	Iteration	600
		Optimiser	Adamax	Output Size	1
RNN, LSTM, GRU,		Input Size	1	Hidden Size	6
		Learning Rate	0.001	Iteration	600
		Sequence Length	6	Optimiser	Adamax
		Output Size	1	–	–
Seq2Seq	Encoder	Input Size	1	Hidden Size	6
		Sequence Length	6	Output Size	6
	Decoder	Input Size	6	Hidden Size	2 × Input Size
		Sequence Length	1	Output Size	1
	General Parameters	Learning Rate	0.001	Iteration	600
		Optimiser	Adamax	–	–
Transformer		Input Size	1	Hidden Size	6
		Sequence Length	6	Output Size	1
		Learning Rate	0.001	Iteration	600
		Optimiser	Adamax	–	–
CNN-LSTM		Size of Each Neighbourhood Matrix	3 × 3	Input Size	32
		Hidden Size	2 × Input Size	Sequence Length	6
		Learning Rate	0.001	Iteration	600
		Optimiser	Adamax	Output Size	1

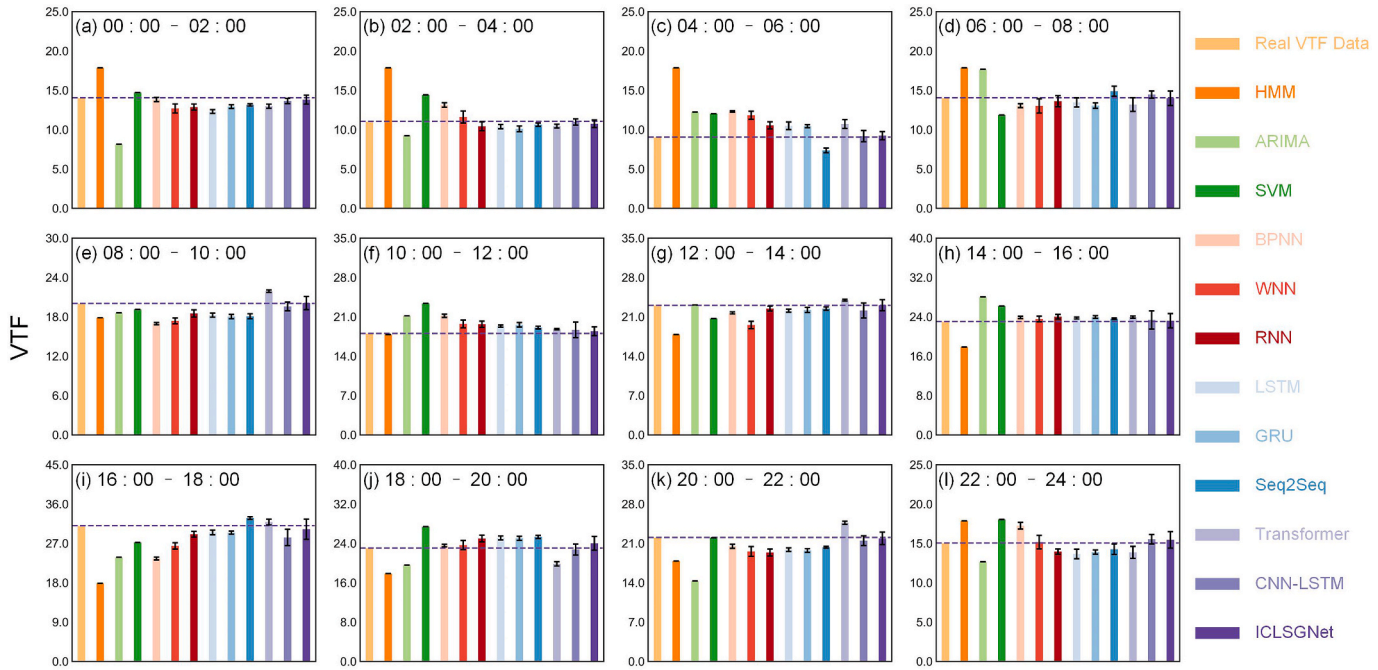

Fig. 18. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_1.

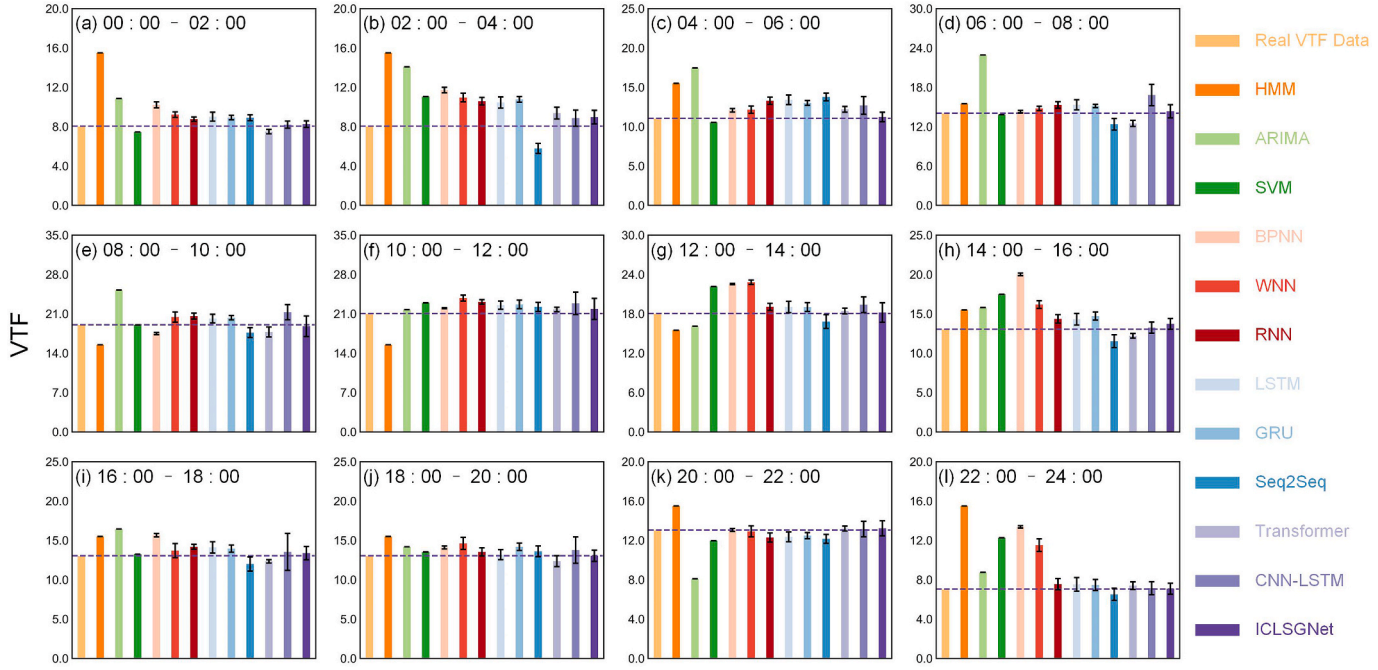

Fig. 19. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_2.

attribute of VTF and obtain accurate and stable prediction results. The size of each neighbourhood matrix is primarily derived from the spatial view in the ICLSGNet method, indicating the influence of periodic variations on VTF data at a specific time node. A larger value for this parameter would suggest a more extended period of influence. However, in real-world application scenarios, the fluctuation pattern of VTF data is predominantly exhibited within shorter cycles. Consequently, a larger parameter might lead to less desirable prediction outcomes.

Furthermore, the impact of varying learning rates (i.e. 0.0001, 0.0005, 0.001, 0.005, and 0.01) on the prediction accuracy of VTF data is compared. The size of each neighbourhood matrix and iteration parameters are fixed at 3 × 3 and 600. Based on the fine granularity prediction results, the percentages of achieving the minimum REMean value are 20.83% (0.0001), 26.04% (0.0005), 30.21% (0.001), 11.46% (0.005), and 11.46% (0.01), as displayed in Fig. 14. Similarly, the proportions of getting the minimum REStd value are 12.50% (0.0001), 29.17% (0.0005), 34.36% (0.001), 14.58% (0.005), and 9.39% (0.01), as shown in Fig. 15.

The coarse granularity prediction results reveal that while the learning rate of 0.0005 yields the highest prediction accuracy in datasets C_3 and C_5, a learning rate of 0.001 results in the lowest RMSE and MAPE across most VTF datasets, as demonstrated in Table 7. This experimental

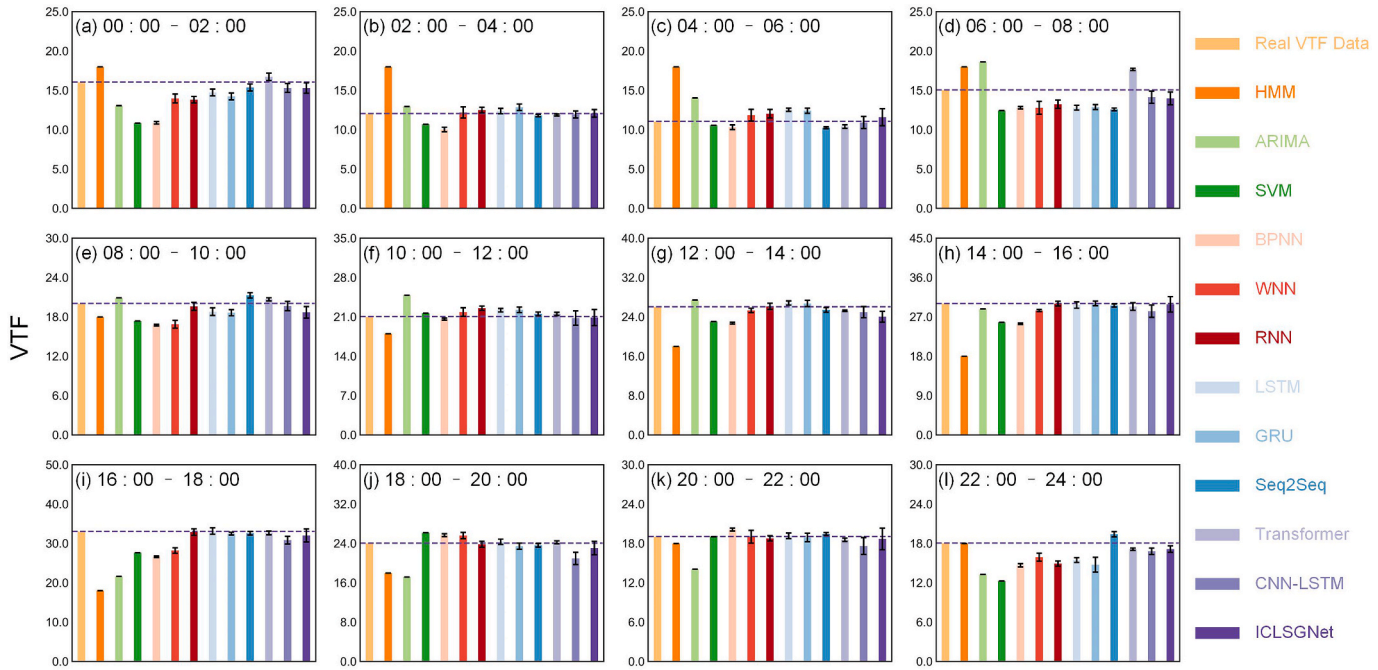

Fig. 20. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_3.

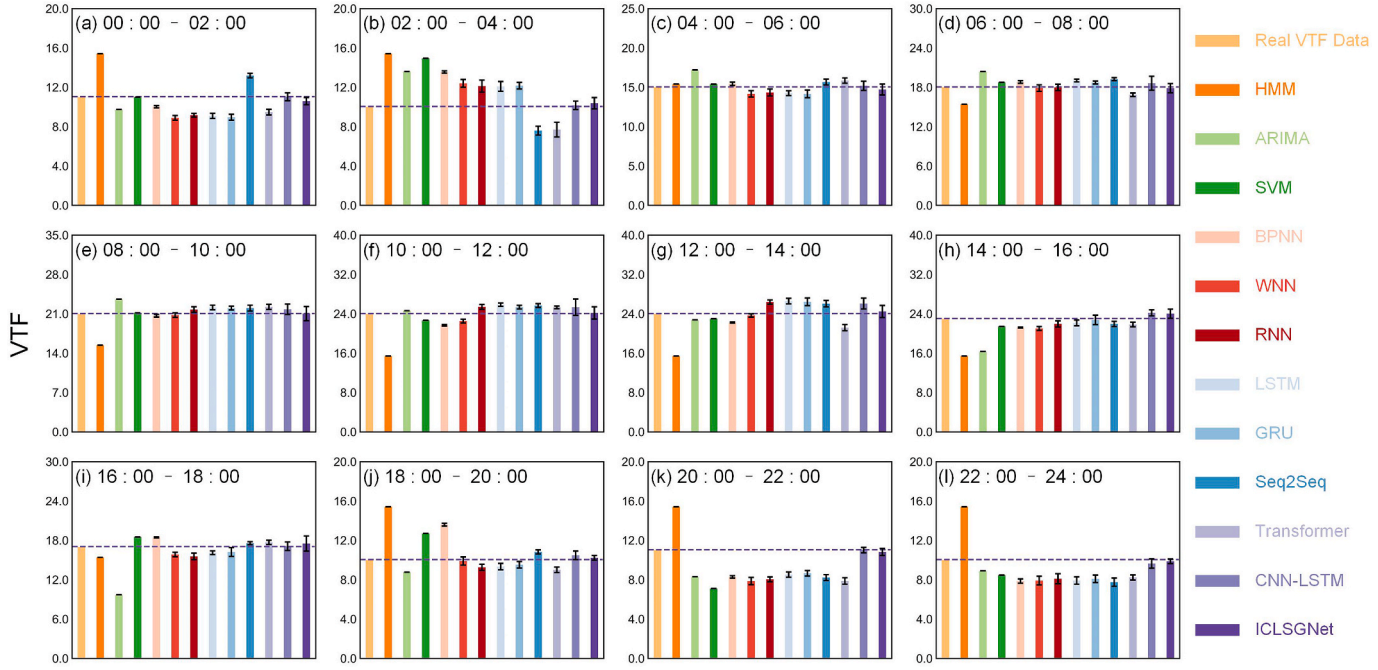

Fig. 21. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_4.

finding can be attributed to the fact that the cost function becomes challenging to minimise when the learning rate is too high or low.

Finally, this experiment analyses how iteration affects VTF prediction performance at the values of 200, 400, 600, 800, and 1000. The size of each neighbourhood matrix and learning rate parameters are set to 3 × 3 and 0.001. Based on the fine granularity experimental results, the proportions of obtaining the minimum REMean value are 16.67% (200), 25.00% (400), 26.42% (600), 21.49% (800), and 10.42% (1000), respectively, as shown in Fig. 16. Additionally, the proportions of getting the minimum REStd value are 17.71% (200), 19.79% (400), 23.96% (600), 22.92% (800), and 15.62% (1000), respectively, as shown in Fig. 17. The error visualisation results in Figs. 16 and 17 show that the best iteration parameter is 600.

To further determine the best iteration parameter, the results of RMSE and MAPE in five different iterations (i.e. 200, 400, 600, 800, and 1000) on September 30, 2020 based on our proposed ICLSGNet in 8 VTF datasets are clearly listed in Table 8. According to the experimental results at a coarse granularity, it is found that the prediction accuracy is the highest when the iteration is set at 400 in datasets C_4 and C_7. The iteration at 600 obtains the minimum RMSE and MAPE in most VTF datasets, as indicated in Table 8. The effect of iteration on the cost function during network training is comparable to that of the learning

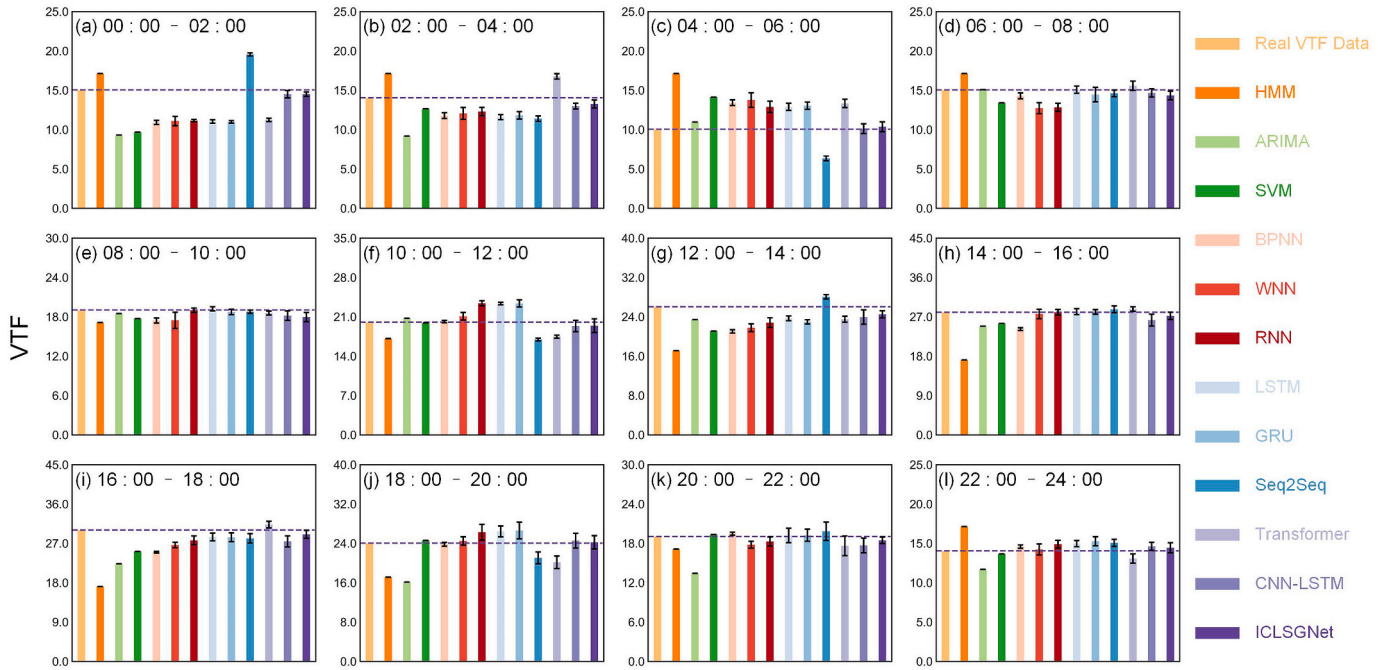

Fig. 22. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_5.

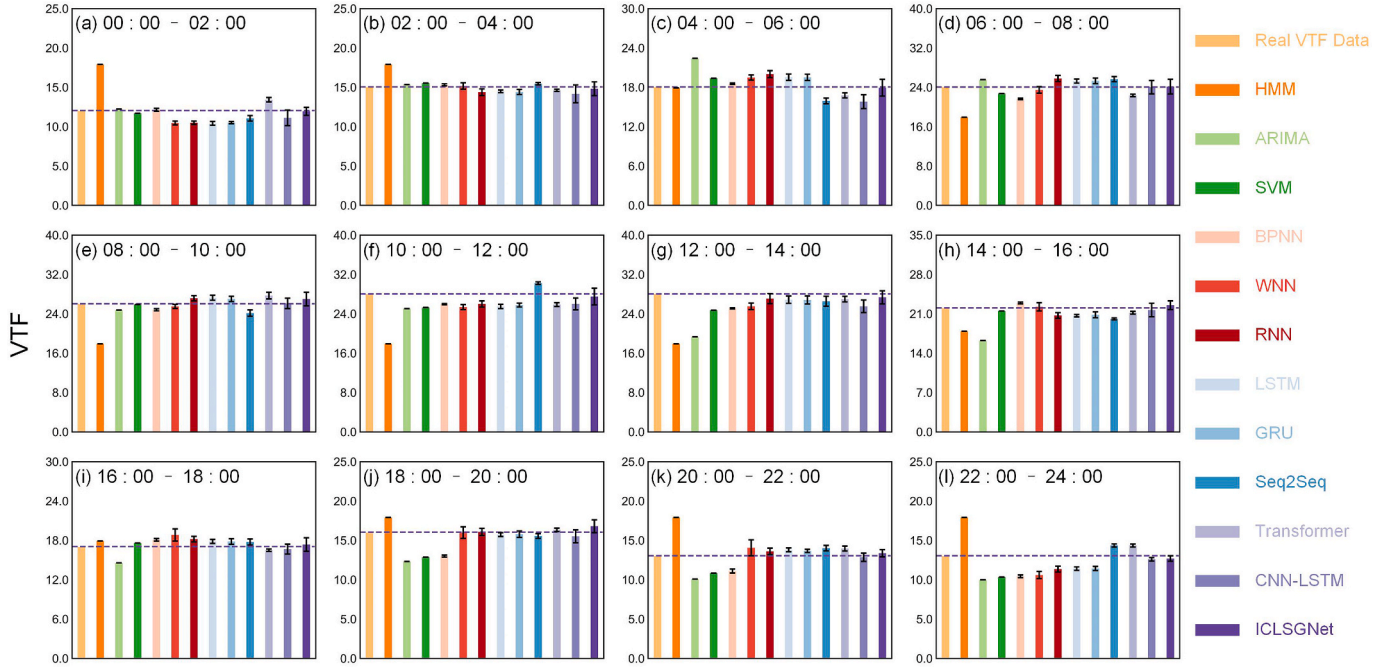

Fig. 23. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_6.

rate. If the iteration value is set too large, the gradient may drop below the lowest point, reducing training accuracy. Conversely, if the iteration value is too small, the cost function cannot reach the optimal local value in training.

The above comparative experiment demonstrates how three essential network parameters affect prediction accuracy and stability at various values. According to fine-grained evaluation results from Figs. 12–17, when the size of each neighbourhood matrix is set to 3, the accuracy and stability of the prediction results for the majority of time nodes reach the optimal level. Meanwhile, when the learning rate and the number of iterations are fixed at 0.001 and 600, respectively, the gradient in the training network can coverage to the lowest point in most scenarios, thus yielding prediction results of high-precision and stable prediction results. Furthermore, the quantitative evaluation results for coarse granularity (Tables 6–8) indicate that the optimal parameters of the size of each neighbourhood matrix, learning rate, and iteration are 3 × 3, 0.001, and 600, respectively. To ensure a fair comparison with other prediction methods in section 4.5.2, the values of general network parameters for the proposed ICLSGNet and eight other learning-based comparison methods should be kept consistent, listed in Table 9.

The Encoder-Decoder framework in Seq2Seq is constructed based on LSTM. Hence, its main parameters come from LSTM. The network

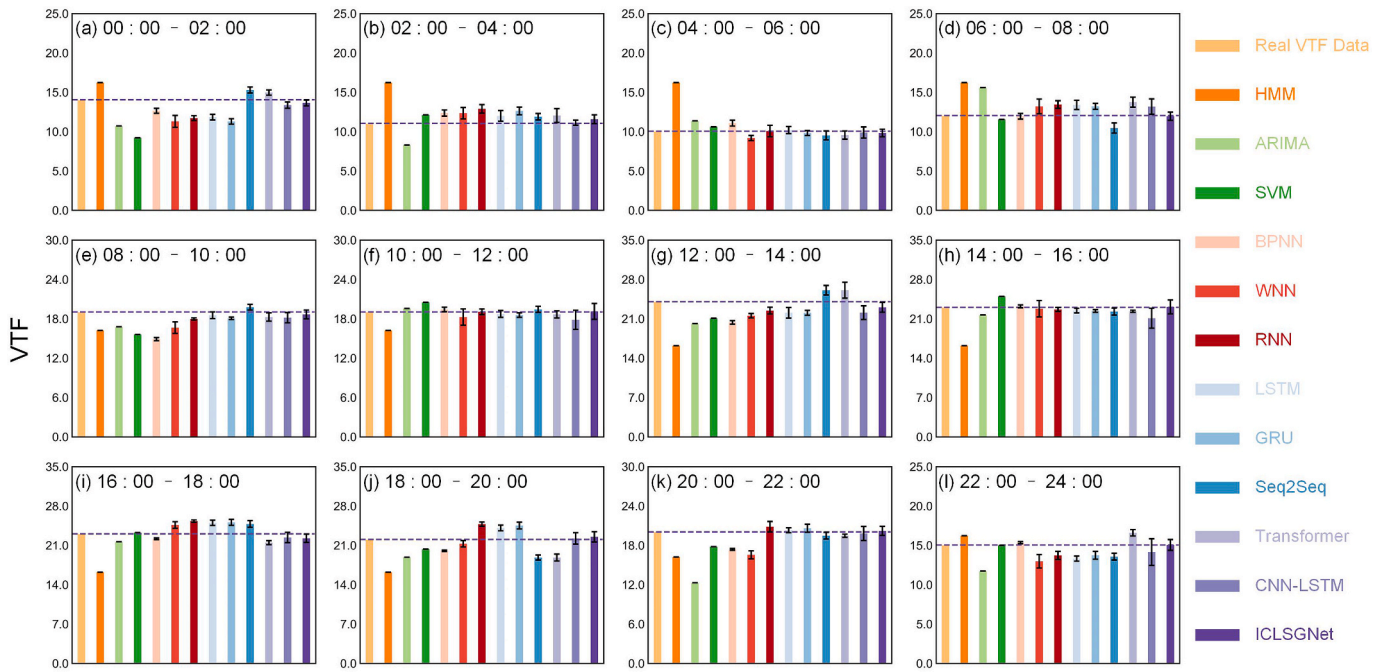

Fig. 24. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_7.

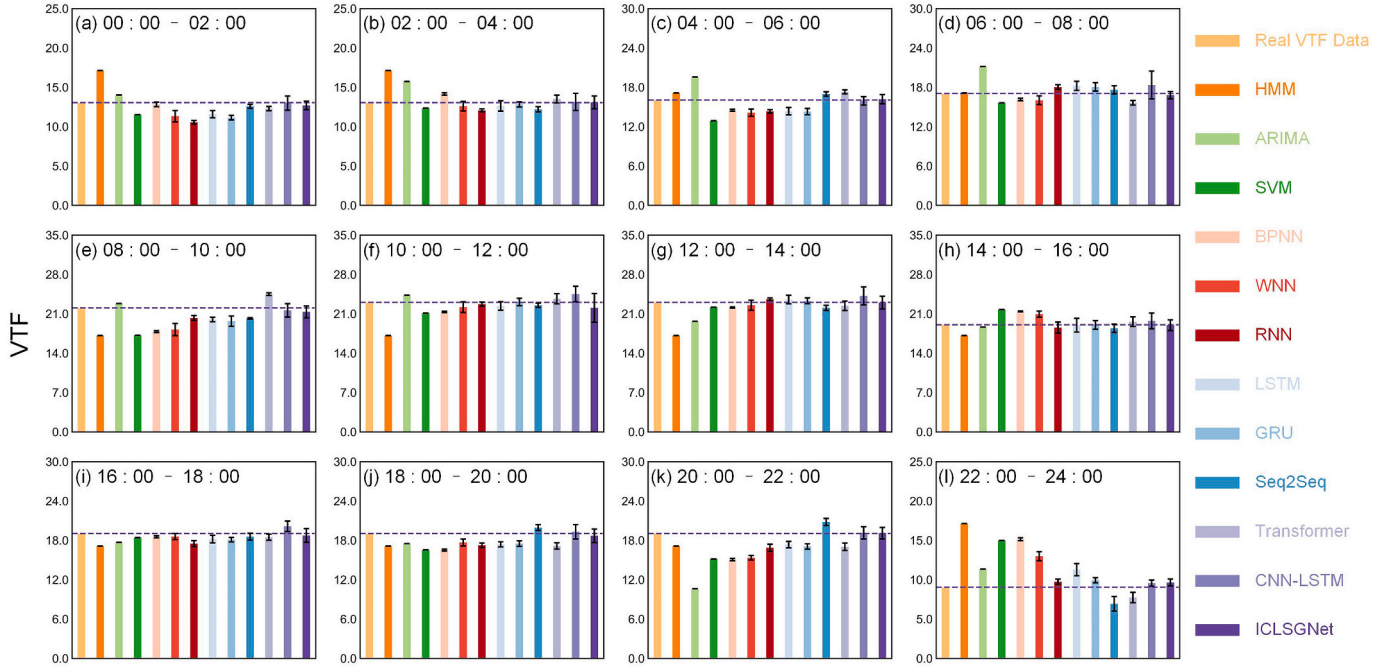

Fig. 25. The VTF prediction results (average value ± standard deviation) for 12 different periods from time point 1093 (00:00–02:00) to 1104 (22:00–24:00) on September 30, 2020 in dataset C_8.

parameters of CNN-LSTM are entirely the same as the spatial and temporal views of our proposed ICLSGNet method. The parameter 'Input Size' in BPNN and WNN corresponds to the 'Sequence Length' parameter in RNNs, Encoder for Seq2Seq, Transformer, and CNN-LSTM, which specifies the number of VTF data nodes that are inputted into the training network. The 'Input Size' parameter in RNNs, Encoder for Seq2Seq, and Transformer signify that each network unit receives 1 VTF data. The 'Output Size' parameter in BPNN, WNN, RNNs, Decoder for Seq2Seq, Transformer, and CNN-LSTM represents the predicted data at one time node in the future. Additionally, the loss function of all the above comparison methods is the same as that of the proposed ICLSGNet method, as outlined in 3.2.4.

4.5.2. *Prediction results of different methods*

According to the design logic of the comparative experiments in section 4.4, this section compares the proposed ICLSGNet method with eleven other classical methods to analyse the predicted performance of VTF. Firstly, this experiment evaluates the prediction accuracy of each method at different time nodes and datasets by comparing the average expected and actual values (as shown in Figs. 18–25) as well as REMean (i.e. Fig. 26). Secondly, the stability of the twelve prediction models is quantitatively assessed by the standard deviation of the predicted

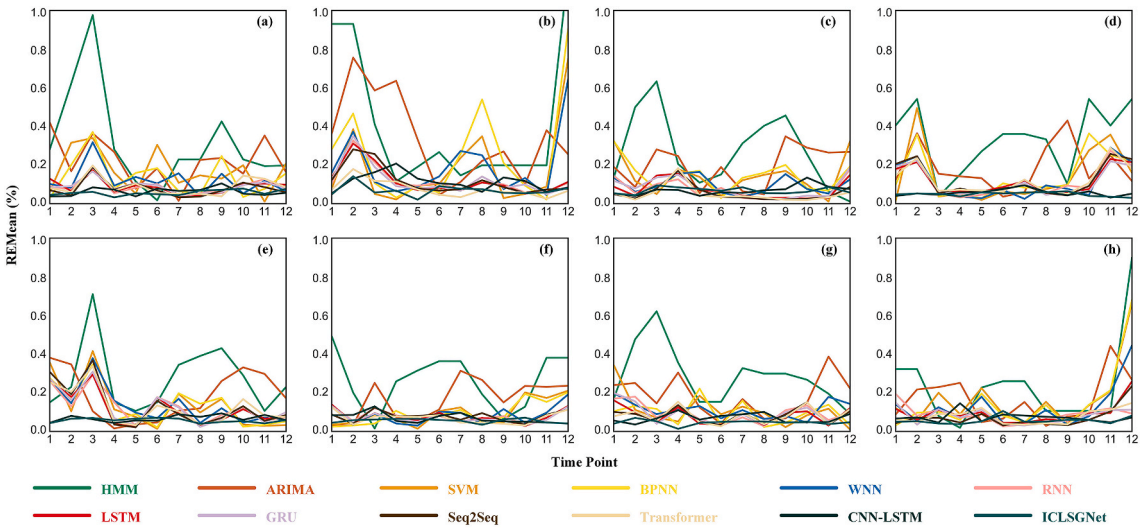

Fig. 26. Visual illustration of REMean for twelve different prediction methods at 12 periods on September 30, 2020 in 8 different VTF datasets. From (a) to (h) represent datasets from C_1 to C_8, respectively.

Table 10
The prediction results (i.e. RMSE and MAPE) of VTF on September 30, 2020 for twelve different methods in 8 VTF datasets.

Dataset	Evaluation Metrics	Methods											
		HMM	ARIMA	SVM	BPNN	WNN	RNN	LSTM	GRU	Seq2Seq	Transformer	CNN-LSTM	ICLSGNet
C_1	RMSE	6.0357	4.3445	3.0894	2.9107	2.2439	1.4739	1.4429	1.4639	1.3482	1.5884	0.8791	**0.4517**
	MAPE	0.3123	0.2130	0.1625	0.1291	0.1027	0.0802	0.0752	0.0785	0.0692	0.0799	0.0263	**0.0200**
C_2	RMSE	4.8401	4.6372	2.5890	3.4410	2.5639	1.4946	1.3895	1.4530	1.4503	0.9039	1.3935	**0.4753**
	MAPE	0.4134	0.3343	0.1860	0.2487	0.1730	0.1115	0.0773	0.1113	0.1102	0.0675	0.1055	**0.0309**
C_3	RMSE	6.9641	4.7755	3.3625	3.3461	2.1123	1.3362	1.2639	1.4583	0.9919	0.9482	1.3824	**0.9264**
	MAPE	0.2641	0.1877	0.1387	0.1362	0.0812	0.0569	0.0479	0.0695	0.0452	0.0407	0.0601	**0.0355**
C_4	RMSE	5.5702	3.4271	2.1884	2.1065	1.6504	1.6493	1.6707	1.5549	1.7226	1.7432	0.8789	**0.4066**
	MAPE	0.3337	0.1791	0.1407	0.1336	0.1067	0.1031	0.1096	0.1010	0.1164	0.1142	0.0345	**0.0198**
C_5	RMSE	6.4789	4.3815	2.9631	2.9024	2.4621	2.3137	2.1588	2.2778	2.3968	2.3410	1.0443	**0.7611**
	MAPE	0.2691	0.1781	0.1287	0.1232	0.1207	0.1148	0.1033	0.1112	0.1211	0.1194	0.0436	**0.0305**
C_6	RMSE	5.9080	3.8437	1.9130	1.8477	1.5390	1.3618	1.3398	1.2560	1.4687	1.2469	1.1880	**0.4959**
	MAPE	0.2577	0.1586	0.0860	0.0848	0.0692	0.0674	0.0660	0.0624	0.0697	0.0604	0.0433	**0.0190**
C_7	RMSE	5.0452	3.3625	2.2010	1.9685	1.8763	1.5783	1.3941	1.5633	1.4820	1.4673	1.0060	**0.4631**
	MAPE	0.2789	0.1734	0.0981	0.0976	0.0841	0.0797	0.0694	0.0795	0.0729	0.0722	0.0444	**0.0185**
C_8	RMSE	4.1631	3.3160	2.9654	2.7482	2.1919	1.4351	1.3578	1.3075	1.1348	1.3198	0.8213	**0.4111**
	MAPE	0.2285	0.1615	0.1555	0.1431	0.1158	0.0776	0.0767	0.0650	0.0634	0.0686	0.0353	**0.0185**

values, visually illustrated in Figs. 18–25. The statistical analysis of these fine-grained evaluation results in this experiment can generate several conclusions. The ICLSGNet method shows superior performance in dataset C_5, achieving the most accurate VTF prediction results for 33.34% of time nodes. The total proportion of the other eight learning-based methods yielding the best prediction results is 33.33%, while the modelling-based method receives the most precise prediction results for 33.33% of all cases in dataset C_5. In datasets C_4, C_6, and C_7, the ICLSGNet method surpasses the other 11 comparative methods, yielding the optimal prediction outcome in nearly 25% of the time nodes. Regarding datasets C_1, C_2, and C_8, the proposed ICLSGNet method obtains the optimal prediction results in approximately 17% of time nodes. It is worth mentioning that the new ICLSGNet method, alongside SVM and CNN-LSTM methods, exhibits excellent performance in executing VTF prediction tasks in these three datasets. In contrast, in dataset C_3, the precision of the proposed ICLSGNet method falls behind that of LSTM, GRU, Seq2Seq, Transformer and CNN-LSTM in predicting VTF for the third to ninth time nodes. In summary, the ICLSGNet method presents the best performance in predicting VTF data at different time nodes across eight datasets, with roughly 20% of the results exhibiting optimal performance. This ratio verifies that the new ICLSGNet is better than the other 11 comparative methods.

To provide a more intuitive comparison, this paper utilises coarse-grained evaluation indicators (i.e. RMSE and MAPE) to evaluate the VTF prediction accuracy of each dataset in one day, comprehensively taking into account the prediction results of 12 time nodes, as shown in Table 10. The importance of coarse-grained evaluation lies in its ability to prevent a method's exceptional or subpar performance at a specific time node from influencing the comprehensive evaluation results. After conducting a coarse granularity comparative analysis, it is found that the proposed ICLSGNet method exhibits the best prediction performance among the twelve methods in 8 datasets with the bold font from Table 10. Furthermore, it can be observed that LSTM has slightly better performance than GRU in most of the datasets (i.e. C_1, C_2, C_3, C_5, and C_7). Therefore, the encoder and decoder in the Seq2Seq method are both constructed using LSTM. Based on the evaluation results in Table 10, the Seq2Seq method outperforms LSTM only in datasets C_1, C_3 and C_8. The prediction performance of the Transformer is similar to LSTM, showcasing accurate results in datasets C_2, C_3, C_6 and C_8. Despite the optimisation in the network training structure of the Seq2Seq method, its execution of VTF prediction tasks falls slightly behind LSTM. The Transformer leverages a deep attention mechanism to optimise training networks based on Seq2Seq, and its predictive performance is close to that of LSTM. To simplify the network structure of the new method, this paper ultimately chose LSTM to construct the temporal and similarity grouping views of the ICLSGNet method.

To address the VTF prediction problem, modelling-based methods such as HMM, ARIMA, and SVM rely on historical data fitting to predict

future data. However, they may fail to accurately predict VTF data if it fluctuates significantly over time. Traditional NN methods like BPNN and WNN can better capture the change characteristics of historical data by building a training network and iteratively optimising it based on the loss function. They usually have better prediction performance than HMM, ARIMA and SVM methods. DL can learn irregular data features more effectively by building a deep training network. RNN, as the most classical DL method, can predict time series data (i.e. TF and VTF). However, the gradient of the RNN training network can disappear or explode during the continuous transmission of historical information, affecting the prediction results. LSTM and GRU are effective solutions to this problem, with LSTM containing three gates and GRU containing two gates. Hence, RNN predicts VTF data better than BPNN and WNN. LSTM and GRU perform better than RNN in completing prediction tasks. Seq2Seq, based on the Encoder-Decoder mechanism, can address the issue of maintaining consistency in the size of each group of data inputted into RNNs. Compared to Seq2Seq, the Transformer's most significant advantage is its flexible utilisation of attention mechanisms in training networks to enhance the ability to capture data feature changes. As a result, the Transformer typically outperforms Seq2Seq in predicting sequence data across most application scenarios. For the VTF prediction research conducted in this paper, the prediction capabilities of LSTM, Seq2Seq, and Transformer methods are relatively similar.

Since VTF has temporal and period attributes, RNNs, Seq2Seq, and Transformer cannot effectively learn the spatial attribute of VTF data. This issue can be addressed by transforming the original one-dimensional VTF into matrix data and inputting it into the CNN-LSTM training network. Thus, CNN-LSTM is more appropriate for predicting VTF data with spatiotemporal characteristics than RNNs. In real-world scenarios, multiple channels may exist in the study area, and they need to perform VTF prediction tasks simultaneously. Our proposed prediction method, ICLSGNet, can learn the temporal and spatial attributes of VTF data and interact with other channels with high similarity. Overall, ICLSGNet is well-suited to solve the VTF prediction problem of a water area consisting of multiple channels, and it exhibits good performance in terms of prediction accuracy and stability.

5. Conclusions and future research

VTF prediction is an important research direction of the maritime intelligent transportation system as it has numerous practical applications in the marine industry. For instance, regulatory authorities can use the VTF prediction results to determine the possibility of vessel congestion and potential collision risks in the future. Thus, accurate and reliable VTF prediction is crucial in theoretical research and practical applications. This paper proposes a new learning-based ICLSGNet method for VTF prediction, which has three views (i.e. spatial, temporal, and similarity grouping) and two primary advantages. Firstly, spatial and temporal views capture the cyclical and time development characteristics of VTF data, including the periodic property of VTF in the adjacent period. To effectively reflect the period attribute of VTF, this paper converts the original one-dimensional data into a two-dimensional matrix (hour of the day × day). The spatial view employs CNN to capture the spatial characteristics (i.e. period attribute) of the area (i.e. a time area) near a certain point in time, while the temporal view uses LSTM to extract the change characteristics of VTF data over time. Secondly, the similarity grouping view calculates the similarity between the historical VTF trends of the current predicted channel and other channels, finds the most similar channel, and learns the VTF historical change information based on LSTM.

Our proposed learning-based VTF prediction method, the ICLSGNet, fuses the learning information from spatial, temporal, and similarity grouping views into a fully connected network to obtain the final prediction result. It is particularly suited for areas with numerous channels, as it allows for interacting VTF information between channels that exhibit high similarity. The experiments with realistic VTF data from eight channels in the CJP water area compare the proposed ICLSGNet with modelling-based methods (i.e. HMM, ARIMA, and SVM) and learning-based methods (i.e. BPNN, WNN, RNN, LSTM, GRU, Seq2Seq, Transformer, and CNN-LSTM). The comparative experimental results demonstrate that the proposed ICLSGNet method has significant potential for VTF prediction in multiple channel water areas, with high accuracy and stability. Overall, the ICLSGNet could provide valuable theoretical support to regulatory authorities in predicting and analyzing VTF data across all channels in a target area.

To align the proposed VTF prediction method in this paper more closely with the real-world scenarios in the maritime industry, future research can incorporate qualitative factors that impact VTF changes into the training network. These factors could include vessel accidents and weather conditions such as fog, thunderstorms, and typhoons, which may affect the traffic flow of the channel in the future. Therefore, it is crucial to convert these qualitative factors into quantitative indicators and integrate them into the network training process. This optimised network can improve the accuracy of predicting future VTF changes in case of emergencies.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgements

This work is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 864724) and a Royal Society International Exchanges 2021 Cost Share (NSFC) project (IEC\NSFC \211211).

References

Ali, A., Zhu, Y., Zakarya, M., 2022. Exploiting dynamic spatio-temporal graph convolutional neural networks for citywide traffic flows prediction. Neural Network. 145, 233–247. https://doi.org/10.1016/j.neunet.2021.10.021.

Allen, D.M., 1971. Mean square error of prediction as a criterion for selecting variables. Technometrics 13, 469–475. https://doi.org/10.1080/00401706.1971.10488811.

Bi, J., Zhang, L., Yuan, H., Zhang, J., 2023. Multi-indicator water quality prediction with attention-assisted bidirectional LSTM and encoder-decoder. Inf. Sci. 625, 65–80. https://doi.org/10.1016/j.ins.2022.12.091.

Cao, S., Wu, L., Wu, J., Wu, D., Li, Q., 2022. A spatio-temporal sequence-to-sequence network for traffic flow prediction. Inf. Sci. 610, 185–203. https://doi.org/10.1016/j.ins.2022.07.125.

Chai, X., Tang, G., Wang, S., Lin, K., Peng, R., 2021. Deep learning for irregularly and regularly missing 3-D data reconstruction. IEEE Trans. Geosci. Rem. Sens. 59, 6244–6265. https://doi.org/10.1109/TGRS.2020.3016343.

Chen, X., Sun, L., 2022. Bayesian temporal factorization for multidimensional time series prediction. IEEE Trans. Pattern Anal. Mach. Intell. 44, 4659–4673. https://doi.org/10.1109/TPAMI.2021.3066551.

Chen, C., Hu, J., Meng, Q., Zhang, Y., 2011. Short-time traffic flow prediction with ARIMA-GARCH model. In: 2011 IEEE Intelligent Vehicles Symposium (IV). Presented at the 2011 IEEE Intelligent Vehicles Symposium (IV), pp. 607–612. https://doi.org/10.1109/IVS.2011.5940418.

Chen, K., Yao, H., Han, Z., 2022. Arithmetic optimization algorithm to optimize support vector machine for chip defect Identification. In: 2022 28th International Conference on Mechatronics and Machine Vision in Practice (M2VIP). Presented at the 2022 28th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), pp. 1–5. https://doi.org/10.1109/M2VIP55626.2022.10041106.

Comert, G., Bezuglov, A., 2013. An online change-point-based model for traffic parameter prediction. IEEE Trans. Intell. Transport. Syst. 14, 1360–1369. https://doi.org/10.1109/TITS.2013.2260540.

Dey, R., Salem, F.M., 2017. Gate-variants of gated recurrent unit (GRU) neural networks. In: 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS). Presented at the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 1597–1600. https://doi.org/10.1109/MWSCAS.2017.8053243.

Djenouri, Y., Belhadi, A., Srivastava, G., Lin, J.C.-W., 2023. Hybrid graph convolution neural network and branch-and-bound optimization for traffic flow forecasting. Future Generat. Comput. Syst. 139, 100–108. https://doi.org/10.1016/j.future.2022.09.018.

Do, L.N.N., Vu, H.L., Vo, B.Q., Liu, Z., Phung, D., 2019. An effective spatial-temporal attention based neural network for traffic flow prediction. Transport. Res. C Emerg. Technol. 108, 12–28. https://doi.org/10.1016/j.trc.2019.09.008.

Do, T.H., Minh Nguyen, D., Deligiannis, N., 2020. Graph auto-encoder for graph signal denoising. In: ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Presented at the ICASSP 2020 - 2020 IEEE International Conference on Acoustics. Speech and Signal Processing (ICASSP), pp. 3322–3326. https://doi.org/10.1109/ICASSP40776.2020.9053623.

Du, L., Gao, R., Suganthan, P.N., Wang, D.Z.W., 2022. Bayesian optimization based dynamic ensemble for time series forecasting. Information Sciences 591, 155–175. https://doi.org/10.1016/j.ins.2022.01.010.

Evans, J., Waterson, B., Hamilton, A., 2019. Forecasting road traffic conditions using a context-based random forest algorithm. Transport. Plann. Technol. 42, 554–572. https://doi.org/10.1080/03081060.2019.1622250.

Fei, J., Liu, L., 2022. Real-time nonlinear model predictive control of active power filter using self-feedback recurrent fuzzy neural network estimator. IEEE Trans. Ind. Electron. 69, 8366–8376. https://doi.org/10.1109/TIE.2021.3106007.

Fu, S., Zhong, S., Lin, L., Zhao, M., 2021. A re-optimized deep auto-encoder for gas turbine unsupervised anomaly detection. Eng. Appl. Artif. Intell. 101, 104199 https://doi.org/10.1016/j.engappai.2021.104199.

Fu, F., Deng, S., Wu, D., Liu, W., Bai, Z., 2022. Research on the spatiotemporal evolution of land use landscape pattern in a county area based on CA-Markov model. Sustain. Cities Soc. 80, 103760 https://doi.org/10.1016/j.scs.2022.103760.

Gao, R., Li, R., Hu, M., Suganthan, P.N., Yuen, K.F., 2023a. Dynamic ensemble deep echo state network for significant wave height forecasting. Applied Energy 329, 120261. https://doi.org/10.1016/j.apenergy.2022.120261.

Gao, R., Li, R., Hu, M., Suganthan, P.N., Yuen, K.F., 2023b. Online dynamic ensemble deep random vector functional link neural network for forecasting. Neural Networks 166, 51–69. https://doi.org/10.1016/j.neunet.2023.06.042.

Gao, M., Shi, G.-Y., 2019. Ship spatiotemporal key feature point online extraction based on AIS multi-sensor data using an improved sliding window algorithm. Sensors 19, 2706. https://doi.org/10.3390/s19122706.

Graves, A., 2012. Long short-term memory. In: Graves, A. (Ed.), Supervised Sequence Labelling with Recurrent Neural Networks, Studies in Computational Intelligence. Springer, Berlin, Heidelberg, pp. 37–45. https://doi.org/10.1007/978-3-642-24797-2_4.

Gu, Y., Lu, W., Xu, X., Qin, L., Shao, Z., Zhang, H., 2020. An improved bayesian combination model for short-term traffic prediction with deep learning. IEEE Trans. Intell. Transport. Syst. 21, 1332–1342. https://doi.org/10.1109/TITS.2019.2939290.

Gu, T., Wang, Z., Chi, Z., Zhu, Y., Du, W., 2021. Unsupervised cycle optimization learning for single-view depth and camera pose with Kalman filter. Eng. Appl. Artif. Intell. 106, 104488 https://doi.org/10.1016/j.engappai.2021.104488.

Haiyan, W., Youzhen, W., 2015. Vessel traffic flow forecasting with the combined model based on support vector machine. In: 2015 International Conference on Transportation Information and Safety (ICTIS). Presented at the 2015 International Conference on Transportation Information and Safety (ICTIS), pp. 695–698. https://doi.org/10.1109/ICTIS.2015.7232151.

Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., Wang, Y., 2021a. Transformer in transformer. In: Advances in Neural Information Processing Systems. Curran Associates, Inc., pp. 15908–15919

Han, Z., Zhao, J., Leung, H., Ma, K.F., Wang, W., 2021b. A review of deep learning models for time series prediction. IEEE Sensor. J. 21, 7833–7848. https://doi.org/10.1109/JSEN.2019.2923982.

Hao, S., Lee, D.-H., Zhao, D., 2019. Sequence to sequence learning with attention mechanism for short-term passenger flow prediction in large-scale metro system. Transport. Res. C Emerg. Technol. 107, 287–300. https://doi.org/10.1016/j.trc.2019.08.005.

Hara, K., Saito, D., Shouno, H., 2015. Analysis of function of rectified linear unit used in deep learning. In: 2015 International Joint Conference on Neural Networks (IJCNN). Presented at the 2015 International Joint Conference on Neural Networks. IJCNN, pp. 1–8. https://doi.org/10.1109/IJCNN.2015.7280578.

Hazarika, B.B., Gupta, D., Borah, P., 2021. An intuitionistic fuzzy kernel ridge regression classifier for binary classification. Appl. Soft Comput. 112, 107816 https://doi.org/10.1016/j.asoc.2021.107816.

He, W., Zhong, C., Sotelo, M.A., Chu, X., Liu, X., Li, Z., 2019. Short-term vessel traffic flow forecasting by using an improved Kalman model. Cluster Comput. 22, 7907–7916. https://doi.org/10.1007/s10586-017-1491-2.

Hecht-nielsen, R., 1992. III.3 - theory of the backpropagation neural Network**Based on “nonindent” by robert hecht-nielsen, which appeared in proceedings of the international joint conference on neural networks 1, 593–611, june 1989. © 1989 IEEE. In: Wechsler, H. (Ed.), Neural Networks for Perception. Academic Press, pp. 65–93. https://doi.org/10.1016/B978-0-12-741252-8.50010-8.

Hinton, G.E., Osindero, S., Teh, Y.-W., 2006. A fast learning algorithm for deep belief nets. Neural Comput. 18, 1527–1554. https://doi.org/10.1162/neco.2006.18.7.1527.

Hranisavljevic, N., Maier, A., Niggemann, O., 2020. Discretization of hybrid CPPS data into timed automaton using restricted Boltzmann machines. Eng. Appl. Artif. Intell. 95, 103826 https://doi.org/10.1016/j.engappai.2020.103826.

Hummels, D., 2007. Transportation costs and international trade in the second era of globalization. J. Econ. Perspect. 21, 131–154. https://doi.org/10.1257/jep.21.3.131.

Islam, MdJ., Ahmad, S., Haque, F., Reaz, M.B.I., Bhuiyan, M.A.S., Islam, MdR., 2022. Application of min-max normalization on subject-invariant EMG pattern recognition. IEEE Trans. Instrum. Meas. 71, 1–12. https://doi.org/10.1109/TIM.2022.3220286.

Jose, J.T., Das, J., Mishra, S.Kr, Wrat, G., 2021. Early detection and classification of internal leakage in boom actuator of mobile hydraulic machines using SVM. Eng. Appl. Artif. Intell. 106, 104492 https://doi.org/10.1016/j.engappai.2021.104492.

Kim, J.K., 2021. Semi-continuous spatial statistical analysis using AIS data for vessel traffic flow characteristics in fairway. J. Mar. Sci. Eng. 9, 378. https://doi.org/10.3390/jmse9040378.

Kim, T.-Y., Cho, S.-B., 2019. Predicting residential energy consumption using CNN-LSTM neural networks. Energy 182, 72–81. https://doi.org/10.1016/j.energy.2019.05.230.

Kong, D., Chen, Y., Li, N., Duan, C., Lu, L., Chen, D., 2020. Tool wear estimation in end milling of titanium alloy using NPE and a novel WOA-SVM model. IEEE Trans. Instrum. Meas. 69, 5219–5232. https://doi.org/10.1109/TIM.2019.2952476.

Kong, X., Xing, W., Wei, X., Bao, P., Zhang, J., Lu, W., 2020. STGAT: spatial-temporal graph attention networks for traffic flow forecasting. IEEE Access 8, 134363–134372. https://doi.org/10.1109/ACCESS.2020.3011186.

Kou, Z., Yang, F., Wu, J., Li, T., 2020. Application of ICEEMDAN energy entropy and AFSA-SVM for fault diagnosis of hoist sheave bearing. Entropy 22, 1347. https://doi.org/10.3390/e22121347.

Lee, S., Fambro, D.B., 1999. Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting. Transport. Res. Rec. 1678, 179–188. https://doi.org/10.3141/1678-22.

Lee, S.-H., Ku, H.-C., 2022. A dual attention-based recurrent neural network for short-term bike sharing usage demand prediction. IEEE Trans. Intell. Transport. Syst. 1–10. https://doi.org/10.1109/TITS.2022.3208087.

Li, C., Hu, J.-W., 2012. A new ARIMA-based neuro-fuzzy approach and swarm intelligence for time series forecasting. Eng. Appl. Artif. Intell. 25, 295–308. https://doi.org/10.1016/j.engappai.2011.10.005.

Li, H., Jiao, H., Yang, Z., 2023. AIS data-driven ship trajectory prediction modelling and analysis based on machine learning and deep learning methods. Transportation Research Part E: Logistics and Transportation Review 175, 103152. https://doi.org/10.1016/j.tre.2023.103152.

Li, Q., Lam, J.S.L., 2017. Conflict resolution for enhancing shipping safety and improving navigational traffic within a seaport: vessel arrival scheduling. Transportmetrica: Transport. Sci. 13, 727–741. https://doi.org/10.1080/23249935.2017.1326068.

Li, Y., Ren, H., 2022. Vessel traffic flow prediction using LSTM encoder-decoder. In: 2022 5th International Conference on Signal Processing and Machine Learning. Presented at the SPML 2022: 2022 5th International Conference on Signal Processing and Machine Learning. ACM, Dalian China, pp. 1–7. https://doi.org/10.1145/3556384.3556385.

Li, M.-W., Han, D.-F., Wang, W., 2015. Vessel traffic flow forecasting by RSVR with chaotic cloud simulated annealing genetic algorithm and KPCA. Neurocomputing 157, 243–255. https://doi.org/10.1016/j.neucom.2015.01.010.

Li, Y., Liu, R.W., Liu, Z., Liu, J., 2019. Similarity grouping-guided neural network modeling for maritime time series prediction. IEEE Access 7, 72647–72659. https://doi.org/10.1109/ACCESS.2019.2920436.

Li, H., Liu, J., Yang, Z., Liu, R.W., Wu, K., Wan, Y., 2020. Adaptively constrained dynamic time warping for time series classification and clustering. Inf. Sci. 534, 97–116. https://doi.org/10.1016/j.ins.2020.04.009.

Li, B., Luo, C., Wang, Z., 2020. Application of GWO-SVM algorithm in arc detection of pantograph. IEEE Access 8, 173865–173873. https://doi.org/10.1109/ACCESS.2020.3025714.

Li, H., Lam, J.S.L., Yang, Z., Liu, J., Liu, R.W., Liang, M., Li, Y., 2022. Unsupervised hierarchical methodology of maritime traffic pattern extraction for knowledge discovery. Transport. Res. C Emerg. Technol. 143, 103856 https://doi.org/10.1016/j.trc.2022.103856.

Li, Y., Li, Z., Mei, Q., Wang, P., Hu, W., Wang, Z., Xie, W., Yang, Y., Chen, Y., 2023. Research on multi-port ship traffic prediction method based on spatiotemporal graph neural networks. J. Mar. Sci. Eng. 11, 1379. https://doi.org/10.3390/jmse11071379.

Li, H., Yang, Z., 2023. Incorporation of AIS data-based machine learning into unsupervised route planning for maritime autonomous surface ships. Transportation Research Part E: Logistics and Transportation Review 176, 103171. https://doi.org/10.1016/j.tre.2023.103171.

Liang, M., Liu, R.W., Zhan, Y., Li, H., Zhu, F., Wang, F.-Y., 2022. Fine-grained vessel traffic flow prediction with a spatio-temporal multigraph convolutional network. IEEE Trans. Intell. Transport. Syst. 23, 23694–23707. https://doi.org/10.1109/TITS.2022.3199160.

Liu, X., Peng, H., Bai, Y., Zhu, Y., Liao, L., 2014. Tourism flows prediction based on an improved grey GM(1,1) model. Procedia - Social and Behavioral Sciences, The 9th International Conference on Traffic and Transportation Studies (ICTTS 2014) 138, 767–775. https://doi.org/10.1016/j.sbspro.2014.07.256.

Liu, R.W., Chen, J., Liu, Z., Li, Y., Liu, Y., Liu, J., 2017. Vessel traffic flow separation-prediction using low-rank and sparse decomposition. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). Presented at the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–6. https://doi.org/10.1109/ITSC.2017.8317741.

Liu, H., Zhao, H., Wang, J., Yuan, S., Feng, W., 2022. LSTM-GAN-AE: a promising approach for fault diagnosis in machine health monitoring. IEEE Trans. Instrum. Meas. 71, 1–13. https://doi.org/10.1109/TIM.2021.3135328.

Meyer, G.P., 2021. An alternative probabilistic interpretation of the huber loss. In: Presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5261–5269.

Millefiori, L.M., Braca, P., Bryan, K., Willett, P., 2016. Modeling vessel kinematics using a stochastic mean-reverting process for long-term prediction. IEEE Trans. Aero. Electron. Syst. 52, 2313–2330. https://doi.org/10.1109/TAES.2016.150596.

Mlepo, A.T., 2022. Attacks on road-freight transporters: a threat to trade participation for landlocked countries in Southern Africa. J Transp Secur 15, 23–40. https://doi.org/10.1007/s12198-021-00242-6.

Okutani, I., Stephanedes, Y.J., 1984. Dynamic prediction of traffic volume through Kalman filtering theory. Transp. Res. Part B Methodol. 18, 1–11. https://doi.org/10.1016/0191-2615(84)90002-X.

Pang, Y., Zhao, X., Yan, H., Liu, Y., 2021. Data-driven trajectory prediction with weather uncertainties: a Bayesian deep learning approach. Transport. Res. C Emerg. Technol. 130, 103326 https://doi.org/10.1016/j.trc.2021.103326.

Pope, P.E., Kolouri, S., Rostami, M., Martin, C.E., Hoffmann, H., 2019. Explainability methods for graph convolutional neural networks. In: Presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10772–10781.

Rahimikelarijani, B., Abedi, A., Hamidi, M., Cho, J., 2018. Simulation modeling of Houston Ship Channel vessel traffic for optimal closure scheduling. Simulat. Model. Pract. Theor. 80, 89–103. https://doi.org/10.1016/j.simpat.2017.10.004.

Rath, S., Tripathy, A., Tripathy, A.R., 2020. Prediction of new active cases of coronavirus disease (COVID-19) pandemic using multiple linear regression model. Diabetes Metabol. Syndr.: Clin. Res. Rev. 14, 1467–1474. https://doi.org/10.1016/j.dsx.2020.07.045.

Rong, H., Teixeira, A.P., Guedes Soares, C., 2022. Maritime traffic probabilistic prediction based on ship motion pattern extraction. Reliab. Eng. Syst. Saf. 217, 108061 https://doi.org/10.1016/j.ress.2021.108061.

Rubi, M., Chowdhury, S., Abdul Rahman, A.A., Meero, A., Zayed, N., Islam, K.M., 2022. Fitting multi-layer feed forward neural network and autoregressive integrated moving average for dhaka stock exchange price predicting. Emerging Science Journal 6, 1046–1061. https://doi.org/10.28991/ESJ-2022-06-05-09.

Safari, M.J.S., 2019. Decision tree (DT), generalized regression neural network (GR) and multivariate adaptive regression splines (MARS) models for sediment transport in sewer pipes. Water Sci. Technol. 79, 1113–1122. https://doi.org/10.2166/wst.2019.106.

Sherstinsky, A., 2020. Fundamentals of recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physica D: Nonlinear Phenomena 404, 132306. https://doi.org/10.1016/j.physd.2019.132306.

Su, Y., Yang, C., Qiao, J., 2023. Self-organizing pipelined recurrent wavelet neural network for time series prediction. Expert Syst. Appl. 214, 119215 https://doi.org/10.1016/j.eswa.2022.119215.

Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems. Curran Associates, Inc.

Tang, L., Tang, Y., Zhang, K., Du, L., Wang, M., 2019. Prediction of grades of ship collision accidents based on random forests and bayesian networks. In: 2019 5th International Conference on Transportation Information and Safety (ICTIS). Presented at the 2019 5th International Conference on Transportation Information and Safety (ICTIS), pp. 1377–1381. https://doi.org/10.1109/ICTIS.2019.8883590.

Van Der Voort, M., Dougherty, M., Watson, S., 1996. Combining kohonen maps with arima time series models to forecast traffic flow. Transport. Res. C Emerg. Technol. 4, 307–318. https://doi.org/10.1016/S0968-090X(97)82903-8.

Vatsa, A., Hati, A.S., 2022. Depolarization current prediction of transformers OPI system affected from detrapped charge using LSTM. IEEE Trans. Instrum. Meas. 71, 1–11. https://doi.org/10.1109/TIM.2022.3181286.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y., 2018. Graph attention networks. https://doi.org/10.48550/arXiv.1710.10903.

Vidya, B., Sasikumar, P., 2022. Parkinson's disease diagnosis and stage prediction based on gait signal analysis using EMD and CNN–LSTM network. Eng. Appl. Artif. Intell. 114, 105099 https://doi.org/10.1016/j.engappai.2022.105099.

Vuković, N., Mitić, M., Miljković, Z., 2015. Trajectory learning and reproduction for differential drive mobile robots based on GMM/HMM and dynamic time warping using learning from demonstration framework. Eng. Appl. Artif. Intell. 45, 388–404. https://doi.org/10.1016/j.engappai.2015.07.002.

Wang, K., Wang, H., Li, S., 2022. Renewable quantile regression for streaming datasets. Knowl. Base Syst. 235, 107675 https://doi.org/10.1016/j.knosys.2021.107675.

Wang, W., Li, B., Wang, H., 2022. A novel end-to-end network based on a bidirectional GRU and a self-attention mechanism for denoising of electroencephalography signals. Neuroscience 505, 10–20. https://doi.org/10.1016/j.neuroscience.2022.10.006.

Wang, Y., Jing, C., Xu, S., Guo, T., 2022. Attention based spatiotemporal graph attention networks for traffic flow forecasting. Inf. Sci. 607, 869–883. https://doi.org/10.1016/j.ins.2022.05.127.

Weerakody, P.B., Wong, K.W., Wang, G., Ela, W., 2021. A review of irregular time series data handling with gated recurrent neural networks. Neurocomputing 441, 161–178. https://doi.org/10.1016/j.neucom.2021.02.046.

Wen, Y., Xu, P., Li, Z., Xu, W., Wang, X., 2023. RPConvformer: a novel Transformer-based deep neural networks for traffic flow prediction. Expert Syst. Appl. 218, 119587 https://doi.org/10.1016/j.eswa.2023.119587.

Williams, B.M., Hoel, L.A., 2003. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results. J. Transport. Eng. 129, 664–672. https://doi.org/10.1061/(ASCE)0733-947X(2003)129:6(664).

Wu, Y., Wu, D., Fei, M., Sørensen, H., Ren, Y., Mou, J., 2023. Application of GA-BPNN on estimating the flow rate of a centrifugal pump. Eng. Appl. Artif. Intell. 119, 105738 https://doi.org/10.1016/j.engappai.2022.105738.

Xiao, R., Pan, L., Xiao, Hanbin, Xiao, Han, Zhu, Z., 2022. Research of intelligent logistics and high-quality economy development for Yangtze River cold chain shipping based on carbon neutrality. J. Mar. Sci. Eng. 10, 1029. https://doi.org/10.3390/jmse10081029.

Xiao, H., Zhao, Y., Zhang, H., 2023. Predict vessel traffic with weather conditions based on multimodal deep learning. J. Mar. Sci. Eng. 11, 39. https://doi.org/10.3390/jmse11010039.

Xie, Y., Niu, J., Zhang, Y., Ren, F., 2022. Multisize patched spatial-temporal transformer network for short- and long-term crowd flow prediction. IEEE Trans. Intell. Transport. Syst. 23, 21548–21568. https://doi.org/10.1109/TITS.2022.3186707.

Xu, T., Zhang, Q., 2022. Ship traffic flow prediction in wind farms water area based on spatiotemporal dependence. J. Mar. Sci. Eng. 10, 295. https://doi.org/10.3390/jmse10020295.

Xin, X., Liu, K., Loughney, S., Wang, J., Li, H., Ekere, N., Yang, Z., 2023a. Multi-scale collision risk estimation for maritime traffic in complex port waters. Reliability Engineering & System Safety 240, 109554. https://doi.org/10.1016/j.ress.2023.109554.

Xin, X., Liu, K., Loughney, S., Wang, J., Li, H., Yang, Z., 2023b. Graph-based ship traffic partitioning for intelligent maritime surveillance in complex port waters. Expert Systems with Applications 231, 120825. https://doi.org/10.1016/j.eswa.2023.120825.

Xu, L., Yu, X., Gulliver, T.A., 2021. Intelligent outage probability prediction for mobile IoT networks based on an IGWO-elman neural network. IEEE Trans. Veh. Technol. 70, 1365–1375. https://doi.org/10.1109/TVT.2021.3051966.

Yang, J., Bao, W., Liu, Y., Li, X., Wang, J., Niu, Y., Li, J., 2021. Joint pairwise graph embedded sparse deep belief network for fault diagnosis. Eng. Appl. Artif. Intell. 99, 104149 https://doi.org/10.1016/j.engappai.2020.104149.

Yao, L., Ge, Z., 2023. Causal variable selection for industrial process quality prediction via attention-based GRU network. Eng. Appl. Artif. Intell. 118, 105658 https://doi.org/10.1016/j.engappai.2022.105658.

Yao, H., Wu, F., Ke, J., Tang, X., Jia, Y., Lu, S., Gong, P., Ye, J., Li, Z., 2018. Deep multi-view spatial-temporal network for taxi demand prediction. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32. https://doi.org/10.1609/aaai.v32i1.11836.

Yu, J., Tang, G., Song, X., Yu, X., Qi, Y., Li, D., Zhang, Y., 2018. Ship arrival prediction and its value on daily container terminal operation. Ocean Eng. 157, 73–86. https://doi.org/10.1016/j.oceaneng.2018.03.038.

Yu, Q., Liu, K., Teixeira, A.P., Soares, C.G., 2020. Assessment of the influence of offshore wind farms on ship traffic flow based on AIS data. J. Navig. 73, 131–148. https://doi.org/10.1017/S0373463319000444.

Yu, G., Ma, J., Xie, C., 2023. Hessian scatter regularized twin support vector machine for semi-supervised classification. Eng. Appl. Artif. Intell. 119, 105751 https://doi.org/10.1016/j.engappai.2022.105751.

Zhang, J., Walter, G.G., Miao, Y., Lee, W.N.W., 1995. Wavelet neural networks for function learning. IEEE Trans. Signal Process. 43, 1485–1497. https://doi.org/10.1109/78.388860.

Zhang, H., Ke, J., Dong, S., Yuan, Z., 2019. Simulation modeling of state-dependent queuing network: impact of deepening on the vessel traffic in Yangtze River Estuary. Adv. Mech. Eng. 11, 1687814019845247 https://doi.org/10.1177/1687814019845247.

Zhang, L., Meng, Q., Fang Fwa, T., 2019. Big AIS data based spatial-temporal analyses of ship traffic in Singapore port waters. Transport. Res. E Logist. Transport. Rev. 129, 287–304. https://doi.org/10.1016/j.tre.2017.07.011.

Zhang, Z., Yin, J., Wang, N., Hui, Z., 2019. Vessel traffic flow analysis and prediction by an improved PSO-BP mechanism based on AIS data. Evolving Systems 10, 397–407. https://doi.org/10.1007/s12530-018-9243-y.

Zhang, F., Tang, X., Tong, A., Wang, B., Wang, J., 2020. An automatic baseline correction method based on the penalized Least squares method. Sensors 20, 2015. https://doi.org/10.3390/s20072015.

Zhang, K., Xin, Y., Xie, Z., Shi, C., 2023. A swimming crab portunus trituberculatus re-identification method based on RNN encoding of striped key regions. Eng. Appl. Artif. Intell. 120, 105900 https://doi.org/10.1016/j.engappai.2023.105900.

Zhao, C., Li, X., Zuo, M., Mo, L., Yang, C., 2022a. Spatiotemporal dynamic network for regional maritime vessel flow prediction amid COVID-19. Transport Pol. 129, 78–89. https://doi.org/10.1016/j.tranpol.2022.09.029.

Zhao, J., Chen, Y., Zhou, Z., Zhao, Jingying, Wang, S., Chen, X., 2022b. Extracting vessel speed based on machine learning and drone images during ship traffic flow prediction. J. Adv. Transport. 2022, e3048611 https://doi.org/10.1155/2022/3048611.

Zhao, Q., Li, L., Zhang, L., Zhao, M., 2023. Recognition of corrosion state of water pipe inner wall based on SMA-SVM under RF feature selection. Coatings 13, 26. https://doi.org/10.3390/coatings13010026.

Zhong, J., Huyan, J., Zhang, W., Cheng, H., Zhang, J., Tong, Z., Jiang, X., Huang, B., 2023. A deeper generative adversarial network for grooved cement concrete pavement crack detection. Eng. Appl. Artif. Intell. 119, 105808 https://doi.org/10.1016/j.engappai.2022.105808.

Zhou, X., Liu, Z., Wang, F., Xie, Y., Zhang, X., 2020. Using deep learning to forecast maritime vessel flows. Sensors 20, 1761. https://doi.org/10.3390/s20061761.

Zhu, H., Liu, C., Wu, W.-Z., Xie, W., Lao, T., 2022. Weakened fractional-order accumulation operator for ill-conditioned discrete grey system models. Appl. Math. Model. 111, 349–362. https://doi.org/10.1016/j.apm.2022.06.042.