Traffic & Transportation

Vessel Trajectory Prediction Method Based on the Time Series Data Fusion Model

Xinyun WU[1], Jiafei CHEN[2], Caiquan XIONG[3], Donghua LIU[4], Xiang WAN[5], Zexi CHEN[6]

Original Scientific Paper
Submitted: 19 June 2024
Accepted: 1 Oct 2024

[1] xinyun@hbut.edu.cn, Hubei University of Technology
[2] chjf6699@163.com, Hubei University of Technology
[3] xiongcq@hbut.edu.cn, Hubei University of Technology
[4] Corresponding author, liudonghua@wti.ac.cn, China Waterborne Transport Research Institute
[5] wanwuhan@yeah.net, Wuhan Second Ship Design & Research Institute
[6] jerseychen998@gmail.com, Xiaomi Technology (Wuhan) Co., Ltd

Publisher:
Faculty of Transport and Traffic Sciences, University of Zagreb

ABSTRACT

Vessel trajectory prediction is important in maritime traffic safety and emergency management. Vessel trajectory prediction using vessel automatic identification system (AIS) data has attracted wide attention. Deep learning techniques have been widely applied to vessel trajectory prediction tasks due to their advantages in fine-grained feature learning and time series modelling. However, most deep learning-based methods use a unified approach for modelling AIS data, ignoring the diversity of AIS data and the impact of noise on prediction performance due to environmental factors. To address this issue, this study introduces a method consisting of temporal convolutional network (TCN), convolutional neural network (CNN) and convolutional long short-term memory (ConvLSTM) to predict vessel trajectories, called TCC. The model employs TCN to capture the complex correlation of the time series, utilises CNN to capture the fine-grained covariate features and then captures the dynamics and complexity of the trajectory sequences through ConvLSTM to predict vessel trajectories. Experiments are conducted on real public datasets, and the results show that the TCC model proposed in this paper outperforms the existing baseline algorithms with high accuracy and robustness in vessel trajectory prediction.

KEYWORDS

Automatic Identification System (AIS) data; vessel trajectory prediction; deep learning; neural network.

1. INTRODUCTION

Maritime transport is one of the most important global trade transports, undertaking the task of trans-oceanic transportation of most goods. With vigorous development of the global economy, maritime activities have been increasing, especially in busy seas, and the water traffic environment has become more and more complex bringing great challenges to maritime vessel traffic management [1] and navigation safety [2]. Therefore, accurate vessel trajectory prediction has become an indispensable part of marine transportation and maritime management [3]. AIS (Automatic Identification System) [4] is a system for the automatic identification and localisation of vessels, which records the vessel's activities at different times and locations, and can provide its dynamic information (current position, current speed to the ground, current heading to the ground, etc.), navigation information and static information of nearby vessels (maritime mobile service identification number, vessel name, etc.), which provides rich data for vessel trajectory prediction.

In recent years, many vessel trajectory prediction studies have been carried out based on AIS data. Earlier, based on traditional mathematical models [5] and physical principles, the basic principles of Newtonian mechanics, nonlinear differential equations, and other methods were applied to model the vessel's trajectory by combining the speed, heading, position and velocity information in the vessel's AIS data, and then the linear interpolation was used to predict the vessel's position [6–9]. However, these methods usually rely on predefined motion patterns, are less robust to noise and error data in AIS data, lacking the ability to respond flexibly to autonomous vessel behaviours and changes in the external environment, thus limiting their application in dynamic and complex marine environments. Later, some researchers implemented machine learning methods such as Support Vector Regression (SVR) [10], and k-Nearest Neighbour Algorithm (k-NN) [11] to model the vessel's motion state and predict the future vessel's position by learning the patterns in the historical AIS data, but when dealing with the large-scale AIS data, the computational and storage requirements are high, the parameter selection is complicated and has limited adaptability to nonlinear dynamic environments.

With the deepening research of deep learning techniques in computer vision, natural language processing and other fields, some researchers started to use deep neural networks to model AIS to achieve the prediction of vessel trajectories. For example, Volkova [12] et al. tried to verify the possibility of vessel trajectory prediction based on the AIS dataset using neural network models. Kim [13–14] et al. used a convolutional neural network (CNN) to extract features from latitude, longitude and other covariate information of AIS data for the prediction of future vessel trajectories. Although CNN can effectively extract spatial features from vessel trajectory data, including the vessel's motion speed, heading change, etc., it is more suitable for extracting local features. However, its capture of global long-term dependencies is relatively limited and may not be able to fully utilise the global information of the vessel's trajectory. Yang [15–16] et al. implemented recurrent neural networks (RNNs) to extract vessel coordinates from AIS data. Features such as speed and heading in AIS data were encoded appropriately thereby realising the prediction of the vessel's future position. However, due to the limited memory units of RNN, it is difficult to capture long-term dependencies in long sequential data, which may lead to performance degradation. Gao [17–18] et al. used a Long Short-Term Memory (LSTM) network to model the vessel's coordinate information to predict the future trajectory of the vessel. The performance of the LSTM model highly depends on the quality and completeness of the input data. The prediction may also be affected by the absence or incorrect data. Wang [19] et al. combined CNN and LSTM to predict the future trajectory of a vessel. They used a CNN module for extracting data on the relation vessel between different variables (e.g. latitude, longitude, speed and ground course) and an LSTM module for capturing temporal dependencies. However, one of the key shortcomings of their approach was trying to capture all the features of the available information and model them comprehensively, leading to a significant increase in the model complexity. This indiscriminate modelling of all information not only increases model complexity but also introduces redundancy and noise that may hinder the model's ability to identify relevant patterns and make accurate predictions. Furthermore, the model may struggle with overfitting, especially when trained on constrained or noisy data, affecting its generalisation performance over unseen real-world trajectories.

Although there have been some achievements in research regarding vessel trajectory prediction based on deep neural network modelling AIS data, there are limitations in the existing research. For example, Zhang [20] et al. proposed a time-series convolutional network model to achieve the prediction of vessels in inland waterways or to show relevant prediction performance over a longer period in specific situations, and it is difficult to predict accurately the behaviour under extreme weather conditions. Second, the complexity of the marine environment and the interaction between vessels and the ocean may lead to noises and relying solely on latitude and longitude information may result in a model unable to capture accurately the characteristics of the data. It may result in large errors [21]. Finally, some researchers use all AIS data directly as inputs, ignoring its heterogeneity. This makes it difficult to capture all the features of AIS data or capture many irrelevant or redundant features through only a single model, leading to performance degradation [22]. In addition, too many features may increase the complexity of the model, making it more difficult to learn effective patterns. Different categories of data stocks may exhibit different features, and the inability to distinguish between different categories may lead to performance degradation of the model [23].

Therefore, when it comes to the navigation situation of vessels at sea, the vessel trajectory data is often affected by various interferences, and even missing latitude and longitude information may occur. So we need a model that can make full use of other information in the dataset to judge the vessel's navigation status. Moreover, the robustness and adaptability of the vessel trajectory prediction model need to be further improved to adapt to the ever-diversifying application scenarios.

To solve the above problems, an innovative vessel trajectory prediction model named TCN-CNN-ConvLSTM model, referred to as the TCC model, is proposed in this paper. The model is based on neural network to realise the modelling of different types of AIS data, separate the data with different focuses in the AIS dataset and finally predict the vessel trajectory through feature fusion [24]. Specifically, for the time series data generated in vessel voyages that contain long-term temporal dependencies, we use TCN to capture the long-term dependencies of the time series. For the covariate information in vessel navigation, we use CNN for modelling to capture fine-grained vessel features. Finally, we implement a feature fusion to predict the behaviour of the vessel accurately. Our main contributions are as follows:

1) We propose a new hybrid network model, TCC, which utilises TCN [25] and CNN models to achieve modelling of time series and feature extraction of covariate information. Our model captures the long-term temporal dependencies by using TCN, and feature extraction of covariate information by using CNN which captures the vessel's navigational state information. Random weights are added to refine the feature representation to capture the vessel navigation features comprehensively.
2) ConvLSTM [26] is utilised to learn the tensor information encoded by TCN and CNN networks, giving our model the ability to predict future vessel trajectories. It combines the benefits of TCN for capturing long-term temporal dependencies with the ability of CNN to extract features of vessel sailing states.
3) The proposed method is compared with commonly used methods based on real AIS datasets and state-of-the-art ones. The results show that the accuracy and robustness of the proposed method are higher than the existing models.

2. TCC MODEL

In this section, the general architecture of the TCC model is described. The model consists of TCN, CNN and ConvLSTM, which we divided into three modules. The principles and roles of each module are described in detail separately. Based on the vessel's AIS dataset, the proposed method predicts the vessel's trajectory using time-series data fusion, optimised for the characteristics of the data.

2.1 General Model Architecture

For vessel trajectory prediction, it is assumed that there is a sequence of consecutive vessel trajectories $l_1, l_{2,} \cdots, l_t, \cdots, l_T$ of length T, where l_t consists of the AIS information of LAT, LON, SOG, COG, and Heading of the vessel at the moment t. The aim is to predict the vessel's trajectory coordinates $\hat{l}_{T+1}, \hat{l}_{T+2}, \cdots, \hat{l}_{T+n}, \cdots, \hat{l}_{T+N}$ within the latter $T+N$ moments, where $\hat{l}_{T+n}$ denotes the vessel's coordinates at the $\hat{l}_{T+n}$ moment and consists of LAT and LON. Therefore, the task can be represented as *Formula 1*.

$$\hat{l}_{T+1}, \hat{l}_{T+2}, \cdots, \hat{l}_{T+n}, \cdots, \hat{l}_{T+N} = f(l_1, l_{2,} \cdots, l_t, \cdots, l_T) \tag{1}$$

The AIS data is firstly segmented into two parts based on the characteristics of the vessel's trajectory. The first part is the vessel's latitude and longitude sequence l, including the vessel's longitude (LON) and latitude (LAT). The second part is the vessel's covariate information c, including surface speed (SOG), surface heading (COG) and true heading angle (Heading).

The sequence of vessel's latitude and longitude coordinates l with history length T is then modelled using the TCN neural network to capture the long-term dependencies. Meanwhile, the second module uses a convolutional neural network CNN to extract features from the vessel covariates c of history length T to mine useful information.Convolution layer extracts patterns in covariate data through local convolution operations and generates feature maps. The extracted features then pass through a linear layer and are mapped to the desired output dimension, thus integrating with the output of the TCN module. Finally, the encoded tensor sequences are inputted into the ConvLSTM module to predict the future vessel trajectories.

The vessel trajectory prediction process is shown in *Figure 1*, detailing the description of each module. More detailed processing and parameterisation of each module is described in the following sections 2.2 to 2.4.

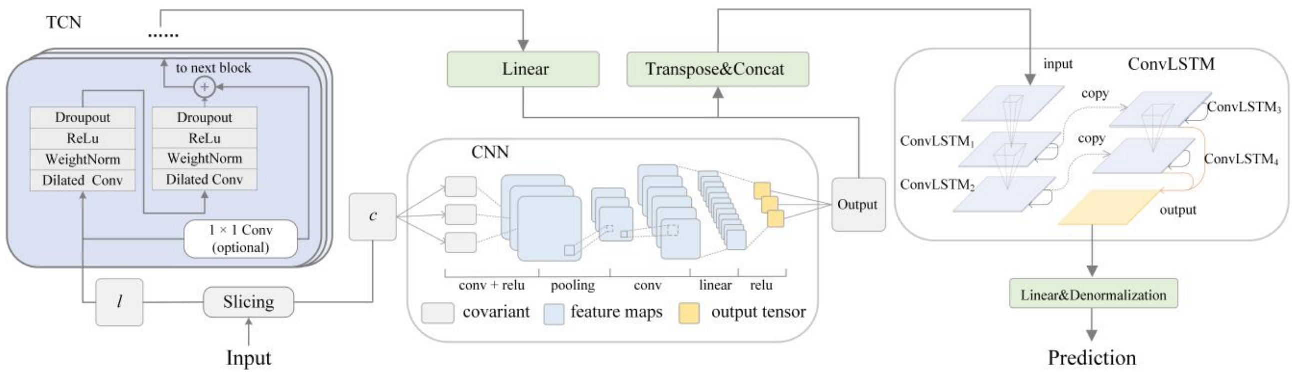


Figure 1 – The overall architecture of the TCC model. l and c denote the position coordinates (LAT, LON) and covariate information (COG, SOG, Heading) of the vessel at time T, respectively.

2.2 Module 1: TCN Trajectory Modelling Module

For the vessel trajectory coordinate sequence, we use the TCN Trajectory Modelling Module to model the long-term dependence of the longitude and latitude sequence. It enhances the model's ability to deal with long-term dependencies by introducing innovative structures such as dilation convolution.

In this network, the number of input features is set to 2 (longitude and latitude). The network consists of three convolutional layers, the number of output channels for each layer is set to 128, and the size of the convolutional kernel is 7. In addition, the dilation rate of the convolutional layer increases layer by layer, starting at 1, and each layer is 1, 2 and 4. The increasing expansion rate of each layer allows the model to capture long-term dependencies at different levels, enhancing the ability to process sequence data, while balancing the model's learning power and computational complexity with appropriate convolution kernel size and the settings of channel number to efficiently process the features of the input data.

By sampling the input at intervals during the convolution process, dilated convolution allows the model to expand the receptive field with a limited number of parameters and thus better capture long-term dependencies. We consider each longitude and latitude as a time step. For a sequence $L \epsilon \{L_0, L_{1,} \cdots, L_t, \cdots, L_{T-1}\}$ of vessel trajectories in time length T, L_t consists of the vessel's longitude and latitude at moment t, the expansion convolution operation F can be expressed as *Formula 2*.

$$F(t) = \sum_{i=0}^{k-1} f(i) \cdot L_{t-d \cdot i} \tag{2}$$

where d is the dilation rate, k is the convolutional kernel size and $k - d \cdot i$ is the past direction. After that, a series of operations are performed, including adding random weights, regularisation and using activation functions to generate the output of a complete residual block. Then the output of the previous residual block is used as the input of the next residual block, and each residual block is connected in sequence. The series of transformations in the residual block is recorded as φ and the result of the residual connection is recorded as $L_{T.}$ The process is expressed as *Formula 3.*

$$L_T = Activation(\varphi(x) + x) \tag{3}$$

Finally, a linear layer mapping is used to generate the final tensor sequence. The final tensor sequence modelled by the TCN Trajectory Modelling Module is denoted as L_m , expressed as *Formula 4.*

$$L_m = Linear(L_T) \tag{4}$$

2.3 Module 2: CNN Feature Extraction Module

The vessel trajectory data is first prepared into a format suitable for CNN. In this module, a one-dimensional convolution layer is used to process context information. The number of input channels is 3, indicating that three features such as speed, heading and true heading are processed, and the number of output channels is also 3. The size of the convolution kernel is set to 2, which defines the window size of the convolution operation.

Each sample has three features, i.e. SOG, COG and Heading. We consider the above information as channels and then apply a Convolutional Neural Network (CNN) for feature extraction. Each feature corresponds to a feature map and the length of the input sequence determines the size of the feature map. In the convolutional layer, local features can be extracted from the sequence data by setting a convolutional kernel of an appropriate size. The convolution kernel performs a convolution operation with the input sequence to produce a feature map. The convolution operation moves in the horizontal direction to capture features at different locations. The output of the convolutional layer is nonlinearly transformed using the ReLU activation function and a bias term is added to the final convolutional operation to make the model more flexible in fitting the data, thus enhancing the expressive power of the network. The convolution operation can be expressed as *Formula 5.*

$$Y_{i,j} = (C * K)(i,j) \sum \sum C(i+m, j+m) K(m,n) + b_f \tag{5}$$

where C is the input sequence, K represents the convolution kernel and Y is the output after convolution. Notation b_f is the bias term of the fth convolution kernel. Notations i and j represent the position coordinates in y, respectively. Notations m and n represent the position coordinates of the convolution kernel, respectively.

The value of each position of the output feature map Y represents the local features of the input sequence. The one-dimensional convolution operation is repeated at different positions of the input data until the entire input sequence is covered, thereby obtaining various local features of the entire input sequence.

After the convolution operation, the data is fed into the pooling layer to reduce the size of the feature map and retain the main feature information, while reducing the number of parameters and the amount of computation.

The convolutional and activation function layers are grouped into one layer and stacked after the pooling operation to build a deeper feature extraction network. This allows the gradual extraction of higher-level feature representations. The data is fed into a fully connected layer to map the pooled data features to the target value space. Finally, a linear layer was added, and a deeper mapping using the activation function ReLU was adjusted to the shape of the data that could be collocated with the TCN network for input into the ConvLSTM network.

Overall, the added linear layer achieves further mapping and feature extraction of the encoded speed, heading and bow direction information by learning weights and bias terms to generate the final feature representation of the speed information.

2.4 Module 3: ConvLSTM Trajectory Prediction Module

The network structure of ConvLSTM can be viewed as an LSTM network with a convolution kernel, where the convolution kernel is used to perform convolution operations on the input data. At each time step, the convolution operation scans the entire sequence in the temporal dimension to capture the spatio-temporal information.

In this ConvLSTM network, the number of feature graph channels is set to 5, and the network has 2 layers of ConvLSTM, the first layer of which has a hidden state dimension of 32 and the second layer of which has a hidden state dimension of 3. Each convolution layer uses a 3×3 convolution kernel. The space size of the input data is 16×16.

ConvLSTM can extract both spatial and temporal features of spatio-temporal data to capture spatio-temporal dependencies in vessel trajectory prediction. Especially for problems that need to consider motion and temporal variations in the vessel trajectory, ConvLSTM combines the memory units of LSTM to capture these long-term dependencies effectively, thus improving the accuracy of vessel trajectory prediction. The following is the detailed processing:

The vessel trajectory information extracted by TCN modelling and CNN features is spliced and adapted into a shape suitable for the ConvLSTM network input, which is represented as a spatio-temporal sequence $X = \{x_1, x_2, \cdots, x_t, \cdots, x_T\}$ where x_t denotes the vessel tensor data at the moment t a five-dimensional tensor data. At each time step, the input data is passed to the ConvLSTM model. The ConvLSTM model consists of multiple ConvLSTM layers. Each ConvLSTM layer consists of a set of Convolutional Kernels and a set of

Gate Units. The Convolutional Kernels are used to process the spatio-temporal features of the input data. The Gate Units are used to control the updating and forgetting of the Memory Units. The input data affects the state and hidden state of the Memory Units through convolutional and gating operations. At each time step, the ConvLSTM model updates the memory cell states and hidden states of the current time step based on the current input data and the ones of the previous time step. These state updates are done through a series of convolution operations, element-by-element multiplications and activation functions. At each time step, we record the hidden state output of the last ConvLSTM layer and repeat this process for the entire sequence until all time steps have been processed. Finally, we added the fully connected layer (linear layer) to map the hidden state to the final prediction, denoted as *Formula 6.*

$$\hat{y} = Linear(h_t) \tag{6}$$

where $\hat{y}$ is the prediction result and $Linear$ is the linear layer operation.

2.5 Loss function

We adopted the commonly used Mean Squared Error (MSE) as the loss function for our trajectory prediction model. The rationale for this choice is the significant impact of changes in coordinates. For example, a 1-degree difference in latitude or longitude can result in an actual distance difference of over 100 nautical miles. Given the substantial amount of vessel data in our dataset, MSE is particularly effective because it squares the errors during computation, thereby penalising larger errors (outliers) more severely. This characteristic ensures that the model focuses on reducing substantial errors, enhancing its robustness and performance even with the presence of anomalies.

Furthermore, MSE is typically calculated after normalising the predicted data. In certain scenarios, if the value of the loss function is too small, the gradients may approach zero. This leads to the vanishing gradient problem, where the model fails to update its parameters effectively, thereby impeding the learning process. In our approach, we addressed this issue by modifying the loss function. Specifically, we adjusted the total loss value to be the sum of the MSE of the denormalised trajectory coordinates and the MSE of the normalised trajectory coordinates.

This method is designed to better reflect the model’s performance on actual trajectory coordinates and ensure consistent optimisation across different data scales. By considering both the accuracy of the model’s predictions and its performance in the actual trajectory space, the proposed approach enhances the model’s effectiveness and generalisation ability in trajectory prediction tasks. The calculation for the loss function is shown in *Formula 7.*

$$\mu_{MSE} = \frac{1}{T}\sum_{s=1}^{T}\left[\left(p_{T+s}^{pred} + p_{T+s}^{true}\right)^2 + \left(n_{T+s}^{pred} + n_{T+s}^{true}\right)^2\right] \tag{7}$$

where p_{T+s}^{pred} is the value representing the true longitude or latitude of the model output; p_{T+s}^{true} is the value representing the true longitude or latitude of the vessel; n_{T+s}^{pred} is the value representing the true longitude or latitude normalised by the model output; n_{T+s}^{true} is the value representing the true longitude or latitude normalised by the vessel; and μ_{MSE} represents the final MSE result.

3. EXPERIMENTS AND RESULTS

In this section, we conduct experiments on the predictive effectiveness of different models using real AIS data to demonstrate the superiority of the proposed model. In Section 3.4, we conduct an ablation experiment on the model to verify the effectiveness of Modules 1 and 2.

3.1 Dataset

We obtained AIS data covering Hawaiian waters from the online portal of the U.S. Marine Cadastre (https://marinecadastre.gov). During the initial clean up, we paid special attention to a key type of vessel (Vessel Nos. 70-79) that carried significant cargo in order to gain a deeper understanding of cargo vessel activity in Hawaiian waters. In the next step of our data cleaning, we further weeded out data that had navigational malfunctions, such as anchoring or stopping midway through a voyage. Such filtering helps ensure that our dataset is more realistic and reliable. In order to generate a smoother and more regularised trajectory dataset, we used linear interpolation to resample the time and regularise the time intervals to 3-minute intervals, thus eliminating some irregularities and noise in the data.

The raw AIS information has many types of information, and in this paper, the fields that may have an impact on the vessel's trajectory are extracted. This includes Maritime Mobile Service Identity (MMSI), Full UTC date and time (BaseDateTime), Latitude (LAT), Longitude (LON), Speed Over Ground (SOG), Course Over Ground (COG) and True Heading Angle (Heading).

We divided the processed AIS dataset in the ratio of training set (80%) and test set (20%). The training dataset covers the period from 1 January 2022 to 28 February 2022, with 294 complete trajectory groups and 66,752 coordinate points. The test dataset is from 1 March 2022 to 29 April 2022 with 90 complete trajectory groups and 16,323 coordinate points.

In this study, each trajectory coordinate point is a real trajectory coordinate with contextual information. The vessel AIS information of consecutive 2, 4, 6 and 8 trajectory coordinate points are used as inputs. The model outputs the latitude and longitude values of the vessel trajectory at the latter 2, 4, 6 and 8 time points, i.e. the data of the first 6, 12, 18 and 24 minutes are used to predict the trajectory data of the latter 6, 12, 18 and 24 minutes, respectively. The batch size is 8. The initial learning rates are all 0.001 and the number of training rounds is 150. Vessel trajectory prediction experiments were conducted on a Windows 10 system with a single NVIDIA GeForce GTX 1080Ti. PyTorch was used for model construction. A stochastic objective function optimisation algorithm (Adam) was used to adjust the learning rate according to the gradient of each parameter, and the parameters were dynamically updated during the training process.

3.2 Evaluation metrics

Evaluation metrics for trajectory prediction are important criteria for assessing the deviation and accuracy of the prediction results from the true trajectory. We used the metrics Mean Absolute Error (MAE), Final Displacement Error (FDE), Average Displacement Error (ADE), which are commonly used in the field of trajectory prediction [29-30].

The Haversine [31] formula calculates the Euclidean distance between latitude and longitude. The formula is based on the cosine theorem of a spherical triangle and calculates the curvilinear distance between two points. The formula is expressed in the form of a great circle distance, which represents the shortest distance between two points on the Earth's surface. The mathematical expression is shown in *Formula 8* (in nautical miles).

$$d = 2 \cdot r \cdot arcsin\left(\sqrt{sin^2\left(\frac{\Delta lat}{2}\right) + cos(lat_1) \cdot cos(lat_2) \cdot sin^2\left(\frac{\Delta lon}{2}\right)}\right) \cdot 1.852 \tag{8}$$

where d is the spherical distance between two points on the Earth and r is the spherical distance between two points on the Earth averaged over the radius of the Earth (approximately 6371 kilometers); lat_1 and lat_2 are the latitudes (in radians) of the two points; Δlat is the difference between the latitudes of the two points; and Δlon is the difference between the longitudes of the two points.

MAE is the average absolute value of the prediction error, which is used to measure the overall prediction accuracy of the model. For vessel trajectory prediction, MAE reflects the average deviation of the predicted trajectory from the actual trajectory at all points. For a trajectory containing n points, MAE can be expressed as *Formula 9.*

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - t_i| \tag{9}$$

where p_i is the value of the longitude or latitude of the point i of the predicted trajectory and t_i is the value of the longitude or latitude of the point i of the true trajectory.

FDE is the distance between the last point of the predicted trajectory and the actual end point. It is specifically used to measure the accuracy of the model in predicting the final position. FDE can be expressed as *Formula 10.*

$$FDE = Haversine(p_l, t_l) \tag{10}$$

where p_l is the last coordinate point of the predicted trajectory and t_l is the last coordinate point of the true trajectory.

ADE is the average offset error between the predicted trajectory and the actual trajectory. It calculates the distance from each point in the predicted trajectory to the corresponding real point and then averages the values. ADE can be expressed as *Formula 11*.

$$ADE = \frac{1}{n}\sum_{i=1}^{n} Haversine(p_i, t_i) \tag{11}$$

3.3 Experimental results and analysis

In this section, we use the real AIS dataset to experiment and discuss the effects of each model. Section 3.3.1 is the metrics results of all models on the real AIS dataset and Section 3.3.2 uses different models to perform trajectory prediction visualisation analysis on real vessel data.

Experimental results

In order to validate the TCC model proposed in this paper, three classical models in the field of vessel trajectory are selected, which are SVR [10], BiLSTM [17] and ConvLSTM [26], and two advanced models are also selected, which are ConvLSTM_Seq2Seq model [27] and LSTM_RNN [28]. We calculate the training time and testing time of all models. The test time here is for the entire test dataset, which contains 90 sets of tracks, and the results are shown in Table 1. The unit is minute.

Table 1 – Results for train and test time over time steps for all models

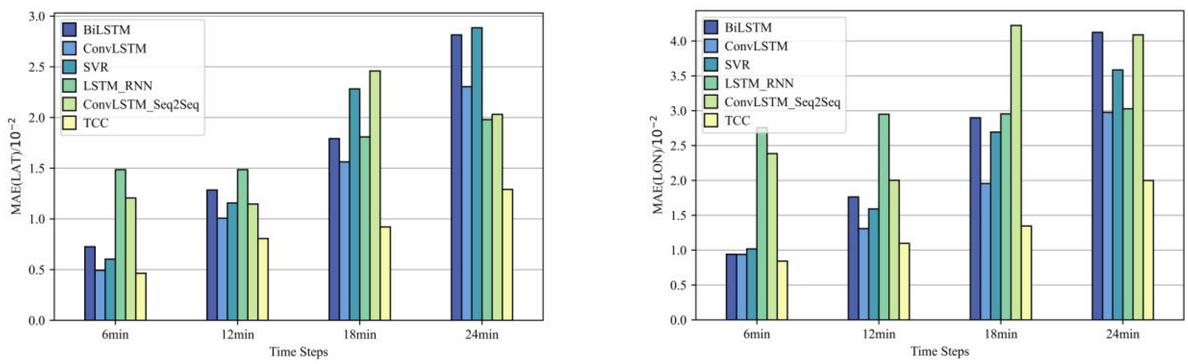
Metric	Time step	TCC	ConvLSTM	BiLSTM	SVR	LSTM_RNN	ConvLSTM _Seq2seq
train	6min	1397.42	680.23	565.10	563.78	607.34	759.43
	12min	1525.17	842.50	598.75	577.55	644.68	823.79
	18min	1562.50	950.22	623.46	628.72	678.99	845.25
	24min	1627.21	1017.50	797.37	636.25	691.11	847.52
test	6min	2.38	1.59	1.08	1.02	2.48	2.95
	12min	2.50	2.02	1.12	1.03	2.41	3.61
	18min	3.01	2.39	1.18	1.08	2.47	3.86
	24min	3.12	3.04	1.21	1.13	2.46	4.41

As can be seen from *Table 1*, although the training time of the TCC model is longer than that of other models, its testing time is not the longest. Therefore, if the TCC model is applied to the real-time prediction of a single trajectory, its effect is still considerable. In order to prove the validity and accuracy of the TCC model, we further carried out validation experiments.

The results of ADE and FDE are shown in *Table 2,* and the MAE results in latitude and longitude directions are shown in *Figure 2a* and *Figure 2b*, respectively. One can observe that the difference between the metrics results of all the models is not very large with a time step of 6 minutes. The ADE and MAE metrics of the TCC model exceed the other models only by a small advantage, while the SVR model is slightly better in terms of the FDE metrics, but its advantage is only 0.0352 nautical miles less than that of the TCC model. This situation may arise because there is relatively less information about the vessel state in the case of a shorter time step, and therefore less influence on the future vessel trajectory. It also implies that the TCN modelling module and the CNN feature extraction module may not use their potential advantages at this time step. As the prediction time step increases, we can observe that the prediction errors of all models show a gradual increase. This trend may be partly due to the increased complexity of the prediction task because of the increased time step, such that longer prediction time steps imply more uncertainty and variability, which in turn increases the prediction error. However, it is encouraging to note that the TCC model maintains its superior performance even when the prediction time step increases. First, the TCC model outperforms other models in all evaluation metrics, including average distance error (ADE), final distance error (FDE) and mean absolute error (MAE), for prediction time steps of 12, 18 and 24 minutes.

Table 2 – Results for the ADE and FDE metrics over time steps for all models

Metric	Time step	TCC	ConvLSTM	BiLSTM	SVR	LSTM_RNN	ConvLSTM_Seq2seq
ADE	6min	0.5856	0.6479	0.7657	0.7310	0.8963	0.9000
	12min	0.8920	1.0513	1.3859	1.2696	1.4827	1.4250
	18min	1.0319	1.6127	2.1041	2.2487	2.1482	2.5507
	24min	1.4950	2.3745	3.1008	2.9885	2.9585	2.7854
FDE	6min	0.7664	0.8359	0.8718	0.7312	0.8873	1.0011
	12min	1.2700	1.5320	1.4752	1.2728	1.4816	1.4809
	18min	1.6560	2.5276	2.2899	2.2590	2.1458	2.7778
	24min	2.4628	3.6234	3.2825	2.9970	3.0352	2.8296



(a) *(b)*

Figure 2 – a) MAE results for different time steps of latitude; b) MAE results for different time steps of longitude

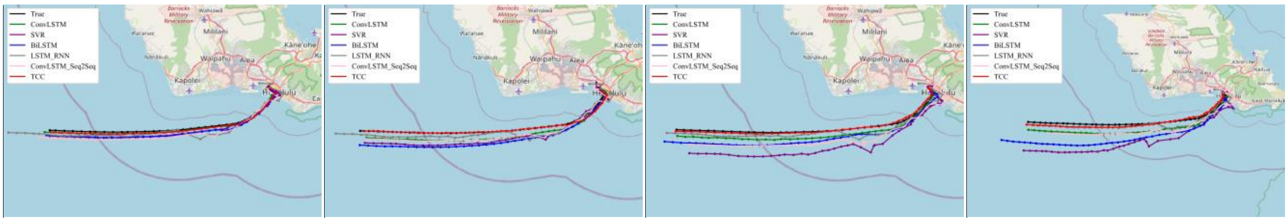
It is particularly noteworthy that in terms of the ADE metrics, the error of the TCC model grows by only 0.1399 nautical miles when the time step increases from 12 to 18 minutes, while the other models show a much larger increase in error. The reason may be that as the prediction time step increases, the feature information carried by the trajectory sequence becomes richer, but the task complexity of trajectory prediction also increases. The increase in time step increases the feature information carried by the trajectory sequence. This makes the TCN modelling module more accurate in modelling longitude and latitude information. It also enriches the CNN feature extraction module's understanding of vessel speed, heading and steering angle information, thereby more accurately capturing the vessel's feature information. When dealing with more complex tasks, other models do not make full use of AIS information at longer prediction time steps, nor do they specifically use the rich vessel navigation characteristics in AIS data, resulting in larger error increments for other models. The error growth of other models is also larger. This shows that the TCC model has better robustness and stability than other baseline models and can better adapt to different prediction time periods, thus maintaining lower prediction errors.

Trajectory visualisation of different models

We selected two vessels with different sailing conditions and predicted real vessel trajectories using the TCC model as well as other models for conditions with time steps of 6, 12, 18 and 24 minutes. The complete trajectories were constructed iteratively, containing 48 trajectory points with a total time step of 144 minutes, and the comparison results of the trajectories are presented in *Figures 3a–3d* and *Figures 4a–4d*. The black line segment 'True' represents the real vessel trajectory, and the red, green, purple, blue, grey and pink colours represent the trajectories predicted by the neural network models TCC, ConvLSTM, SVR, BiLSTM, LSTM_RNN, and ConvLSTM_Seq2Seq, respectively.

(a) (b) (c) (d)

Figure 3 – a), b), c) and d) are the predicted results of different models for vessel 1 with time step of 6, 12, 18, 24 minutes respectively



(a) (b) (c) (d)

Figure 4 – a), b), c) and d) are the predicted results of different models for vessel 2 with time step of 6, 12, 18, 24 minutes respectively

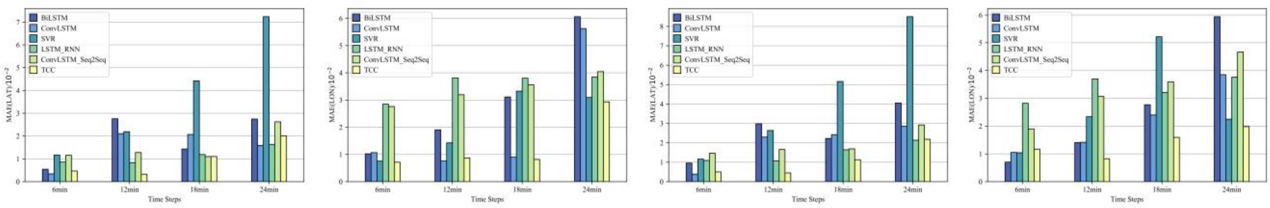
From the experimental results, we find that the TCC model and other baseline models can predict the future sailing trend of the vessel at different time steps. This indicates that all models have the ability to predict vessel trajectory, capturing the characteristics and trends of vessel navigation to some extent. Comparing the trajectory prediction results of vessel 1 in *Figures 3a–3d* and vessel 2 in *Figures 4a–4d*, we can clearly see the prediction effects of different models at different time steps. As the time step increases, the difference in predicted trajectories is gradually revealed, which may be because the vessel trajectory changes less in a shorter period of time, whereas it may be affected by more factors in a longer period of time, and thus the performance of the model varies in different time periods. For a more detailed analysis, we calculated more detailed metrics results for the predictions of vessel 1 and vessel 2 separately. *Table 3* and *Table 4* show the detailed metrics results for vessel 1 and vessel 2, respectively, for all models under different forecasting time step conditions. *Figures 5a–5b* and *Figures 5c–5d* are the results of calculating the MAE for vessel 1 and vessel 2, respectively. For the prediction trajectory of vessel 1, under the condition of a prediction time step of 6 minutes, the SVR model exhibited the poorest performance. The final point of the predicted trajectory of LSTM_RNN exceeds the actual trajectory by a large margin, which may be caused by the fact that it fails to capture the change in the vessel's speed. The TCC model slightly outperforms the others by a weak advantage, and the difference of predicted trajectories between the models is not significant. This circumstance may be attributed to the fact that the navigation trajectory of vessel 1 is predominantly linear, with minimal variations in vessel status.

Table 3 – Predictive performance results of different models for vessel 1

Metric	Time step	TCC	ConvLSTM	BiLSTM	SVR	LSTM_RNN	ConvLSTM_Seq2seq
ADE	6min	0.5158	0.6415	0.6747	0.8525	1.8289	1.7973
	12min	0.5700	1.3662	2.1515	1.6946	2.2631	2.0774
	18min	0.9340	1.4182	2.0066	3.5395	2.3586	2.2336
	24min	2.0514	3.3680	3.8638	4.6798	2.4922	3.1515
FDE	6min	0.5890	0.7381	0.8124	0.8539	2.0448	1.5177
	12min	0.7516	1.8366	2.0471	1.7303	2.4725	2.1672
	18min	1.8386	2.3022	2.4757	3.6407	2.3140	2.3214
	24min	3.1106	5.6325	4.2193	4.7436	2.6054	3.8013

Table 4 – Predictive performance results of different models for vessel 2

Metric	Time step	TCC	ConvLSTM	BiLSTM	SVR	LSTM_RNN	ConvLSTM_Seq2seq
ADE	6min	0.7427	0.6535	0.7284	0.9534	1.9285	1.511
	12min	0.6003	1.6233	2.1282	2.1743	2.2580	2.1673
	18min	1.0312	2.2192	2.3626	3.5916	2.1566	2.4029
	24min	1.7570	3.0040	4.2240	5.3269	2.628	3.0804
FDE	6min	1.0415	0.7330	0.8301	0.9558	1.979	0.9531
	12min	0.7822	2.1597	1.7526	2.2197	2.1184	1.9530
	18min	1.9712	3.5878	2.5600	3.6738	2.405	2.3870
	24min	2.5745	4.8363	4.4395	5.4840	2.7191	3.429



(a) (b) (c) (d)

Figure 5 – a) and b) represent the MAE performance results of different models for vessel 1 in latitude and longitude, respectively; c) and d) represent the MAE performance results of different models for vessel 2 in latitude and longitude, respectively

Consequently, the TCC model failed to fully exploit its potential, aligning with the findings in Section 3.3.1 of the performance metrics. As the prediction time step increased, in the prediction of vessel 2, the vessel's acceleration and turning behaviours upon exiting the harbour posed challenges to trajectory prediction, resulting in varying degrees of deviation across all trajectories. As illustrated in *Figures 3c–3d* and *Figures 4c–4d*, with the increase in prediction time step, the deviation of trajectories predicted by the models under comparison increased; however, the TCC model consistently maintained its superior performance. It is particularly notable when vessel 2 initially departs from the harbour and executes turning manoeuvres, where the richer information embedded in the AIS data contributes to superior prediction results compared to other models, as evidenced in *Figures 4c–4d*. One possible explanation lies in the enrichment of navigation information within AIS data as the time step increases, enabling the modelling and feature extraction modules of the TCC model to capture and accurately identify such scenarios. Based on the analysis results shown in *Figures 5a–5d*, as the time step increases, the MAE errors of all models also increase. However, the TCC model shows a significantly lower increase in errors compared to the other models. Particularly for vessel 2 trajectories characterised by turning manoeuvres, the modelling and feature extraction modules within the TCC model effectively capture state information, resulting in more precise trajectory predictions compared to other models. Thus, from the testing results, the TCC model outperforms other models in predicting vessel trajectories especially with increasing time steps.

3.4 Ablation experiment

To verify the effectiveness of adding the feature extraction module, we set up an ablation experiment by removing the feature extraction module and the Trajectory Modelling Module to test their effectiveness in improving the accuracy of vessel trajectory prediction. We label the model with the TCN Trajectory Modelling Module removed from the TCC as TCC_1 and the TCC model with the CNN feature extraction module removed from the TCC as TCC_2.

Results

We set the time step to 6min, 12min, 18min or 24min respectively, and used the TCC model, TCC_1 and TCC_2 to test all the data in the test set in four groups. The results of the average value of each metric in the overall test set are in *Table 5* and *Table 6*.

Table 5 – ADE and FDE results of different models

Metric	**Time step**	**TCC**	**TCC_1**	**TCC_2**
ADE	6min	0.5856	0.4933	0.7425
	12min	0.8920	0.9635	1.1225
	18min	1.0319	1.4136	2.2633
	24min	1.4950	1.6857	1.8953
FDE	6min	0.7664	0.6255	0.8656
	12min	1.2700	1.4911	1.4936
	18min	1.6560	2.1775	3.1154
	24min	2.4628	2.8677	3.1399

Table 6 – MAE(LAT) and MAE(LON) results of different model

Metric	**Time step**	**TCC**	**TCC_1**	**TCC_2**
MAE(LAT)	6min	0.4641	0.4103	0.6322
	12min	0.8066	0.7716	1.1097
	18min	0.9206	1.1167	2.7425
	24min	1.2909	1.3975	1.5886
MAE(LON)	6min	0.8432	0.6618	1.0239
	12min	1.0980	1.3535	1.4052
	18min	1.3471	1.9616	2.1392
	24min	1.9989	2.3914	2.6305

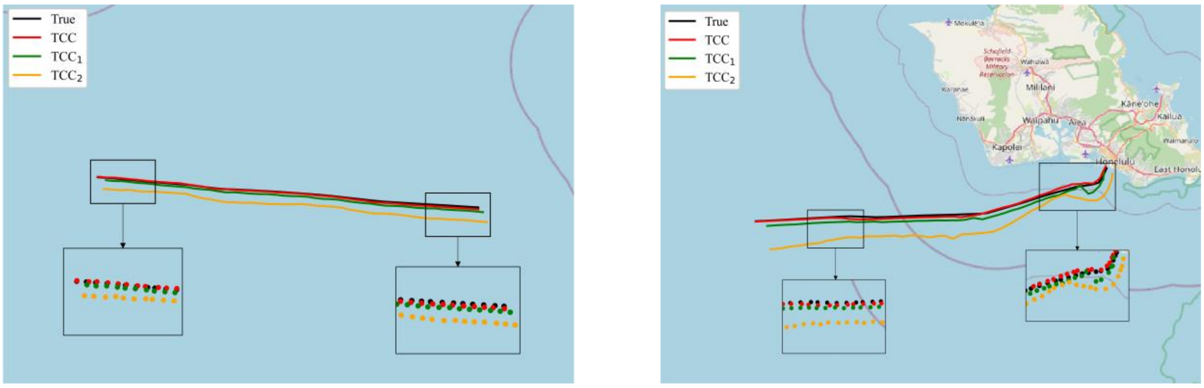
Based on the test results in *Tables 5–6*, the performance of the TCC model and TCC_1 and TCC_2 in trajectory prediction at different time steps can be observed. Comparing the prediction performance metrics of the three, the feature extraction module and the Trajectory Modelling Module significantly impact the prediction accuracy of the TCC model. In the TCC_2 model with the feature extraction module removed, the results of the prediction performance metrics decreased significantly especially at longer time steps. While TCN modelling does not show an advantage in the case of shorter time steps, the advantage has been shown as the time step increases, which indicates that for longer trajectory data, more information will be contained in it. The modelling effect of TCN can precisely capture the effective information in it, and learn the change trend in it, effectively improving the model performance. From the results of each index, under the prediction step of 6 min, the results of ADE, FDE and MAE are not as good as those of the TCC_1 model. It may be because the amount of information carried in the AIS data is less when the time step is shorter. The TCC model does not fully dig out the information about the vessel's state changes, which also indicates that in the case of shorter time steps. The simpler model may be better for prediction, but as the time step increases, the modelling effect of the TCN can precisely capture the effective information therein and learn the changing trend, thus effectively enhancing the model performance.

However, as the time step increases, the errors of each model show an increasing trend. The ADE, FDE and MAE (LON) index results of the TCC model are better than the other two ablation models in the test after 12 minutes, and the MAE (LAT) index is better than the other two ablation models in the test after 18 minutes. Its prediction performance is stronger than the other two ablation models. Therefore, it can be concluded that the Trajectory Modelling Module and the feature extraction module play a vital role in the TCC model. Differentiating and modelling all information not only reduces the complexity of the model, but also improves the generalisation performance of the real trajectory. The TCC model reduces the impact of other uncertain factors on the model trajectory prediction and improves the stability and robustness of the model, especially when the time step is long and the amount of information carried by AIS data is large.

The effect of other uncertainties on the model's trajectory prediction, the stability and robustness of the model are better, and for longer time predictions, the use of the TCC model is more effective than the ablated models TCC_1 and TCC_2.

Trajectory visualisation of ablation experiments

We use the TCC model, TCC_1 and TCC_2 models to select two vessel trajectories with different navigation states for real trajectory prediction. The vessel trajectory comparison results are shown in *Figures 6a–6b*. The vessel in *Figure 6a* is basically in a straight-line state, while the second vessel in *Figure 6b* has turning information. The prediction time step is 18 minutes, the trajectory coordinate points are 48, and the duration is 144 minutes. The MAE (LON) and MAE (LAT) results are calculated for each point in the two trajectories and plotted as line graphs in *Figures 7a–7d*. The black line segment "True" represents the real vessel trajectory, while the red, green and yellow lines are the predicted trajectories of TCC model, TCC_1 model and TCC_2 model respectively.



(a) *(b)*

Figure 6 – a) and b) respectively represent the prediction results of different models for vessel 1 and vessel 2 when the time step is 18 minutes

As can be seen in *Figures 6a–6b*, the TCC model performs best while the TCC_2 model performs worst. In order to highlight the comparison, we plot the scatter plots of the trajectories in *Figures 6a–6b*. *Figures 6a* shows the straight-line heading state, the TCC model predicts the results slightly more accurately than the TCC_1 model, the trajectory coordinate points of the TCC model almost coincide with the real trajectory, the trajectory points of the TCC1 model are slightly shifted, and the trajectory points of the TCC_2 are shifted the most. *Figure 6b* shows the heading state with a turn, and this visualisation of the trajectory scatter plot shows that the TCC model has a clear advantage, in the turning environment, the trajectory points predicted by the models of TCC_1 and TCC_2 are both shifted significantly, but the trajectory points predicted by the TCC model are still close to the real trajectory. We speculate that in the case of straight-line sailing, the AIS covariate information of the vessel changes less, and the advantage of the CNN feature extraction module is not obvious. In the case of turning, the AIS covariate information of the vessel changes a lot, e.g. the vessel needs to change the course direction and reduce the speed greatly when turning, and at this time, the CNN feature extraction module can capture the change of such sailing state, and at the same time, the TCN Trajectory Modelling Module can also capture the change of such sailing state by modelling the trajectory features. The TCN trajectory modelling module can also improve the accuracy of the model in predicting the trajectory by modelling the trajectory features. This further shows that the AIS covariate information of the vessel is very useful, and the effective use of the AIS covariate information can improve the robustness and accuracy of the model in predicting the future trajectory.

Therefore, combining the above results, the feature extraction module and the trajectory modelling module introduced in the prediction process can significantly enhance the accuracy of the model, and both the feature extraction module and the modelling module play a key role in significantly improving the accuracy of trajectory prediction. This enhances the stability and robustness of the model.

4. CONCLUSION AND FUTURE WORK

This study presents a novel vessel trajectory prediction model. The proposed model based on a network of TCN, CNN and ConvLSTM achieves advanced accuracy in vessel trajectory prediction. TCN plays an important role in vessel trajectory modelling by efficiently modelling the sequence of latitude and longitude coordinates through effective causal convolution, residual joining and other mechanisms to capture the long-term dependencies therein. The CNN operates by convolution on a local region to extract features and gradually expands the sensory field through pooling operations to obtain a more global representation. For vessel trajectory modelling, CNNs can understand the motion patterns and behaviours of vessels from local to global, and learn to adapt the feature representations to different data, which enables the CNNs to adapt to different types and sizes of vessel data, thus improving the generalisation ability and applicability of the model. ConvLSTM combines convolutional and cyclic operations and can simultaneously consider the temporal and spatial features of vessel trajectory data, thus better understanding temporal and spatial features of vessel trajectory data and spatial features at the same time, to better understand the motion behaviours of vessels at different times and positions and predict accurately the vessel trajectory. A series of experiments are conducted to verify the performance and robustness of the proposed vessel trajectory model. The experimental results show that the prediction accuracy of the proposed TCC model exceeds other neural network models. In addition, the TCC model demonstrates superiority and accuracy in predicting vessel trajectory under the condition of longer time steps. To improve the model performance, the following directions can be further extended. First, we can optimise the network layers in the TCC model by incorporating an attention mechanism to improve the accuracy of vessel trajectory prediction when the time step is shorter. Second, different levels of the prediction network layers may have an unexpected effect on the model prediction performance, thus we can improve the model performance by introducing additional network layers for predictions with longer time steps. In the future, we will explore integrating ensemble learning techniques to enhance the robustness and adaptability of TCC models. The weaknesses of individual models are mitigated by combining the predictions of multiple models to improve overall accuracy.

ACKNOWLEDGMENTS

The research was supported by the National Natural Science Foundation of China (Grant No. 62206116), the Key R&D Program of Hubei Provincial Department of Science and Technology (Grant No.ZZCXHZYF2022000815), National Natural Science Foundation of Hubei (2021CFB273) and Chunhui Plan Collaborative Research Project, Ministry of Education, China (HZKY20220350).

REFERENCES

[1] Peng Z, et al. Spatiotemporal distribution and evolution characteristics of water traffic accidents in Asia since the 21st century. *Journal of marine science and engineering*. 2023;11(11):2112. DOI: 10.3390/jmse11112112.

[2] Wang H, et al. An analysis of factors affecting the severity of marine accidents. *Reliability Engineering & System Safety*. 2021;210:107513. DOI: 10.1016/j.ress.2021.107513.

[3] Xiao F, et al. Comparison study on AIS data of ship traffic behavior. *Ocean Engineering*. 2015;95:84-93. DOI: 10.1016/j.oceaneng.2014.11.020.

[4] Harati-Mokhtari A, et al. Automatic identification system (AIS): data reliability and human error implications. *Journal of Navigation*. 2007;60(3):373-389. DOI: 10.1017/S0373463307004298.

[5] Sang L, et al. CPA calculation method based on AIS position prediction. *Journal of Navigation*. 2016;69(6):1409-1426. DOI: 10.1017/S0373463316000229.

[6] Kim KI, Jeong JS, Lee BG. Study on the analysis of near-miss ship collisions using logistic regression. *Journal of Advanced Computational Intelligence and Intelligent Informatics*. 2017;21(3):467-473. DOI: 10.20965/jaciii.2017.p0467.

[7] Murray B, Perera LP. A data-driven approach to vessel trajectory prediction for safe autonomous ship operations. In: *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*. Berlin, Germany. IEEE. 2018;240-247. DOI: 10.1109/ICDIM.2018.8847003.

[8] Sheng K, et al. Research on ship classification based on trajectory features. *Journal of Navigation*. 2018;71(1):100-116. DOI: 10.1017/S0373463317000546.

[9] Sutulo S, Moreira L, Soares CG. Mathematical models for ship path prediction in manoeuvring simulation systems. *Ocean Engineering*. 2002;29(1):1-19. DOI: 10.1016/S0029-8018(01)00023-3.

[10] Kawan, et al. Data-driven modeling of ship motion prediction based on support vector regression. In: *Proceedings of the 58th Conference on Simulation and Modelling (SIMS 58) Reykjavik, Iceland, September 25th – 27th, 2017*. Reykjavik, Iceland. 2017;350-354. DOI: 10.2316/P.2011.721-107.

[11] Virjonen, et al. Ship movement prediction using k-NN Method. In: *2018 Baltic Geodetic Congress (BGC Geomatics)*. Olsztyn, Poland. IEEE. 2018;304-309. DOI:10.1109/BGC-Geomatics.2018.00064.

[12] Volkova TA, Balykina YE, Bespalov A. Predicting ship trajectory based on neural networks using AIS data. *Journal of Marine Science and Engineering*. 2021;9(3):254. DOI:10.1109/BGC-Geomatics.2018.00064.

[13] Kim KI, Lee KM. Deep learning-based caution area traffic prediction with automatic identification system sensor data. *Sensors*. 2018;18(9):3172. DOI: 10.3390/s18093172.

[14] Liu RW, et al. An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system. *Ocean Engineering*. 2021;235:109435. DOI: 10.1016/j.oceaneng.2021.109435.

[15] Yang Sun, et al. An approach to ship behavior prediction based on AIS and RNN optimization model. *International Journal of Transportation Engineering and Technology*. 2020;6(1):16. DOI: 10.11648/j.ijtet.20200601.13.

[16] Zhong C, et al. Inland ship trajectory restoration by recurrent neural network. *Journal of Navigation*. 2019;72(6):1359-1377. DOI: 10.1017/S0373463319000316.

[17] Gao M, Shi G, Li S. Online prediction of ship behavior with automatic identification system sensor data using bidirectional long short-term memory recurrent neural network. *Sensors*. 2018;18(12):4211. DOI: 10.3390/s18124211.

[18] Wang X, Xiao Y. A deep learning model for ship trajectory prediction using automatic identification system (AIS) data. *Information*. 2023;14(4):212. DOI: 10.3390/info14040212.

[19] Zhang Yang, et al. Navigation trajectory prediction method of inland ships based on multi-model fusion. *China Mechanical Engineering*. 2022;33(10):1142. DOI: 10.3969/j.issn.1004-132X.2022.10.002.

[20] Ljunggren H. Using deep learning for classifying ship trajectories. In: *2018 21st International Conference on Information Fusion (FUSION)*. Cambridge, UK. IEEE. 2018;2158-2164. DOI: 10.23919/ICIF.2018.8455776.

[21] Zhou H, Chen Y, Zhang S. Ship trajectory prediction based on BP neural network. *Journal of Artificial Intelligence*. 2019;1(1):29. DOI: 10.32604/jai.2019.05939.

[22] Park J, Jeong J, Park Y. Ship trajectory prediction based on bi-LSTM using spectral-clustered AIS data. *Journal of Marine Science and Engineering*. 2021;9(9):1037. DOI: 10.3390/jmse9091037.

[23] You Lan, et al. Improved sequence-to-sequence model for short-term vessel trajectory prediction using AIS data streams. *Computer Science.* 2020;47(9):169-174. DOI: 10.11896/jsjkx.190800060.

[24] Patrikar J, et al. Predicting like a pilot: Dataset and method to predict socially-aware aircraft trajectories in non-towered terminal airspace. In: *2022 International Conference on Robotics and Automation (ICRA)*. Philadelphia (PA), USA. 2022;2525-2531. DOI: 10.1109/ICRA46639.2022.9811972.

[25] Bai S, Kolter JZ, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. Arxiv. 2018. [preprint] https://arxiv.org/pdf/1803.01271.

[26] Shi, et al. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Neural Information Processing Systems.* 2015. DOI: 10.48550/arXiv.1506.04214.

[27] Wu W, et al. Ship trajectory prediction: An integrated approach using ConvLSTM-based sequence-to-sequence model. *Journal of Marine Science and Engineering*. 2023;11(8):1484. DOI: 10.3390/jmse11081484.

[28] Tian Y, Li Z, Ai W, Han X. LSTM-RNN based black-box modeling of ship manoeuvring motion. *Ship Science and Technology*. 2024;46(11):80-84. DOI: 10.3404/j.issn.1672-7649.2024.11.015.

[29] Liu Z, et al. Hybrid deep learning models for ship trajectory prediction in complex scenarios based on AIS data. *Applied Ocean Research*. 2024. DOI: 10.1016/j.apor.2024.104231.

[30] Li T, Xu H, Zeng W. Ship classification method for massive AIS trajectories based on GNN. In: *2021 3rd International Conference on Artificial Intelligence and Computer Science (AICS) 2021 29-31 July 2021*. Beijing, China. IOP Publishing. 2021. DOI: 10.1088/1742-6596/2025/1/012024.

[31] Prasetya DA, et al. Resolving the shortest path problem using the haversine algorithm. *Journal of Critical Reviews*. 2020;7(1):62-64. DOI: 10.22159/jcr.07.01.11.

吴歆韵，陈家飞，熊才权，刘东华（通讯作者），万祥，陈泽熙

基于时序数据融合模型的船舶轨迹预测方法

摘要：

船舶轨迹预测在海上交通安全和应急管理中至关重要。利用船舶自动识别系统（AIS）数据进行的船舶轨迹预测引起了广泛关注。深度学习技术因其在细粒度特征学习和时间序列建模方面的优势，已广泛应用于船舶轨迹预测任务。然而，大多数基于深度学习的方法对 AIS 数据采用统一建模方法，忽视了 AIS 数据的多样性及环境因素对预测性能的噪声影响。为了解决这个问题，本研究提出了一种结合时序卷积网络（TCN）、卷积神经网络（CNN）和卷积长短期记忆网络（ConvLSTM）的方法来预测船舶轨迹，称为 TCC。该模型利用 TCN 捕捉时间序列的复杂相关性，利用 CNN 提取细粒度协变量特征，然后通过 ConvLSTM 捕捉轨迹序列的动态性和复杂性以预测船舶轨迹。在真实公共数据集上进行的实验结果表明，本文提出的 TCC 模型在船舶轨迹预测中表现出较高的准确性和鲁棒性，优于现有的基线算法。

关键词：

自动识别系统（AIS）数据；船舶轨迹预测；深度学习；神经网络