applied sciences

MDPI

Article

Predicting Vessel Trajectories Using ASTGCN with StemGNN-Derived Correlation Matrix

Ran Zhang [1], Xiaohui Chen [1,*], Lin Ye [1], Wentao Yu [1,2], Bing Zhang [1] and Junnan Liu [3]

[1] Institute of Data and Target Engineering, Information Engineering University, Zhengzhou 450001, China; zhr_ang@163.com (R.Z.); yleaflin@163.com (L.Y.); yuwentao@ict.ac.cn (W.Y.); 18337161789@163.com (B.Z.)
[2] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[3] Institute of Geo-Science and Technology, Zhengzhou University, Zhengzhou 450001, China; ljnzzu@zzu.edu.cn
* Correspondence: cxh_vrlab@163.com

Abstract: This study proposes a vessel position prediction method using attention spatiotemporal graph convolutional networks, which addresses the issue of low prediction accuracy due to less consideration of inter-feature dependencies in current vessel trajectory prediction methods. First, the method cleans the vessel trajectory data and uses the Time-ratio trajectory compression algorithm to compress the trajectory data, avoiding data redundancy and providing feature points for vessel trajectories. Second, the Spectral Temporal Graph Neural Network (StemGNN) extracts the correlation matrix that describes the relationship between multiple variables as a priori matrix input to the prediction model. Then the vessel trajectory prediction model is constructed, and the attention mechanism is added to the spatial and temporal dimensions of the trajectory data based on the spatio-temporal graph convolutional network at the same time as the above operations are performed on different time scales. Finally, the features extracted from different time scales are fused through the full connectivity layer to predict the future trajectories. Experimental results show that this method achieves higher accuracy and more stable prediction results in trajectory prediction. The attention-based spatio-temporal graph convolutional networks effectively capture the spatio-temporal correlations of the main features in vessel trajectories, and the spatio-temporal attention mechanism and graph convolution have certain interpretability for the prediction results.

Keywords: ais; trajectory prediction; attention mechanism; spatio-temporal graph convolution

Citation: Zhang, R.; Chen, X.; Ye, L.; Yu, W.; Zhang, B.; Liu, J. Predicting Vessel Trajectories Using ASTGCN with StemGNN-Derived Correlation Matrix. *Appl. Sci.* **2024**, *14*, 4104. https://doi.org/10.3390/app14104104

Academic Editor: José A. Orosa

Received: 7 March 2024
Revised: 30 April 2024
Accepted: 10 May 2024
Published: 12 May 2024

1. Introduction

Against the backdrop of increasingly frequent global vessel activities, vessel monitoring in coastal areas is particularly important for navigation safety, emergency response, and marine management. Traditional monitoring methods primarily rely on sensors to detect if vessels appear in key areas of interest. However, during certain unexpected events, such as signal interference, vessel data may not be obtained promptly, resulting in the inability to monitor vessels in real-time. This information lag becomes more prominent when sensors are turned off because the real-time vessel data cannot be acquired promptly, making it difficult to monitor them in real time.

Since the Automatic Identification System (AIS) can monitor real-time information such as position, sailing speed, and direction angle during ship navigation, the application value of AIS trajectory data in the fields of ship monitoring, navigation safety, emergency response, and marine management is becoming more and more prominent [1–4]. Through deep mining and analysis of AIS trajectory data, ship activity patterns can be revealed [5], and future ship trajectory trends can be further predicted [6]. The ultimate realization of this method provides important support for marine navigation safety decision-making, effectively improves the efficiency and pertinence of emergency response, and provides a more scientific and precise decision-making basis for marine management.

During the process of using AIS trajectory data for trajectory prediction, there are several key steps involved, including data preprocessing and trajectory prediction model construction [7,8]. The data preprocessing stage involves steps such as data parsing, cleaning and extraction, and trajectory compression, intending to select valid information and provide an accurate and complete data foundation for subsequent analysis. Subsequently, in the trajectory prediction step, deep learning techniques like recurrent neural networks (RNNs) play a crucial role [8]. This technique constructs ship trajectory prediction models to dig deeper into the patterns and relationships hidden in AIS trajectory data; when dealing with large amounts of data with multiple attributes and strong spatiotemporal correlations, it shows high efficiency and accuracy. Since trajectory data has multiple attributes, large data volume, and strong spatiotemporal correlation, vessel trajectory prediction models must have the ability to capture non-linear relationships between multiple variables.

In response to the above difficulties, this study proposes a trajectory prediction framework based on Attention Spatio-Temporal Graph Convolutional Network (ASTGCN). The primary objective is to accurately capture the intricate interrelationships among variables inherent in vessel trajectory data. Through comparative analysis with existing models, we demonstrate the significance of considering feature dependencies in improving trajectory prediction accuracy. Moreover, we validate the efficacy of ASTGCN with a correlation matrix derived from StemGNN in enhancing vessel trajectory prediction. The proposed method adopts a multi-step approach. First, the Time-Ratio (TR) trajectory compression algorithm is used to extract key feature points from vessel trajectories. Second, to capture the non-linear relationships between the data more accurately, a self-attention mechanism is used to construct the prior matrix input of the model, which learns the inherent connections and dependencies between the data. Simultaneously, attention mechanisms are applied in both time and space dimensions to assign higher weights to the main features in the data, helping the model to focus on the key information that has a greater impact on the prediction results, thus improving the accuracy of the prediction. Finally, by leveraging the powerful capabilities of spatiotemporal graph convolutional networks in handling complex patterns and temporal dependencies, a ship trajectory prediction model is constructed to achieve accurate ship trajectory prediction. This approach effectively utilizes the rich semantic information hidden in trajectory data and directly processes sequential data, such as trajectories, in the form of graphs, enabling precise prediction of ship navigation trajectories. This method not only improves prediction accuracy but also provides a powerful tool for understanding and interpreting ship navigation behaviors.

2. Related Work

2.1. Risk Collision Analysis

With the rapid development of the maritime industry, the safety of maritime transportation has attracted the attention of people in related fields, and the risk analysis methods of maritime transportation have been widely studied [9]. Risk collision analysis, as a broader scenario of trajectory prediction, plays an important role in improving the safety of maritime navigation. Ship collision risk analysis methods can be divided into probabilistic model-based methods, geometric-based methods, and artificial neural network-based methods [10].

Approach based on Geometry. Geometry-based methods can be divided into two categories: those based on the Closest Point of Approach (CPA) and those based on the Safe Domain (SD). The CPA method analyzes collision risk by predicting the closest possible point of encounter between ships in the future, considering parameters such as their current position, heading, and speed [11–13]. By calculating the shortest distance between ships and the time required to reach that distance, potential collision risks can be identified, allowing for the implementation of actions to avoid collisions. On the other hand, SD-based methods calculate potential collisions by establishing a navigational safety zone around a ship. However, these methods are highly sensitive to parameterization and may not yield highly accurate results.

Approach based on probabilistic model. The probabilistic model-based approach is primarily based on support vector machines (SVMs) and Bayesian networks. SVM-based methods calculate the probability of ship collisions by considering the ship's motion characteristics and environmental factors, enabling quantitative risk assessment. Zheng et al. [14] implemented SVM-based probability computation by using the Safe Domain (SD) of the owner ship (OS) and the target ship (TS) as input features. Building upon this foundation, Liu et al. [15] developed a fuzzy quaternion ship domain (FQSD) model for ships, which enables the calculation of ship collision risk by solving the maximum interval of SD (MISD) and the violation degree of SD (VDSD). The method based on Bayesian Networks (BNs) [16] is an effective means of modeling factors affecting accidents and nonlinear causal relationships. Montewka et al. [17] integrated ship collision simulation results and expert knowledge into Bayesian Networks to obtain the probability of specific collision occurrences. Jiang et al. [18] also proposed a ship collision risk analysis method based on the K2 algorithm for Bayesian networks (BNs), predicting probabilities of various types of maritime accidents along Maritime Silk Road through maritime accident reports.

Approach based on artificial neural network. The traditional approach [19] to analyzing collision risk using neural networks focuses on predicting ship positions to aid in collision avoidance. To enhance the prediction of risk index, Feng et al. [20] developed a convolutional neural network (CNN) for ship collision avoidance based on expert knowledge and Automatic Identification System (AIS) trajectory data. The network initially calculates the original ship collision risk based on the ships' motion characteristics and distance between them, then adjusts this value using expert experience. Finally, the corrected data and remotely sensed images are used as inputs for predicting the risk index. In order to reduce the learning time of the network, reinforcement learning is applied in collision risk analysis. By combining model-based and model-free algorithms with asynchronous advantage actor-critic (A3C) [21] algorithms with existing models, significant reductions in model learning time can be achieved. For example, Xie et al. [22] combined a long short-term memory neural network (LSTM), Q-learning, and the A3C algorithm to improve efficiency in the reinforcement learning process. To facilitate more effective learning of collision avoidance strategies, Zhang et al. [23] proposed Constrained-DQN (Deep Q Network). This approach reduces state-action space complexity by incorporating constraints based on International Collision Avoidance Rules at Sea (COLREG), thereby enhancing efficacy in collision avoidance outcomes.

2.2. Trajectory Prediction

Currently, trajectory prediction methods can be primarily divided into shallow learning-based and deep learning-based approaches [24]. Shallow learning-based methods have been around for a longer time and show certain effectiveness in handling simple trajectory prediction tasks. However, their application is limited due to the lack of evaluation standards and limited adaptability in complex scenarios. With the widespread application of deep learning techniques in various fields, more researchers are leveraging the advantages of deep learning in capturing long-term dependencies and complex patterns in data and applying them to the handling of trajectory prediction problems.

Trajectory prediction method based on shallow learning. Early trajectory prediction methods usually combine kinematic models and Bayesian filters or their extensions to make predictions by propagating the current state to the future state [25]. This method is simple and easy to implement, but the prediction accuracy is limited in complex scenarios.

To describe nonlinear motion, Pavlovic et al. [26] proposed a switched linear dynamical system model, and Sadeghian et al. [27] proposed a dynamic Bayesian network model that considers social and physical constraints for path prediction under specific constraints. These methods have certain advantages in dealing with complex problems, but they need to consume a lot of computing resources, and it is difficult to make full use of some additional scenarios and information. The rapid advancements in machine learning have facilitated the application of tracking algorithms to enhance trajectory prediction models, such as the Kalman Filter (KF), Markov Model (MM), and Gaussian Process (GP) [28–30]. KF has high short-term forecasting accuracy, but its long-term forecasting ability is limited. MM is sensitive to trajectory fluctuations and is not suitable for medium and long-term trajectory prediction. GP is well-suited for predicting noisy point trajectory data as it effectively mitigates the issue of insufficiently discrete trajectory data and accurately represents the statistical characteristics of the trajectory distribution. However, constructing a GP is quite complex and requires a long computation time.

Although the trajectory prediction method of shallow learning has achieved some results in the early stage, the model based on kinematics combined with a machine learning algorithm has certain limitations. The prior assumptions of these models may constrain overall performance and present challenges when dealing with complex scenarios and large data sets. Due to the lack of specific scene information, lack of motion feature information, complex model construction, and limited samples of large data sets, there is a certain gap between the prediction effect of these methods and the actual situation.

Trajectory prediction method based on deep learning. Compared with the shallow learning complex model construction, the trajectory prediction method based on deep learning does not need a fixed mathematical model. This method is based on the construction of a network and relies on large-scale data sets to learn a more reasonable mapping relationship to better deal with complex trajectory data. In recent years, with the rise of deep learning, a variety of temporal prediction models have emerged, among which Recurrent Neural Networks (RNN) and their variants, such as Long Short Term Memory (LSTM), have emerged. LSTM and Gate Recurrent Unit (GRU) have achieved remarkable success in trajectory prediction.

Antonios et al. [31] extended LSTM for human trajectory and solved the problem that the performance of the Seq2Seq sequence model decreases with the increase of input sequence; they also verified the effectiveness of Seq2Seq sequence model in trajectory modeling and motion pattern prediction. On this basis, different from single-trajectory prediction, STA-LSTM [32] and O-LSTM models take into account the interaction between research objects in a certain space-time region and the impact of environmental information to different degrees, and perform well on ETH and UCY data sets. Wang and Xiao [33] combined the characteristics of the two networks and proposed a CNN-LSTM-SE model for ship trajectory prediction, which performed well on several indexes.

However, when using a pure CNN model or a fusion of CNN and LSTM for temporal prediction of non-Euclidean spatial data such as vessel trajectories, the model is unable to adequately capture the dependencies between trajectory features and mine the temporal patterns of trajectories. Aiming at the characteristics of trajectory data, a graph-based neural network, which learns and trains on the graph structure, is capable of mining the potential feature relationships and temporal patterns of non-Euclidean data such as trajectories.

The Spatio-Temporal Graph Convolutional Networks (STGCN) proposed by Yu et al. [34] are suitable for non-linear and complex non-Euclidean spatial data, such as traffic flow, and can effectively capture spatio-temporal correlation. Guo et al. [35] proposed Attention Based Spatial-Temporal Graph Convolutional Networks (ASTGCN), which impose an attention mechanism in the temporal and spatial dimensions and achieve better results in capturing the spatio-temporal dependence in traffic flow data.

It is worth noting that no matter whether STGCN, ASTGCN, or some other graph-based time series prediction models, they all need the dependency between multiple variables as prior knowledge, and this prior input will greatly affect the subsequent prediction results. To solve this problem, Cao et al. [36] proposed a Spectral Temporal Graph Neural Network (StemGNN) for multivariate temporal prediction. The model learns the implicit correlation between variables through the structure of its latent correlation layer, and then inputs the learned adjacency matrix into the model for time domain, space domain to frequency domain transformation, and graph convolution operations, and finally returns to the original domain and outputs the prediction results. The model can capture the spatiotemporal dependence in multivariate time series without prior output. Therefore, this study takes the correlation matrix extracted from the potential correlation layer of StemGNN after training as the prior input of ASTGCN and then experiments with ship trajectory prediction.

3. Vessel Data Preprocessing

AIS data often comes with various issues during the collection process, such as missing values, noise points, and missing coordinate information, etc. To ensure complete and accurate trajectory data, it is essential to perform trajectory cleaning operations. On this basis, trajectory extraction is performed by statistically analyzing the time span between different trajectories associated with the same MMSI. Therefore, the specific tasks in the pre-processing stage of trajectory data are: removing invalid values, detecting and cleaning outliers, and extracting trajectories.

3.1. Data Preprocessing

3.1.1. Removing Invalid Values

The absence of latitude and longitude information in ship trajectory data can lead to track discontinuity and result in incorrect interpretations of ship behavior. Therefore, it is necessary to remove these data points when traversing the data to ensure the integrity and continuity of the trajectory data.

3.1.2. Outlier Detection and Cleaning

For erroneous data, which refers to the cases where the sog, cog, or rot of trajectory points exceed normal thresholds, this study employs a rule-based method for identification, followed by linear interpolation to replace the erroneous values. This approach not only preserves the key information of the original data and eliminates the interference of outliers but also considers the spatiotemporal correlation between trajectory points through interpolation, thereby enhancing the accuracy and reliability of the data.

3.1.3. Trajectory Extraction

Since the same MMSI may correspond to multiple trajectories in different periods, it is necessary to accurately extract each trajectory based on the time span. The specific steps are as follows: first, the AIS data are sorted according to timestamps to ensure the temporal order of the data; second, each MMSI is traversed, and the time span between adjacent data points is measured; finally, a suitable time threshold (such as several hours or days) is set. When the time span between a data point and its previous data point exceeds this threshold, it is considered the starting point of a new trajectory. Figure 1 shows the trajectory extraction process based on MMSI and timestamps. Through this processing, clear and accurate vessel trajectory information can be extracted from the original AIS data, providing strong support for subsequent navigation path analysis and prediction.

The essence of trajectory compression is to represent the raw trajectory data more simply while maintaining its key spatial and temporal relationships as well as trend features. In the application of AIS data, due to the large scale of AIS data, it is difficult to process and store the data, and the recorded data during navigation often contain a large amount of dense and redundant information. Therefore, to reduce data volume and improve data usability, it is necessary to compress trajectories.

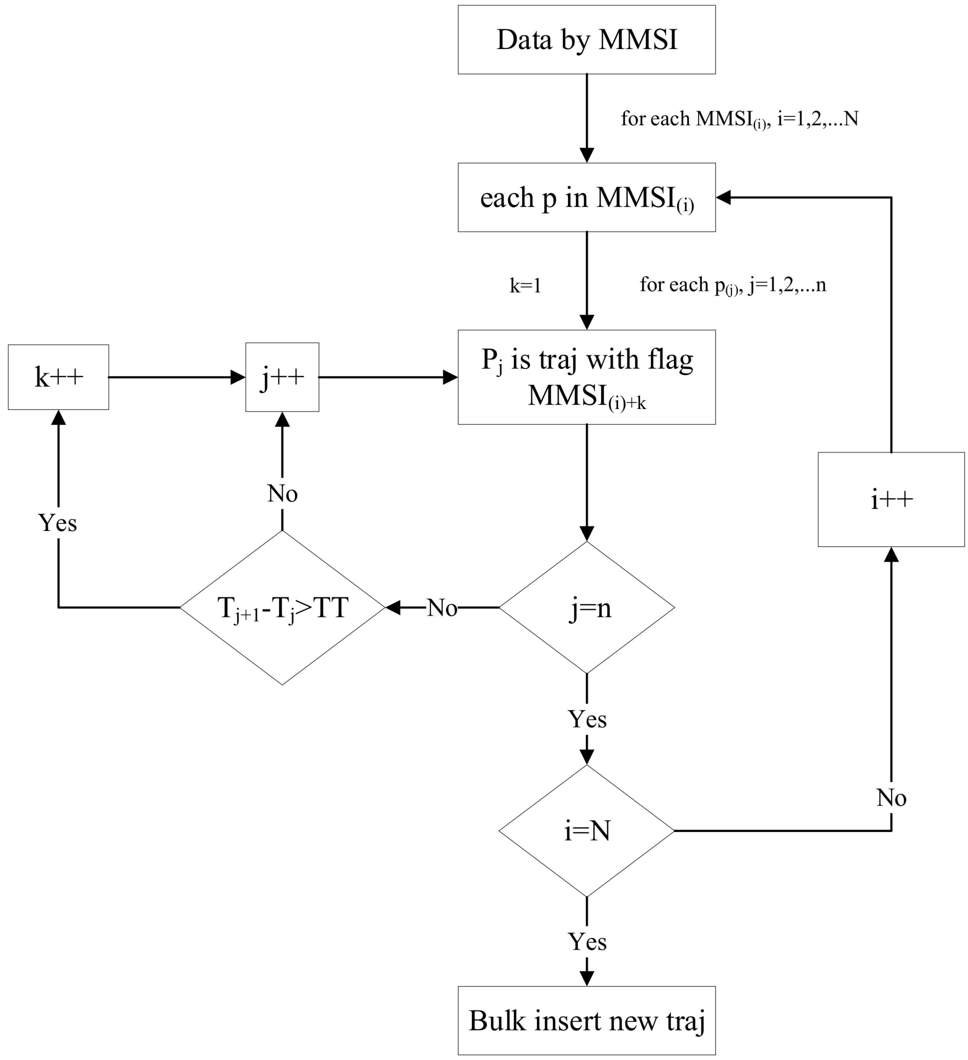


Figure 1. Pipeline for Trajectory Extraction.

3.2. Trajectory Compression

The essence of trajectory compression is to represent the raw trajectory data in a simpler way while maintaining its key spatial and temporal relationships as well as trend features. In AIS data applications, the difficulty of processing and storing arises from the large scale of AIS data and the presence of dense and redundant information recorded during vessel operations. Therefore, to reduce data volume and improve data usability, it is necessary to compress trajectories.

In this study, the TR algorithm is selected for trajectory compression, which is based on the synchronized Euclidean distance (SED) error for trajectory compression. The synchronized Euclidean distance error can measure the distance between two positions at the same time, which is the Euclidean distance between the point I on the trajectory segment and its time-synchronized point I' as shown in Formulas (1)–(3),

$$r = \frac{I_{time} - S_{time}}{E_{time} - S_{time}}, \tag{1}$$

$$I'_{lat} = S_{lat} + r(E_{lat} - S_{lat}), \tag{2}$$

$$I'_{lon} = S_{lon} + r(E_{lon} - S_{lon}), \tag{3}$$

where r represents the time ratio of $\overline{IS}$ and $\overline{ES}$, and S, E represents the start and end points of the trajectory segment.

4. Vessel Position Spatio-Temporal Prediction Model

4.1. Description of the Problem

AIS-based trajectory prediction is a multivariate time series forecasting problem [37] that aims to predict the future movement of vessels using historical trajectory data. This involves using pre-processed historical trajectory data features as inputs to the model and inferring the position and timing of future vessels from these features. Specifically, this can be described as follows:

Given the trajectory data of T position points in the past: $X = (x_{t-T+1}, \ldots, x_t)$, where $x_t \in R^n$ represents the eigenvalue $x_t = \left(vessel_{type}, draught_t, lon_t, lat_t, sog_t, cog_t, rot_t, navstatus_t, time\right)$ of the position point at time t. Input X into the prediction model to obtain the predicted trajectory $Y = \left(y_{t+1}, \ldots, y_{t+p}\right)$ of position points in the future, where $y_{t+1} \in R^n$ represents the predicted trajectory point information at time $t+1$.

ASTGCN, as a deep learning model specially designed for processing spatiotemporal data, has unique advantages in directly processing time series data, such as trajectory and traffic flow. The core of the model consists of three parts: space-time attention mechanism, space-time graph convolution, and prior matrix determination.

① The spatio-temporal attention mechanism focuses on learning dynamic spatiotemporal dependencies in trajectory data, where the spatial attention mechanism is used to simulate complex dynamic associations between different features to better understand and capture the internal patterns of the data, while the temporal attention mechanism is used to capture dynamic temporal associations between different time points, which enables the model to better understand the temporal evolution of data. ② Space-time graph convolution is a kind of convolution operation based on graph structure, which includes graph convolution and time dimension convolution. Graph convolution extracts the correlation of feature nodes from the graph-based trajectory network structure to help the model understand the network structure characteristics of the data. Convolution in the time dimension describes the dependencies between adjacent time segments, allowing the model to better capture the temporal dynamics of the data. ③ The prior matrix is determined to deal with the irregular shape of the graph, and the structure of the graph is constrained by introducing prior knowledge. Since spatiotemporal data usually have irregular forms, a priori matrices need to be introduced to deal with this irregularity to enable the model to better adapt to graph structures of different forms, thereby enhancing the model's generalization ability. The overall ASTGCN network framework is shown in Figure 2.

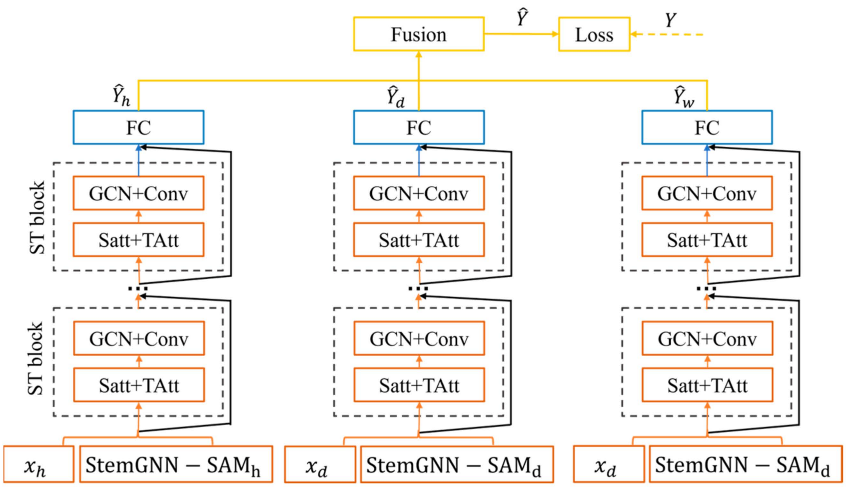


Figure 2. ASTGCN network structure. The self-attention matrix output by the StemGNN model ($\mathrm{StemGNN-SAM}_i$) is used as the prior matrix input for ASTGCN.

4.2. Prediction Model

4.2.1. The Spatiotemporal Attention Mechanism

In the module of spatio-temporal attention mechanism, the additive attention mechanism is used to process the spatial and temporal dimensions of the input data. Through training, the model can learn and acquire the corresponding attention weights, and then capture the relationship between the input data in different times and spaces. This mechanism not only enhances the model's understanding of spatiotemporal data but also provides more abundant spatiotemporal information for improving the model's prediction performance. The spatial attention mechanism is formulated as follows:

$$S = V_s \cdot \sigma\left(\left(X_h^{(r-1)} W_1\right) W_2 \left(W_3 X_h^{(r-1)}\right)^T + b_s\right), \tag{4}$$

$$S'_{i,j} = \frac{\exp(S_{i,j})}{\sum\limits_{i=1}^{N} \exp(S_{i,j})}. \tag{5}$$

$X_h^{(r-1)} = \left(X_1, X_2, ...X_{T_{r-1}}\right) \in R^{N \times C_{r-1} \times T_{r-1}}$ is the input of the r-th STblock, C_{r-1} represents the number of channels for the r-th layer input data, and T_{r-1} is the length of the time dimension of the r-th layer input data. $V_s, b_s \in R^{N \times N}, W_1 \in R^{T_{r-1}}$, $W_2 \in R^{C_{r-1} \times T_{r-1}}, W_3 \in R^{C_{r-1}}$ are the matrices involved in training. σ indicates the sigmoid activation function. The attention matrix S is computed dynamically from the input of this layer, the element $S_{i,j}$ in S' represents the degree of correlation of nodes semantically, and the softmax Equation (5) is used for weight normalization. When performing graph convolution, the spatial attention matrix S is multiplied by the adjacency matrix W to calculate the dynamic influence between nodes.

The time attention mechanism is calculated as follows:

$$E = V_e \cdot \sigma(((X_h^{(r-1)})^T U_1) U_2 (U_3 X_h^{(r-1)}) + b_e), \tag{6}$$

$$E'_{i,j} = \frac{exp(E_{i,j})}{\sum\limits_{j=1}^{T_{r-1}} exp(E_{i,j})}, \tag{7}$$

$V_e, b_e \in R^{T_{r-1} \times T_{r-1}}, U_1 \in R^N, U_2 \in R^{C_{r-1} \times N}, W_3 \in R^{C_{r-1}}$ are the matrices involved in training.The time-dimensional correlation matrix E is determined by the variable input, and the element $E_{i,j}$ in E semantically represents the degree of correlation of nodes i, j. Finally, weight normalization is performed for E through Formula (7). The normalized time-attention matrix is directly applied to the input to obtain $\widehat{X_h^{r-1}} = \left(\widehat{X_1}, \widehat{X_2}, \ldots, \widehat{X_{r-1}}\right) = (X_1, X_2, \ldots, X_{T_r-1})E' \in R^{N \times C_{r-1} \times T_{r-1}}$, thereby integrating relevant information and dynamically adjusting the input.

The spatio-temporal attention mechanism is used to enhance the learning ability of the model. Specifically, an additive attention mechanism is added to the temporal and spatial dimensions of the input of each STblock module, which is used to calculate the attention weights between different temporal and spatial nodes. The combination of spatio-temporal attention mechanism and graph convolutional network will enhance the modeling ability of the trajectory prediction model, expand the range of spatio-temporal feature expression, and improve the accuracy and robustness of trajectory prediction.

4.2.2. The Spatio-Temporal Graph Convolution

The spatio-temporal graph convolutions include convolutions in the spatial dimension and the temporal dimension. The former captures the spatial dependence from the neighborhood, and the latter mines the temporal dependence of the neighboring time. The dependence between trajectory features is regarded as a graph structure, and the value of each node is regarded as a signal on the graph. To fully exploit the topological properties of the network, the graph convolution based on the spectrogram conclusion is used to directly process the signal, and the signal correlation of the network is used in the spatial dimension.

In spectrogram analysis, the structural properties of a graph can be obtained by analyzing the corresponding Laplacian matrix of the graph and its characteristic values. The Laplacian matrix of the graph is defined as $L = D - A$, and its normalized form is $L = I_N - D^{-\frac{1}{2}}AD^{-\frac{1}{2}} \in R^{N\times N}$; A represents the graph adjacency matrix, I_N is the identity matrix, and the degree matrix D is a diagonal matrix consisting of node degree values $D_{ii} = \sum\limits_j A_{ij}$. Eigenvalue decomposition of the Laplacian matrix:

$$L = U\Lambda U^T. \tag{8}$$

$\Lambda = diag([\lambda_0, ..., \lambda_{N-1}]) \in R^{N\times N}$ is a diagonal matrix of eigenvalues and U represents an orthogonal matrix composed of eigenvectors. Compared to traditional CNNs that only work on regular data in Euclidean space, the following graph convolution can better capture the interaction and information exchange between nodes in the corresponding graph structure of non-Euclidean data. This is achieved by using a diagonalizable linear operator in the Fourier domain to replace the traditional convolution. The specific convolution formula is as follows:

$$g_\theta \ *_G x = g_\theta(L)x = g_\theta(U\Lambda U^T)x = Ug_\theta(\Lambda)U^Tx. \tag{9}$$

$*_G$ here denotes the graph convolution operation. By using Chebyshev polynomials, it is possible to maintain computational accuracy and significantly improve the efficiency of processing large-scale graph data without performing costly feature decomposition:

$$g_\theta \ *_G x = g_\theta(L)x = \textstyle\sum_{k=0}^{K-1} \theta_k T_k(\widetilde{L})x. \tag{10}$$

To dynamically adjust the correlation between points, each term of the Chebyshev polynomial and the spatial attention matrix S' are performed by the Hamada product, where the parameter $\theta \in R^K$ represents the vector of polynomial coefficients and $\widetilde{L} = \frac{2}{\lambda_{mx}}L - I_N$, λ_{inax} is the largest eigenvalue of the Laplacian matrix. The recursion of the Chebyshev polynomial is defined as $T_{\dot{\kappa}}(x) = 2xT_{\dot{\kappa}-1}(x) - T_{\dot{\kappa}-2}(x)$, $T_0(x) = 1$, $T_1(x) = x$. To dynamically adjust the correlation between points, each term of the Chebyshev polynomial and the spatial attention matrix S' are performed by the Hamada product (denoted by $\odot$). The graph convolution is transformed as follows:

$$g_\theta \ *_G x = g_\theta(L)x = \sum_{k=0}^{K-1} \theta_k\big(T_k(L) \odot S'\big)x. \tag{11}$$

In the spatial dimension, the graph convolution operation has captured the neighbor information on each node, and then a standard convolutional layer is superimposed on the temporal dimension for calculation, and the node information is updated by merging the information of adjacent time segments:

$$X_h^{(r)} = ReLU\Big(\phi * \big(ReLU(g_\theta *_G \hat{X}_h^{(r-1)})\big)\Big)R^{C_r\times N\times T_r}, \tag{12}$$

where $*$ represents the standard convolution operation, ϕ is the temporal dimension convolution kernel parameter, and $ReLU$ is the activation function.

In short, the spatio-temporal convolution module can well capture the dependencies in terms of spatio-temporal characteristics of trajectory data. STblock is the core component of the spatio-temporal convolution module, which is composed of a spatio-temporal attention module, spatio-temporal convolution module, and residual module. When stacked, multiple STblocks can further extract a wider range of dynamic spatio-temporal correlations. Finally, a fully connected layer is introduced to map the spatio-temporal features extracted by the STblock module to the dimensional space of the prediction target in order to realize the effective prediction of trajectory data.

4.2.3. Determining a Priori Matrix

When the ASTGCN model processes trajectory data, it needs to define the prior matrix of node information, so that based on the spatio-temporal attention mechanism, the temporal and spatial features of trajectory data can be effectively captured through the graph convolutional layer, and the dependencies between feature nodes can be further established. For the specific task of AIS trajectory prediction, the prior matrix can be regarded as the correlation coefficient matrix between trajectory features, and the self-attention matrix output by the StemGNN model can be used as the prior input of ASTGCN. When dealing with multivariate time series prediction problems such as trajectory prediction, the StemGNN model learns the hidden association between variables through the Latent Correlation Layer as an adjacency matrix, which is passed into the two-layer StemGNN block. This model is universal to all multi-dimensional time series without predefined topology structures, and the output self-attention matrix can be used as the prior input of other graph-based time series prediction models. Therefore, this paper uses StemGNN as one of the comparison models. The Latent correlation layer correlation matrix of the StemGNN after training is used as the prior input of ASTGCN, and the experiment of vessel trajectory prediction is then carried out.

5. Experiment and Analysis

5.1. Experimental Data

AIS data from January to February 2017 are used in the study. The data volume of this part is 10.9 G and contains daily AIS vessel trajectory information during this period. The information is stored in the format of CSV files. Each CSV file contains millions of data records, and each record shows the static and dynamic information of the vessel in detail.

Among them, the static information includes MMSI number, IMO number, vessel name, type, length and width, position, etc. The dynamic information includes vessel position, time, ground heading, ground speed, bow direction, turning speed, sailing state, etc. The detailed information is shown in Tables 1 and 2.

Table 1. Static information of the vessel.

Attribute Name	Attribute Meaning	Attribute Type
MMSI	Vessel Unique identification code	Int
Name	Vessel Name	Char
Vessel Type	Vessel Type	Char
Length	Vessel Length	Int
Width	Vessel Width	Int

Table 2. Dynamic information of the vessel.

Attribute Name	Attribute Meaning	Attribute Type
Lon	Longitude	Float
Lat	Latitude	Float
Sog	Ground speed	Float
Cog	Course over ground	Float
Rot	Rate of turn	Float
DT_Pos_Utc	AIS Dynamic Position Update Time	Time
Draught	Vessel Draught	Float
Nav_status	Vessel Status	Char

5.2. Experimental Setup

The time span of the data set is from 1 January 2017 to 5 January 2017. The preprocessed trajectory data are first grouped according to MMSI, and then the sliding window method is used to process each trajectory segment to generate the input data required by the model.

The features in the training data include $(vessel_{type}, draught_t, lon_t, lat_t, sog_t, cog_t, rot_t, navstatus_t, time)$ and the output of the model predicts the location and time of future trajectory points, which is $(lon_{t+1}, lat_{t+1}, time)$. To assess the model's performance, we partitioned the dataset into three subsets, train, validation, and test, in a 7:2:1 ratio. In the training process, a specific sequence step size is set, that is, the training sequence step size (window) is 15, and the prediction sequence step size (horizon) is 5. Optim is set as Adam optimizer. To prevent the model from overfitting or gradient explosion during the training process, the gradient clipping technique is adopted, and the learning rate decay is used to gradually approach the optimal solution.

STGCN, CNN_LSTM_CBAM, TCN, StemGNN, and other models are selected as baseline methods to conduct comparative experiments with ASTGCN. Among them, there are some structural similarities between STGCN and ASTGCN. Both adopt Gated CNNs to extract features in both spatial and temporal dimensions and both perform graph convolution operations in both dimensions separately. This design enables the model to deeply capture complex dependencies in spatio-temporal data. CNN_LSTM_CBAM is a model that combines a CNN and a LSTM network. It realizes the fusion and screening of features by introducing the convolutional module attention mechanism. This design not only enhances the feature extraction ability of the model but also provides it with excellent model recognition ability. TCN adopts a pure convolutional approach that combines causal convolutions, dilated convolutions, as well as the design of residual networks. This structure enables the model to effectively capture temporal patterns, enhance the memory of long-term dependencies, and compute features at multiple locations in parallel, thereby improving the training speed. StemGNN is a relatively unique model. It maps the data to the spectral domain by transforming it to the temporal and spatial domain and performs the corresponding convolution operation on this basis. StemGNN can extract temporal patterns and combine the self-attention mechanism to capture the dependency information between features, thus providing the adaptive ability for time series prediction tasks. Through the comparative experiments of these baseline methods, we can comprehensively evaluate the performance and advantages of ASTGCN in dealing with spatio-temporal data.

When evaluating the performance of trajectory prediction models, we usually use MSE (Mean Square Error) and L1 Loss as two evaluation metrics. MSE is a crucial metric for evaluating the predictive performance of a regression model. It measures the accuracy of the model's predictions by calculating the mean of the sum of the squares of the differences between the predicted and actual values. Despite its sensitivity to outliers, MSE remains widely used in trajectory prediction because it captures the continuity and smoothness of the model's predictions. The formula is as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_l)^2. \tag{13}$$

L1 Loss, also known as the mean absolute error (MAE), is another commonly used metric for evaluating the accuracy of regression models. It is computed by taking the absolute difference between the predicted values and the true values for each sample, and then averaging them. Unlike mean squared error (MSE), L1 Loss is more robust to outliers. The formula is as follows:

$$\mathrm{L1Loss} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_l|. \tag{14}$$

where n represents the number of trajectory points, y_i represents the true value of the i-th sample, and y_i represents the predicted value of the i-th sample.

In the trajectory prediction model, MSE and L1 Loss are used to measure the prediction ability of the model, where MSE is suitable for smoother prediction results, and L1 Loss is suitable for more attention to outliers or more robust prediction results.

5.3. Experimental Results Analysis

We deeply discussed the prediction accuracy of the five models on train/val/test track data and the results shown in Figures 3–5. According to the training and testing results of each model, CNN_LSTM_CBAM has relatively weak performance, high error, and the slowest convergence speed. The other four models are similar in performance, but ASTGCN has the smallest error and the fastest convergence on the training data set. StemGNN and STGCN followed closely behind, while TCN performed slightly worse than STGCN. ASTGCN showed the best performance on the validation dataset (val) and test dataset (test). The error is quite close to the error on the training data set, and the results are stable. StemGNN and STGCN followed closely behind, while TCN showed clear signs of volatility and elevated errors. CNN_LSTM_CBAM continues to underperform on all datasets.

In conclusion, ASTGCN shows excellent performance and stability in this experiment. Compared with other models, it shows lower error and faster convergence rates on training and validation data sets. This result shows that ASTGCN has strong potential and capability in processing spatiotemporal trajectory data.

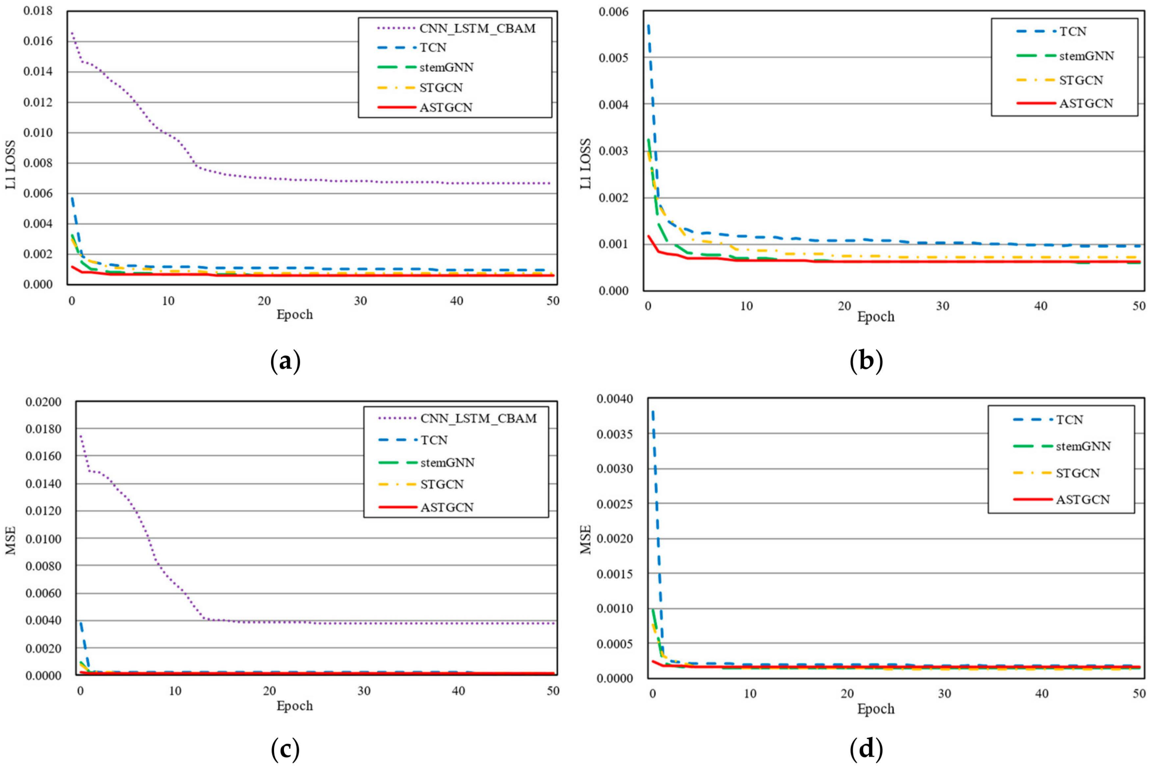


Figure 3. Error situation of each model on the training dataset: (**a**) Comparison chart of L1 indicators; (**b**) Comparison chart of L1 indicator details; (**c**) Comparison chart of MSE indicators; (**d**) Comparison chart of MSE indicator details.

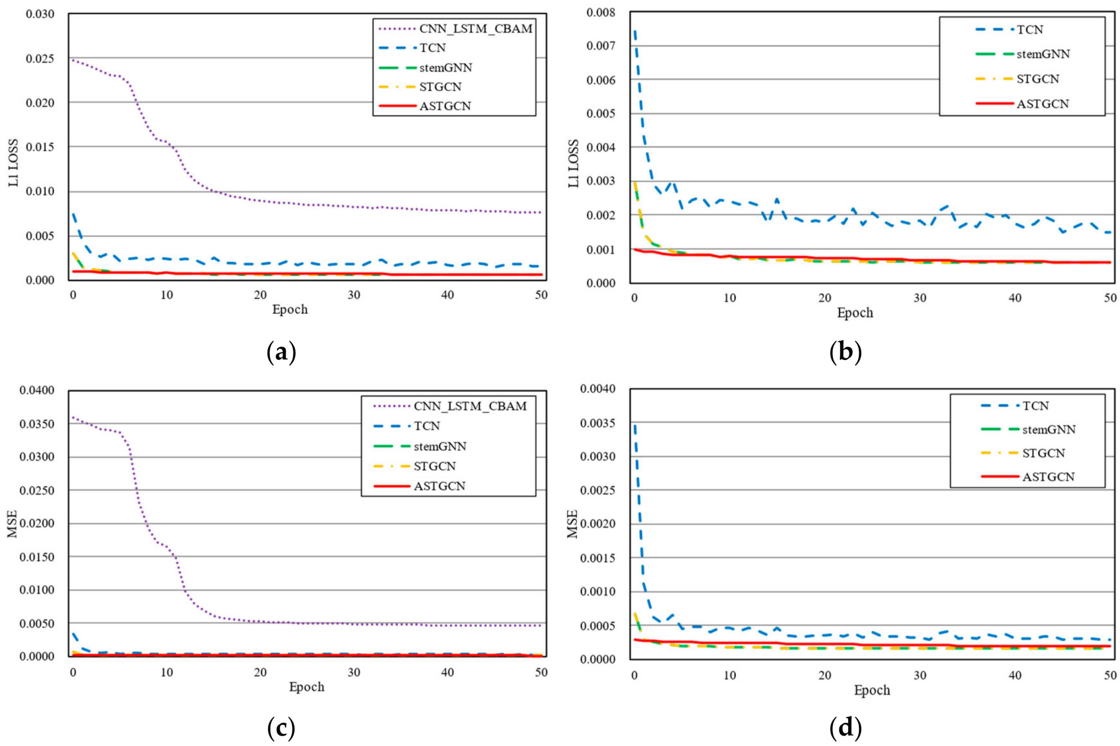


Figure 4. Error situation of each model on the val dataset: (**a**) Comparison chart of L1 indicators; (**b**) Comparison chart of L1 indicator details; (**c**) Comparison chart of MSE indicators; (**d**) Comparison chart of MSE indicator details.

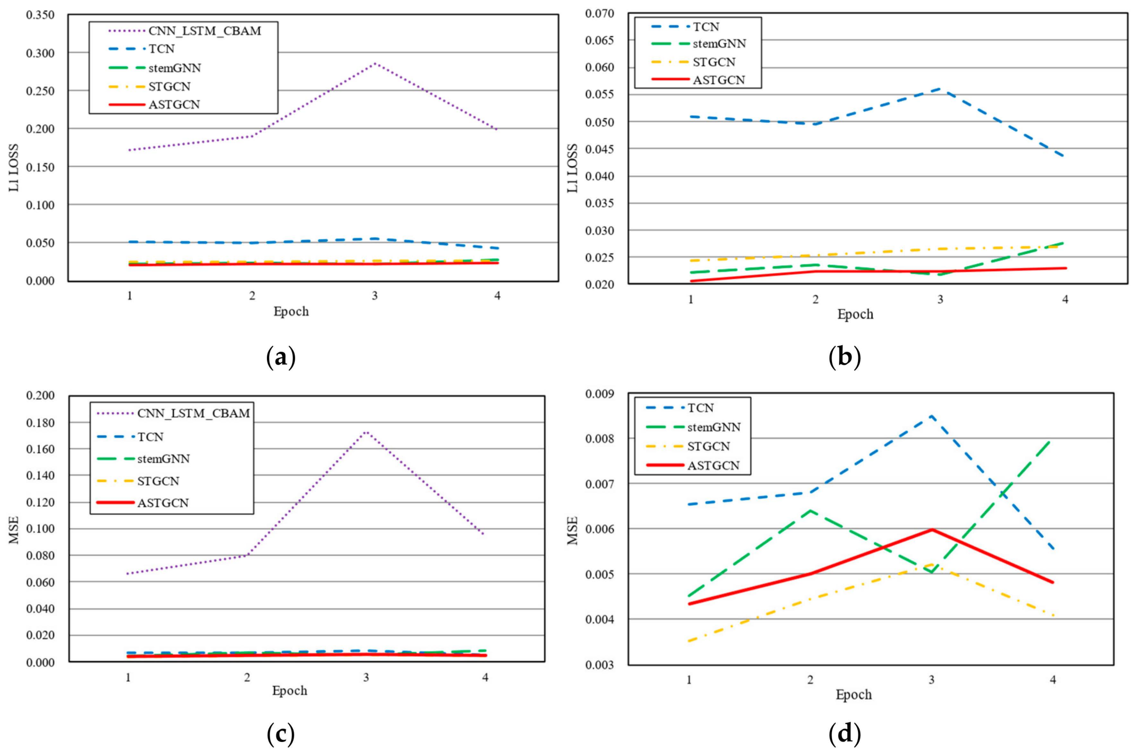


Figure 5. Error situation of each model on the test dataset: (**a**) Comparison chart of L1 indicators; (**b**) Comparison chart of L1 indicator details; (**c**) Comparison chart of MSE indicators; (**d**) Comparison chart of MSE indicator details.

By calculating the accuracy of each trajectory prediction model, the performance of each model in predicting trajectory data is deeply discussed. The statistical results are shown in Table 3. It can be seen from the data in the table that the prediction accuracy of the ASTGCN model is superior to other models on each data set. Among them, it can be seen from the prediction of CNN_LSTM_CBAM and TCN that although the relevant models introduce standard convolution technology, which is helpful for trajectory prediction, they only consider the local information of each position of the input data, ignoring the important information of the time dimension. This makes these models limited in making full use of the timing feature information in vessel trajectories. In contrast, graph-based trajectory models such as StemGNN, STGCN, and ASTGCN are more suitable for dealing with vessel trajectories with complex nonlinear relationships. These models can capture the dependencies between features and the pattern information in time series to provide more reliable and stable vessel trajectory prediction results. In summary, the accuracy of vessel trajectory prediction of the ASTGCN model is significantly better than that of other models on each data set, especially on training and verification data sets, and the error is stable and kept at a low level. This shows that the ASTGCN model has high trajectory prediction accuracy and good generalization ability and can effectively adapt to various vessel trajectory data.

Table 3. Trajectory prediction accuracy of each model.

Model	Train		Val		Test	
	MAE	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**
CNN_LSTM_CBAM	0.00660	0.00370	0.00760	0.004600	0.1980	0.0950
TCN	0.00094	0.00017	0.00150	0.000287	0.0433	0.0056
StemGNN	0.00060	0.00014	0.00060	0.000190	0.0230	0.0048
STGCN	0.00070	0.00013	0.00078	0.000160	0.0270	0.0043
ASTGCN	0.00060	0.00016	0.00060	0.000160	0.0227	0.0041

By calculating the accuracy of each trajectory prediction model, the performance of each model in predicting trajectory data is presented. The statistical results are shown in Table 3. It can be seen from the data in the table that the prediction accuracy of the ASTGCN model is superior to other models on each data set. Among them, it can be seen from the prediction of CNN_LSTM_CBAM and TCN that although the relevant models introduce standard convolution technology, which is helpful for trajectory prediction, they only consider the local information of each position of the input data, ignoring the important information of the time dimension. This makes these models limited in making full use of the timing feature information in vessel trajectories. In contrast, graph-based trajectory models such as StemGNN, STGCN, and ASTGCN are more suitable for dealing with vessel trajectories with complex nonlinear relationships. These models can capture the dependencies between features and the pattern information in time series to provide more reliable and stable trajectory prediction results.

To verify the robustness and generalizability of the model on different datasets, the trajectory data from April 2023 were used in this study to validate the model, and the experimental results are shown in Table 4. From the data in the table, it can be seen that the prediction accuracy of ASTGCN is higher than that of all other models. Combined with Table 3, it can be seen that when the collective data volume is small, the prediction accuracy of convolution-based models such as CNN_LSTM_CBAM and TCN is lower, while graph-based prediction models can still produce more accurate predictions on smaller volume datasets due to their ability to capture the dependencies between features.

Table 4. Trajectory prediction accuracy of each model on April 2023 trajectory data.

Model	Train		Val		Test	
	MAE	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**
CNN_LSTM_CBAM	0.00644	0.00667	0.01408	0.0150	0.15941	0.06184
TCN	0.00045	0.00011	0.0032	0.00125	0.07312	0.00847
StemGNN	0.00028	0.00007	0.00048	0.00019	0.01446	0.00487
STGCN	0.00036	0.00011	0.00042	0.00031	0.06107	0.00481
ASTGCN	0.00018	0.00008	0.00019	0.00019	0.01442	0.00289

In summary, the accuracy of vessel trajectory prediction of the ASTGCN model is significantly better than that of other models on each data set, especially on training and verification data sets, and the error is stable and kept at a low level. This shows that the ASTGCN model has high trajectory prediction accuracy and good generalization ability, and that it can effectively adapt to various vessel trajectory data.

Additionally, ASTGCN possesses a certain level of interpretability when performing trajectory prediction. During the training process, after the training of the StemGNN model is completed, the attention matrix in the Latent correlation layer is displayed in the form of a heat map, and the dependence between feature nodes can be visually observed, as shown in Figure 6. In training the ASTGCN network, we use StemGNN's attention matrix as a prior input. After completing the training, the attention matrix was visualized in both spatio-temporal and temporal dimensions, as shown in Figure 7. The visualization reveals significant interactions between different time steps in the temporal dimension. In the spatial dimension, the ship's motion speed (SOG) and steering angular velocity (ROT) have a greater effect on the variables lon, lat, and time, while the interactions among the other variables are weaker. The visual result of this dependence relationship is in accord with the actual motion principle of the vessel.

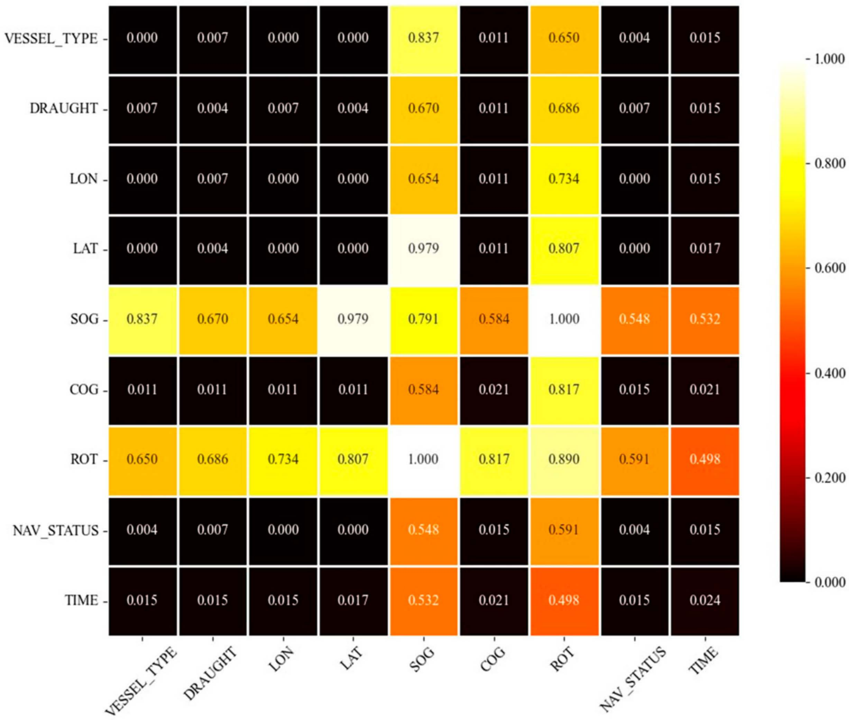
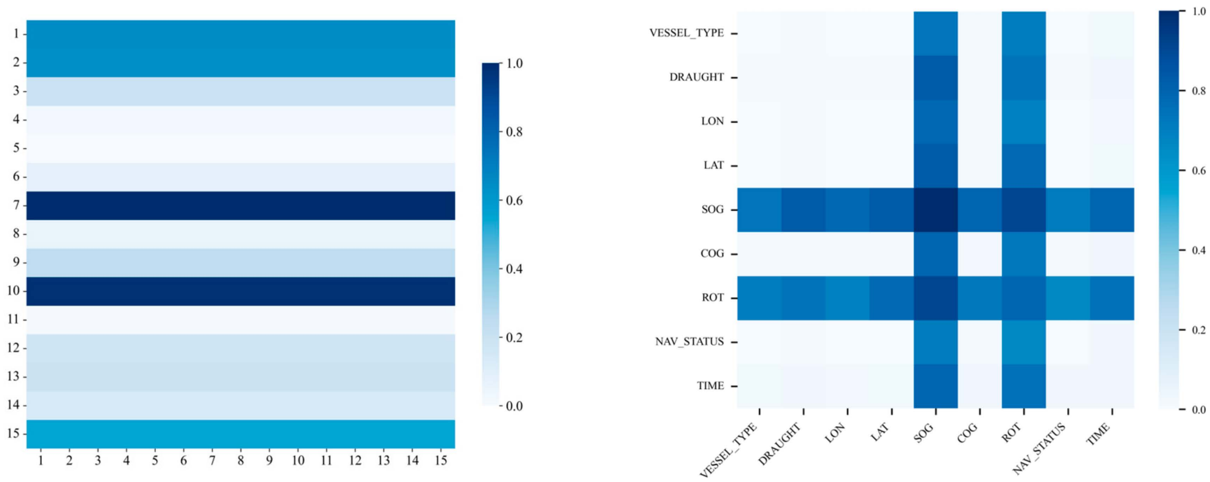


Figure 6. StemGNN attention matrix heat map.



(**a**) (**b**)

Figure 7. Heatmaps of the attention matrices in the temporal and spatial dimensions during the trajectory prediction process of ASTGCN. (**a**) Heatmap of the attention matrix in the temporal dimension; (**b**) Heatmap of the attention matrix in the spatial dimension.

The computational complexity of the ASTGCN model is mainly affected by the graph convolution layer and the attention mechanism. The computational complexity of the graph convolution operation mainly depends on the number of nodes and the structure of the adjacency matrix, while the adaptive adjacency matrix is used in ASTGCN, the model can dynamically adjust the number of adjacency matrices through the learning process, which reduces the number of nodes that need to be processed in the traditional graph convolution network and further reduces the computational complexity. In addition, although the spatio-temporal attention mechanism of the model increases the computational complexity compared to other models, this allows the model to better capture the correlation between ship trajectory data, which improves the prediction accuracy and generalization ability.

In summary, the ASTGCN model not only performs well in prediction accuracy, but its interpretability also provides us with the ability to deeply observe the dependencies between feature nodes and the patterns in the space-time dimension. This interpretability not only enhances the reliability of model prediction but also enables us to accurately understand the internal structure of vessel trajectory data in order to provide a more reliable and accurate basis for offshore vessel management decisions.

To verify the application effect of the model in real scenarios, we developed a trajectory visual analysis platform, as shown in Figure 8, and integrated this trajectory prediction framework. In this platform, users can realize trajectory prediction for any number of steps by setting the number of prediction steps. Figure 9 shows the visualization of the prediction results of the model on straight and curved trajectory.

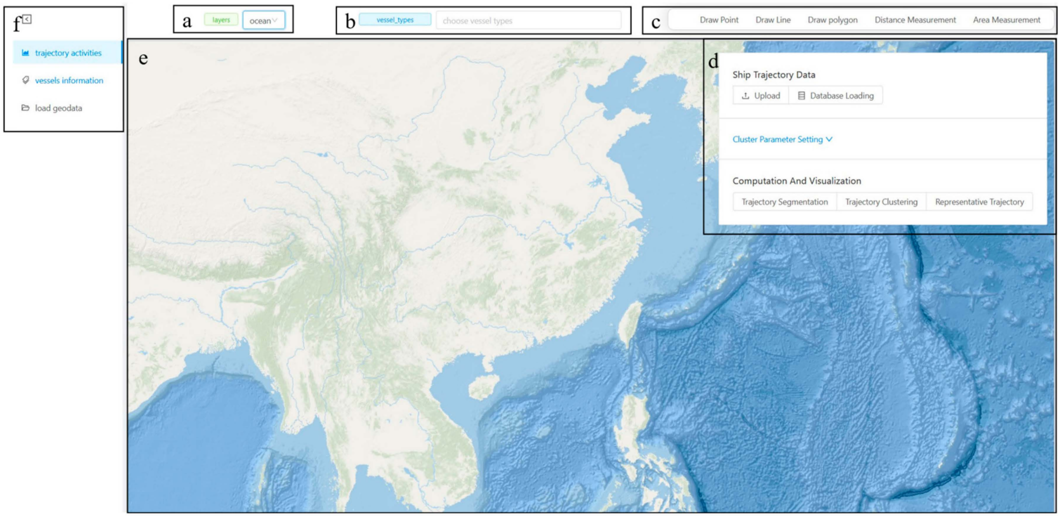


Figure 8. Visual analytics platform interface. (**a**) Selection box of the underlying map; (**b**) Ship type filter; (**c**) Operation box of GIS (measurement, screening, marking); (**d**) Multi-functional menu bar, the functions of ship trajectory data include ship trajectory data loading items, screening of the time range of ship activities and screening of ship types; the functions of calculation and visualization include the algorithm selection of trajectory segmentation, trajectory clustering and representative trajectory display; (**e**) Corresponding to the visualization results of trajectory data loaded according to the filtering conditions as well as the prediction results; (**f**) Interface navigation bar for switching the interface.

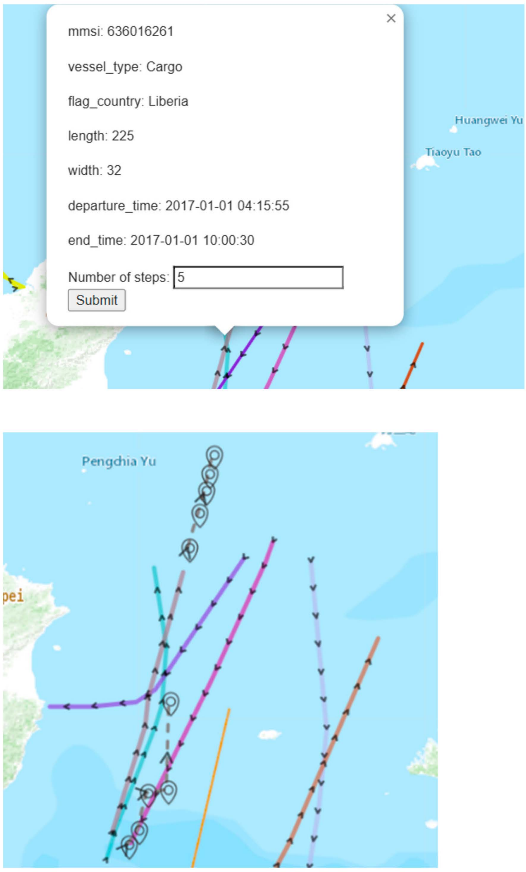


Figure 9. Visualization of the effect of trajectory prediction with a step size of 5. The arrows indicate the direction of trajectory travel, the localization icon represents the predicted trajectory point, the solid line represents the actual trajectory of the ship, and the dotted line represents the driving path of the real trajectory in the validation set.

6. Conclusions

Aiming at the problem of mining and forecasting vessel trajectory, this study presents a method of constructing a vessel trajectory prediction model based on the Attention Spatiotemporal graph Convolution network (ASTGCN). The core of this method is to capture the spatio-temporal characteristics of vessel trajectories by introducing spatio-temporal attention mechanism and spatio-temporal graph convolution. In addition, to adapt the model to the graph structure corresponding to the vessel trajectory and improve the prediction accuracy, a method to determine the prior matrix is also designed.

The experimental results proved the effectiveness of ASTGCN, which demonstrated significant advantages in several performance indicators. This is mainly due to the attention mechanism introduced in time and space dimensions, and the powerful processing ability of nonlinear trajectory data based on the graph structure of non-Euclidean space. This design allows the model to better capture major feature associations and spatiotemporal dependencies in vessel trajectory data. The excellent performance of ASTGCN indicates that the trajectory prediction model proposed in this study, based on the convolution network of attentional spatiotemporal graphs, has a good performance in capturing the spatiotemporal characteristics of vessel trajectory, which not only provides an accurate prediction for the future trajectory of vessels but also has a certain interpretability for the spatiotemporal attention mechanism of the network structure and the convolution of spatiotemporal graphs. It provides a more transparent and reliable forecasting basis for decision-makers.

Since the computational complexity of the ASTGCN model is mainly affected by the number of nodes, the number of neighbors, the filter parameters, and the convolution order (in the graph convolution layer), as well as the number of time steps and the vector dimensions (in the attention mechanism), the scalability of the ASTGCN model can be improved by considering the strategies of data parallelism, model parallelism, lightweight design, and algorithmic optimization in the subsequent study. This will allow for effective resource utilization and performance improvement when dealing with large spatio-temporal data.

Author Contributions: Conceptualization, X.C. and R.Z.; methodology, R.Z. and L.Y.; validation, W.Y.; investigation, B.Z.; resources, J.L.; writing—original draft preparation, R.Z. and L.Y.; writing—review and editing, X.C.; visualization, J.L.; supervision, X.C.; project administration, B.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 42371438 and No. 42301526).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data supporting the findings of this study can be found at https://www.hifleet.com/data/index.html (accessed on 23 October 2023).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Rong, H.; Teixeira, A.P.; Soares, C.G. Ship collision avoidance behaviour recognition and analysis based on AIS data. *Ocean Eng.* **2022**, *245*, 110479. [CrossRef]
2. Liang, M.; Weng, L.; Gao, R.; Li, Y.; Du, L. Unsupervised maritime anomaly detection for intelligent situational awareness using AIS data. *Knowl.-Based Syst.* **2024**, *284*, 111313. [CrossRef]
3. Galdelli, A.; Mancini, A.; Ferrà, C.; Tassetti, A.N. A synergic integration of AIS data and SAR imagery to monitor fisheries and detect suspicious activities. *Sensors* **2021**, *21*, 2756. [CrossRef]
4. Lee, J.S.; Lee, H.T.; Cho, I.S. Maritime traffic route detection framework based on statistical density analysis from AIS data using a clustering algorithm. *IEEE Access* **2022**, *10*, 23355–23366. [CrossRef]
5. Lee, H.-T.; Lee, J.-S.; Yang, H.; Cho, I.-S. An AIS Data-Driven Approach to Analyze the Pattern of Ship Trajectories in Ports Using the DBSCAN Algorithm. *Appl. Sci.* **2021**, *11*, 799. [CrossRef]

6. Qian, L.; Zheng, Y.; Li, L.; Ma, Y.; Zhou, C.; Zhang, D. A New Method of Inland Water Ship Trajectory Prediction Based on Long Short-Term Memory Network Optimized by Genetic Algorithm. *Appl. Sci.* **2022**, *12*, 4073. [CrossRef]
7. Guo, S.; Mou, J.; Chen, L.; Chen, P. Improved kinematic interpolation for AIS trajectory reconstruction. *Ocean Eng.* **2021**, *234*, 109256. [CrossRef]
8. Li, H.; Jiao, H.; Yang, Z. Ship trajectory prediction based on machine learning and deep learning: A systematic review and methods analysis. *Eng. Appl. Artif. Intell.* **2023**, *126*, 107062. [CrossRef]
9. Goerlandt, F.; Montewka, J. Maritime transportation risk analysis: Review and analysis in light of some foundational issues. *Reliab. Eng. Syst. Saf.* **2015**, *138*, 115–134. [CrossRef]
10. Chen, P.; Huang, Y.; Mou, J.; van Gelder, P. Probabilistic risk analysis for ship-ship collision: State-of-the-art. *Saf. Sci.* **2019**, *117*, 108–122. [CrossRef]
11. Zhang, J.; Zhang, D.; Yan, X.; Haugen, S.; Guedes Soares, C. A distributed anti-collision decision support formulation in multi-ship encounter situations under COLREGs. *Ocean Eng.* **2015**, *105*, 336–348. [CrossRef]
12. Zhen, R.; Riveiro, M.; Jin, Y. A novel analytic framework of real-time multi vessel collision risk assessment for maritime traffic surveillance. *Ocean Eng.* **2017**, *145*, 492–501. [CrossRef]
13. Wang, X.; Liu, Z.; Cai, Y. The ship maneuverability based collision avoidance dynamic support system in close-quarters situation. *Ocean Eng.* **2017**, *146*, 486–497. [CrossRef]
14. Zheng, K.; Chen, Y.; Jiang, Y.; Qiao, S. A SVM based ship collision risk assessment algorithm. *Ocean. Eng.* **2020**, *202*, 107062. [CrossRef]
15. Liu, J.; Shi, G.-Y.; Zhu, K.-G. A novel ship collision risk evaluation algorithm based on the maximum interval of two ship domains and the violation degree of two ship domains. *Ocean. Eng.* **2022**, *255*, 111431. [CrossRef]
16. Sotiralis, P.; Ventikos, N.P.; Hamann, R.; Golyshev, P.; Teixeira, A.P. Incorporation of human factors into ship collision risk models focusing on human centred design aspects. *Reliab. Eng. Syst. Saf.* **2016**, *156*, 210–227. [CrossRef]
17. Montewka, J.; Ehlers, S.; Goerlandt, F.; Hinz, T.; Tabri, K.; Kujala, P. A frameworkfor risk assessment for maritime transportation systems—A case study for open sea collisions involving RoPax vessels. *Reliab. Eng. Syst. Saf.* **2014**, *124*, 142–157. [CrossRef]
18. Jiang, M.; Lu, J.; Yang, Z.; Li, J. Risk analysis of maritime accidents along the main route of the Maritime Silk Road: A Bayesian network approach. *Marit. Policy Manag.* **2020**, *47*, 815–832. [CrossRef]
19. Inaishi, M.; Matsumura, H. Basic research on a collision avoidance system using neural networks. *J. Navig.* **1992**, *112*, 22–28.
20. Feng, X. Modelling of Regional Vessel Near Collision Risk Assessment with Convolutional Neural Network. In Proceedings of the Inaugural World Transport Convention, Beijing, China, 4–6 June 2019.
21. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In Proceedings of the International Conference on Machine Learning 2016, New York, NY, USA, 19–24 June 2016; pp. 1928–1937.
22. Xie, S.; Chu, X.; Zheng, M.; Liu, C. A composite learning method for multi-ship collision avoidance based on reinforcement learning and inverse control. *Neurocomputing* **2020**, *411*, 375–392. [CrossRef]
23. Zhang, R.; Wang, X.; Liu, K.; Wu, X.; Lu, T.; Chao, Z. Ship collision avoidance using constrained deep reinforcement learning. In Proceedings of the IEEE 2018 5th International Conference on Behavioral, Economic, and SocioCultural Computing (BESC), Kaohsiung, Taiwan, 12–14 November 2018; pp. 115–120.
24. Ilie, B.S.; Ion, R.S.; Daniel, C.C. A Review of Deep Learning-Based Methods for Pedestrian Trajectory Prediction. *Sensors* **2021**, *21*, 7543. [CrossRef]
25. Yuhao, W.; Yutian, P.; Oliver, C.; Hari, I.; Parikshit, D.; Menon, P.K.; Yongming, L. Uncertainty quantification and reduction in aircraft trajectory prediction using Bayesian-Entropy information fusion. *Reliab. Eng. Syst. Saf.* **2021**, *212*, 107650.
26. Pavlovic, V.; Rehg, J.M.; Cham, T.; Murphy, K.P. A dynamic Bayesian network approach to figure tracking using learned dynamic models. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; pp. 94–101.
27. Sadeghian, A.; Kosaraju, V.; Sadeghian, A.; Hirose, N.; Rezatofighi, H.; Savarese, S. SoPhie: An Attentive GAN for Predicting Paths Compliant to Social and Physical Constraints. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
28. Deng, M.; Li, S.; Jiang, X.; Li, X. Vehicle Trajectory Prediction Method Based on "Current" Statistical Model and Cubature Kalman Filter. *Electronics* **2023**, *12*, 2464. [CrossRef]
29. Ladekar, A.; Mohol, B.; Gaikwad, A.; Shingade, S.; Kulkarni, A.; Naval, Y. PULM: Prediction of User's Location using Machine Learning with Markov Model. In Proceedings of the 6th International Conference on Trends in Electronics and Informatics (ICOEI), Xiamen, China, 21–23 October 2022; pp. 1195–1199.
30. Ji, R.; Liang, Y.; Xu, L.; Wei, Z. Trajectory prediction of ballistic missiles using Gaussian process error model. *Chin. J. Aeronaut.* **2022**, *35*, 458–469. [CrossRef]
31. Antonios, K.; Adrian, J.; Michael, B. A Seq2Seq learning approach for modeling semantic trajectories and predicting the next location. In Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Washington, DC, USA, 6–9 November 2018; pp. 528–531.
32. Jiang, R.; Xu, H.; Gong, G.; Kuang, Y.; Liu, Z. Spatial-Temporal Attentive LSTM for Vehicle-Trajectory Prediction. *ISPRS Int. J. Geo-Inf.* **2022**, *11*, 354. [CrossRef]

33. Wang, X.; Xiao, Y. A Deep Learning Model for Ship Trajectory Prediction Using Automatic Identification System (AIS) Data. *Information* **2023**, *14*, 212. [CrossRef]
34. Yu, B.; Yin, H.; Zhu, Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In Proceedings of the International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 9–19 July 2018.
35. Guo, S.; Lin, Y.; Feng, N.; Song, C.; Wan, H. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. In Proceedings of the National Conference on Artificial Intelligence, Association for the Advancement of Artificial Intelligence (AAAI), New York, NY, USA, 5 September 2019; pp. 922–929.
36. Cao, D.; Wang, Y.; Duan, J.; Zhang, C.; Zhu, X.; Huang, C.; Tong, Y.; Xu, B.; Bai, J.; Tong, J.; et al. Spectral Temporal Graph Neural Network for Multivariate Time-series Forecasting. *arXiv* **2020**, arXiv:2103.07719.
37. Lin, Z.; Yue, W.; Huang, J.; Wan, J. Ship Trajectory Prediction Based on the TTCN-Attention-GRU Model. *Electronics* **2022**, *12*, 2556. [CrossRef]