mathematics

Article

Deep Learning for Vessel Trajectory Prediction Using Clustered AIS Data

Cheng-Hong Yang [1,2,3,4,5], Guan-Cheng Lin [2], Chih-Hsien Wu [2], Yen-Hsien Liu [2,*], Yi-Chuan Wang [6,*] and Kuo-Chang Chen [2,*]

[1] Department of Information Management, Tainan University of Technology, Tainan 71002, Taiwan
[2] Department of Electronic Engineering, National Kaohsiung University of Science and Technology, Kaohsiung 80778, Taiwan
[3] Ph.D. Program in Biomedical Engineering, Kaohsiung Medical University, Kaohsiung 80708, Taiwan
[4] School of Dentistry, Kaohsiung Medical University, Kaohsiung 80708, Taiwan
[5] Drug Development and Value Creation Research Center, Kaohsiung Medical University, Kaohsiung 80708, Taiwan
[6] Department of Business Administration, CTBC Business School, Tainan 709, Taiwan
[*] Correspondence: I109152106@nkust.edu.tw (Y.-H.L.); ycwang@ctbc.edu.tw (Y.-C.W.); I109152104@nkust.edu.tw (K.-C.C.)

Abstract: Accurate vessel track prediction is key for maritime traffic control and management. Accurate prediction results can enable collision avoidance, in addition to being suitable for planning routes in advance, shortening the sailing distance, and improving navigation efficiency. Vessel track prediction using automatic identification system (AIS) data has attracted extensive attention in the maritime traffic community. In this study, a combining density-based spatial clustering of applications with noise (DBSCAN)-based long short-term memory (LSTM) model (denoted as DLSTM) was developed for vessel prediction. DBSCAN was used to cluster vessel tracks, and LSTM was then used for training and prediction. The performance of the DLSTM model was compared with that of support vector regression, recurrent neural network, and conventional LSTM models. The results revealed that the proposed DLSTM model outperformed these models by approximately 2–8%. The proposed model is able to provide a better prediction performance of vessel tracks, which can subsequently improve the efficiency and safety of maritime traffic control.

Keywords: automatic identification system; density-based spatial clustering of applications with noise; long short-term memory

MSC: 68T07

check for updates

Citation: Yang, C.-H.; Lin, G.-C.; Wu, C.-H.; Liu, Y.-H.; Wang, Y.-C.; Chen, K.-C. Deep Learning for Vessel Trajectory Prediction Using Clustered AIS Data. *Mathematics* **2022**, *10*, 2936. https://doi.org/10.3390/math10162936

Academic Editors: Wanquan Liu, Xianchao Xiu and Xuefang Li

Received: 10 July 2022
Accepted: 10 August 2022
Published: 15 August 2022

1. Introduction

Approximately 79% of the world's surface area is covered by water, and more than 80% of all cargo is transported over the ocean [1,2]. Maritime traffic accidents often result in large property losses and environmental damage. Therefore, the development of measures for improving maritime traffic safety has attracted research attention. The availability of maritime information collected by the automatic identification system (AIS) has created an unprecedented opportunity to avoid maritime accidents and improve maritime situational awareness. AIS collects various maritime surveillance data, including maritime traffic spatial information, to provide accurate early warning information to maritime traffic participants and to support various navigation operation decisions [3]. The widespread adoption of AIS has resulted in the accumulation of a large quantity of ship navigation data [4]; approximately 1 trillion pieces of information are stored in the system every day. Thus, ship transportation has also entered the era of big data. Numerous researchers have proposed various machine learning methods, theories, and technologies for big data processing and exploration. Methods of fully exploiting AIS data and promoting the intellectualization of maritime transportation have thus become a key research priority [5,6].

Numerous studies have been conducted on trajectory prediction problems. For example, studies predicted the trajectory of human actions [7] and that of vehicles or aircraft [8]. The Markov model is the most commonly used model for trajectory prediction [9]. In recent years, neural networks have also been used for trajectory prediction [10]. However, the aforementioned studies have mainly focused on trajectory prediction in confined areas, such as public spaces, road intersections, and highways. To the best of our knowledge, the previous studies have not comprehensively explored trajectory prediction using sequence-to-sequence models in a broad geographical area (such as the ocean). A possible reason for this research gap is that these predictions are highly complex due to the presence of various influencing factors (such as tide level, wind speed, and weather conditions) and differences in ship characteristics (such as between fishing boats, cargo ships, and ferries). In the present study, we used DBSCAN to identify similar tracks and used various neural network models, namely support vector regression (SVR), recurrent neural networks (RNN), and LSTM models, to produce prediction models for solving various tracks and schemes. A prediction system for the destinations and arrival times of maritime traffic must be accurate [11,12], efficient, and scalable. Such a system must also be applicable to large and complex geographical areas, such as oceans or large cities. Deep learning methods have been successfully applied for sequence-to-sequence prediction, in which an input sequence is mapped to a predicted output sequence. Long short-term memory (LSTM) is an effective and scalable sequence-to-sequence prediction model. Due to the lag period between the input sequence and the output prediction, LSTM is the best choice for track prediction as it demonstrates high suitability for learning data with a long-term time dependence [13].

Since AIS data always contain inaccurate and uncertain noise, outlier detection and filtering are required during organizing and modeling with AIS data. Additionally, given the significant size of the AIS dataset, unsupervised learning is in modeling and anomaly detection processes. Scholars developed various methods to identify similar tracks. For example, in 1914, Hausdorff proposed the Hausdorff distance algorithm, which uses a similarity measure between two points, and if each point of X is close to y, the set X and y are considered similar. The Hausdorff distance is calculated as the maximum of all minimum distances from each point in X to any point in y. However, if the number of calculations for the Hausdorff distance is excessive, the metric is prone to noise [14]. The Douglas–Peucker algorithm downsamples a curve comprising line segments into similar curves with fewer points. It is one of the earliest algorithms successfully used in cartographic generalization, which is used in processing vector graphics and cartography [15]. These two algorithms both typically produce good results for adjacent tracks. On the basis of these algorithms, one can consider that if similar tracks must be in the same group, a variable must be created to represent the characteristics of the track paths. Thus, the clustering model provides foundations for route planning, monitoring vessel behaviors, and detecting anomalies. By organizing similar AIS data and clustering them together, vessel behaviors can be profiled under labeled clusters, with each representing a specific vessel behavior stage. Within each behavior stage, the vessel behaviors share maximum similarities and are different from other clusters.

Trajectory data for a ship can be confirmed from its past motion and mode. AIS data for ships in the sea near Taiwan can be downloaded from a public database. These AIS data comprise various pieces of information, including ship speed over ground (SOG), course over ground (COG), type, length, and width. Because raw AIS data are not preprocessed, some data items may be lost or inaccurate; hence, quality control methods such as track separation, data denoising, and standardization must be applied to process the data. In this study, abnormal values in raw AIS data were identified by using the moving average to smooth the SOG data, standardizing the noisy data, and finally subtracting the COG data to judge the data. COG does not change due to fluctuations; thus, any changes in COG are due to a change in the ship's heading relative to the ground. Accordingly, changes in COG were used to identify abnormal values. Subsequently, DBSCAN was used to determine the characteristics of the ship track and similar tracks [16], and LSTM was used to make predictions for similar ships. These predictions were then compared with those of support vector machine (SVM) [17] and RNN models. The results revealed that among all models included in this study, the proposed DLSTM model had the best performance, which exceeded that of the other models by approximately 2–8%. If more similar trajectories were used in the training model, both the data accuracy and the model's track prediction could be improved. Accordingly, in this study, a density-based spatial clustering of applications with noise (DBSCAN)-based long short-term memory (LSTM) model (hereafter denoted as DLSTM) was developed for vessel prediction. Moreover, a recurrent neural network (RNN) model, a clustering approach, and a prediction model were used to predict future ship tracks on the basis of a given AIS observation sequence for comparison.

2. Related Work

The challenges of automation differ considerably between sea and land scenarios. Cars on the road typically travel in a lane; by contrast, ships can travel in any direction. Moreover, compared with automobile traffic rules, the International Regulations for Preventing Collisions at Sea (COLREGs) are less strictly codified and depend on operator experience, thus increasing the difficulty of predicting ship behavior. Early studies on ship track prediction have relied on the physical motion of a ship, which is mainly based on a curve model [18], transverse model [19,20], and ship model [21,22]. The physical motion of a ship can be expressed by combining mathematical equations and physical laws in which all possible influencing factors, such as mass, size, and inertia, are considered. The accuracy of such models depends on the accuracy of assumptions about the environment and the ideal state; making such assumptions is difficult in most real-world ship track prediction tasks. Numerous studies executed track prediction processes by using AIS data and techniques or models such as the Kalman filter model [23], Markov model [9], and optimal route estimation based on clustering and the ant colony algorithm [24]. Some researchers also applied neural networks to solve prediction problems without using professional knowledge. For example, Wang et al., used two-way gated recurrent units (GRU) to predict ship tracks [25]. Thus, neural networks play an important role in trajectory prediction [26–28].

Large collections of AIS data can be used in various research fields and for ship prediction to maximize the public benefit of these data. Trajectory prediction is a challenging research problem because it must be completed rapidly with both high accuracy and efficiency. Numerous studies in various fields have predicted the trajectories of various entities, including the trajectories of human actions [7] and those of aircraft and vehicles [29]. Trajectory data constitute a unique type of time series data that includes both spatial and temporal dimensions [30]. Conventionally, the Kalman filter is applied to waypoints to predict ship trajectories. Advancements in deep learning have enhanced the effectiveness of RNNs and variational automatic encoders in trajectory prediction [31]. AIS data are crucial for constructing a ship operation model, and such data contain information about COG, SOG, turn rate, and navigation. These data enable understanding of the course and running track of a ship, which can help prevent ship accidents [11].

Neural networks can be considered as a set of weighted connections between points that mimic neurons in the human brain. Neural networks can learn, imitate, and predict the behavior of a system [32]. The algorithms of neural networks can classify input data to identify correlations between the data items, and such algorithms can continually learn and improve over time. They are ideal for solving complex real-world problems. Support Vector Regression (SVR), an extension of the SVM model, is a machine learning method proposed by the Russian statistician Vapnik [33]. It has been successfully applied for predicting ship trajectories. In SVR, statistical learning is used to find a hyperplane with minimal distance to each of the two sets. SVR has three parameters: C, the penalty constant; ε, the loss function; and σ, the width of the kernel function. Recurrent neural networks (RNNs) were proposed by David Rumelhart in 1986 [34]. However, RNNs are simple and cannot handle recursively accumulated weight indices, which may lead to gradient explosion or gradient disappearance [35]. Therefore, Hochreiter and Schmidhuber proposed LSTM as an improvement to RNNs in 1997. LSTM has been used in numerous fields, including text classification [36], stock analysis [37], and trajectory prediction [38]. RNNs have lower computational requirements than LSTM; however, LSTM has outstanding feature selection performance, resulting in improved track prediction.

3. Method

3.1. Study Design

The steps involved in the study method are detailed in the flowchart presented in Figure 1. The main steps are as follows: (a) collecting ship track, speed, and course data; (b) cleaning the data through track separation, outlier deletion, and data standardization; (c) calculating changes in COG track; (d) preliminarily clustering the collected tracks by using DBSCAN; (e) applying the RNN, SVR or LSTM model to predict the trajectory from the denoised data; (f) predicting the ship tracks using the proposed model and comparing the prediction results with those obtained by other algorithms; (g) comparing the predicted trajectory with the original trajectory data to verify the accuracy of the proposed model.

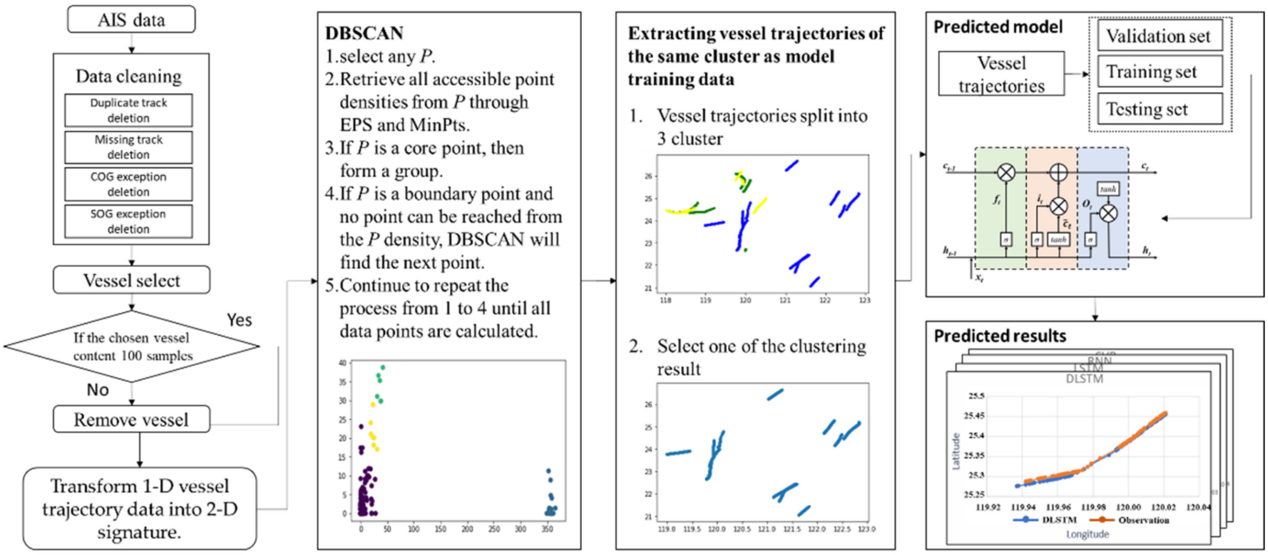

AIS data
Data cleaning
Duplicate track deletion
Missing track deletion
COG exception deletion
SOG exception deletion
Vessel select
If the chosen vessel content 100 samples
Yes
No
Remove vessel
Transform 1-D vessel trajectory data into 2-D signature.
DBSCAN
1.select any P.
2.Retrieve all accessible point densities from P through EPS and MinPts.
3.If P is a core point, then form a group.
4.If P is a boundary point and no point can be reached from the P density, DBSCAN will find the next point.
5.Continue to repeat the process from 1 to 4 until all data points are calculated.
Extracting vessel trajectories of the same cluster as model training data
1. Vessel trajectories split into 3 cluster
2. Select one of the clustering result
Predicted model
Vessel trajectories
Validation set
Training set
Testing set
Predicted results
RNN
LSTM
DLSTM
Latitude
Longitude
DLSTM
Observation

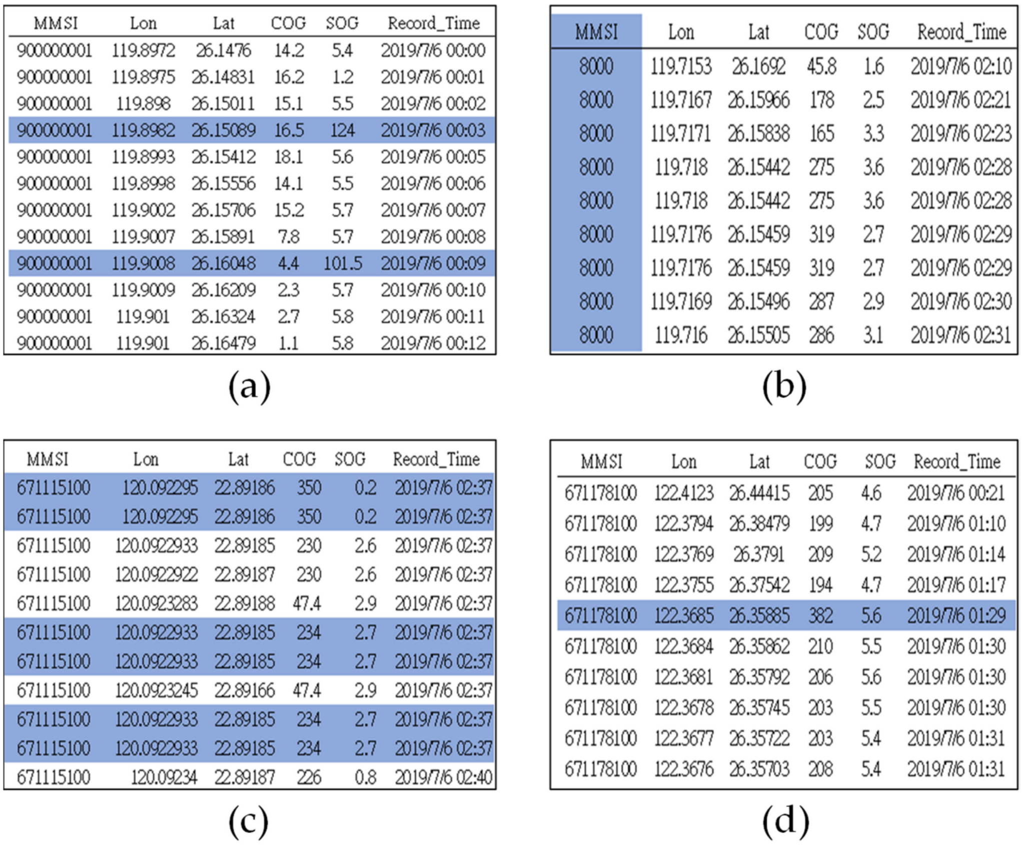
Figure 1. Flowchart of the study method.

3.1.1. Collection of Data on Vessel Trajectory, Speed, Course, and Other Features

The AIS data set was downloaded from a public database on 6 July 2019, and each entry included the time and the maritime mobile service identity (MMSI), SOG, COG, and boat length of a ship. A total of 1,048,256 entries for 1326 ships were collected. After the clustering process, six ships were selected for trajectory prediction and comparison.

3.1.2. Data Cleaning through Track Separation and Outlier Deletion

AIS data may contain abnormalities that must be removed before vessel track analysis. The quality of the collected AIS vessel trajectory data was enhanced through data cleaning, data standardization, and deduplication. The raw AIS data set contained numerous ship features for hundreds of ships. The MMSI, a unique identifier for ships, was used to classify the AIS data by ship. Classification by ship improves the accuracy of neural network prediction and enables the establishment of different prediction models for the trajectories of different vessels. Generally, four types of erroneous and noisy effects on the AIS data, including (a) abnormalities that exist in the data in the SOG column. It is difficult for a vessel to accelerate or decelerate very quickly over a short period (it would require 1 s to speed up from 5.5 to 124 knots); therefore, the values highlighted are deemed abnormal. (b) The MMSI column is abnormal; the standard MMSI length should be 9. (c) Duplicate data, which are not conducive to neural network training and may cause overfitting, are removed. (d) Abnormalities exist in the data in the COG column; COG values range from 0 to 360. Figure 2 presents the example of four common error types in AIS data.



MMSI	Lon	Lat	COG	SOG	Record_Time
900000001	119.8972	26.1476	14.2	5.4	2019/7/6 00:00
900000001	119.8975	26.14831	16.2	1.2	2019/7/6 00:01
900000001	119.898	26.15011	15.1	5.5	2019/7/6 00:02
900000001	119.8982	26.15089	16.5	124	2019/7/6 00:03
900000001	119.8993	26.15412	18.1	5.6	2019/7/6 00:05
900000001	119.8998	26.15556	14.1	5.5	2019/7/6 00:06
900000001	119.9002	26.15706	15.2	5.7	2019/7/6 00:07
900000001	119.9007	26.15891	7.8	5.7	2019/7/6 00:08
900000001	119.9008	26.16048	4.4	101.5	2019/7/6 00:09
900000001	119.9009	26.16209	2.3	5.7	2019/7/6 00:10
900000001	119.901	26.16324	2.7	5.8	2019/7/6 00:11
900000001	119.901	26.16479	1.1	5.8	2019/7/6 00:12

(a)

MMSI	Lon	Lat	COG	SOG	Record_Time
8000	119.7153	26.1692	45.8	1.6	2019/7/6 02:10
8000	119.7167	26.15966	178	2.5	2019/7/6 02:21
8000	119.7171	26.15838	165	3.3	2019/7/6 02:23
8000	119.718	26.15442	275	3.6	2019/7/6 02:28
8000	119.718	26.15442	275	3.6	2019/7/6 02:28
8000	119.7176	26.15459	319	2.7	2019/7/6 02:29
8000	119.7176	26.15459	319	2.7	2019/7/6 02:29
8000	119.7169	26.15496	287	2.9	2019/7/6 02:30
8000	119.716	26.15505	286	3.1	2019/7/6 02:31

(b)

MMSI	Lon	Lat	COG	SOG	Record_Time
671115100	120.092295	22.89186	350	0.2	2019/7/6 02:37
671115100	120.092295	22.89186	350	0.2	2019/7/6 02:37
671115100	120.0922933	22.89185	230	2.6	2019/7/6 02:37
671115100	120.0922922	22.89187	230	2.6	2019/7/6 02:37
671115100	120.0923283	22.89188	47.4	2.9	2019/7/6 02:37
671115100	120.0922933	22.89185	234	2.7	2019/7/6 02:37
671115100	120.0922933	22.89185	234	2.7	2019/7/6 02:37
671115100	120.0923245	22.89166	47.4	2.9	2019/7/6 02:37
671115100	120.0922933	22.89185	234	2.7	2019/7/6 02:37
671115100	120.0922933	22.89185	234	2.7	2019/7/6 02:37
671115100	120.09234	22.89187	226	0.8	2019/7/6 02:40

(c)

MMSI	Lon	Lat	COG	SOG	Record_Time
671178100	122.4123	26.44415	205	4.6	2019/7/6 00:21
671178100	122.3794	26.38479	199	4.7	2019/7/6 01:10
671178100	122.3769	26.3791	209	5.2	2019/7/6 01:14
671178100	122.3755	26.37542	194	4.7	2019/7/6 01:17
671178100	122.3685	26.35885	382	5.6	2019/7/6 01:29
671178100	122.3684	26.35862	210	5.5	2019/7/6 01:30
671178100	122.3681	26.35792	206	5.6	2019/7/6 01:30
671178100	122.3678	26.35745	203	5.5	2019/7/6 01:30
671178100	122.3677	26.35722	203	5.4	2019/7/6 01:31
671178100	122.3676	26.35703	208	5.4	2019/7/6 01:31

(d)

Figure 2. Four types of erroneous and noisy effects. (**a**) Abnormalities exist in the data in the SOG column; (**b**) MMSI column is abnormal; (**c**) Duplicate data; (**d**) Abnormalities exist in the data in the COG column.

The data were acquired from ships off the island of Taiwan at latitudes between 20° and 25° N and longitudes between 120° and 123° E. Raw AIS data collected from a single vessel were cleaned by deleting duplicate data and abnormal COG, SOG, and MMSI data. Figure 2 shows examples of noisy AIS data. For example, manual inspection of the original AIS data revealed abnormal vessel speeds exceeding 30 knots; such abnormal data were deleted.

3.2. Trajectory Similarity Measurement

A ship's course changes in response to changes in the winds and currents. The COG is the ship's actual direction of travel and does not change unless the ship's course changes. Figure 3 presents the relationship between COG and a vessel. At point B, the vessel is under heavy wind and changes its heading to 135° to counteract the effects of the wind. Despite the change in heading, the ship's actual course (i.e., COG) does not change due to the wind and remains 90°. After the wind fades at point C, the ship's heading is still 135°; thus, its COG becomes 135° at point D. In order to return to its original course, the ship turns left to return its heading to 90°, and at point E, both the heading and COG are 90°. The trajectory of a ship can be determined by using left- and right-turn data obtained from COG data (Table 1). An increase or decrease in COG indicates a right or left turn, respectively. Figure 4 presents (for all vessels): Each unit in Figure 4 represents 5 entries, with the yellow and red plots indicating right and left turns, respectively. Figure 5 presents the vessel's two-dimensional (2D) trajectory calculated from the one-dimensional (1D) COG data. Figure 6 illustrates the algorithm used for this transformation in this study.

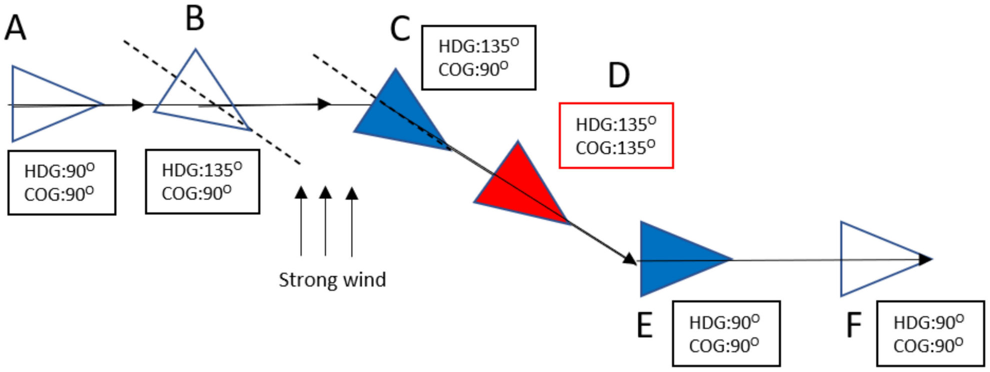


Figure 3. Judgment of COG transform.

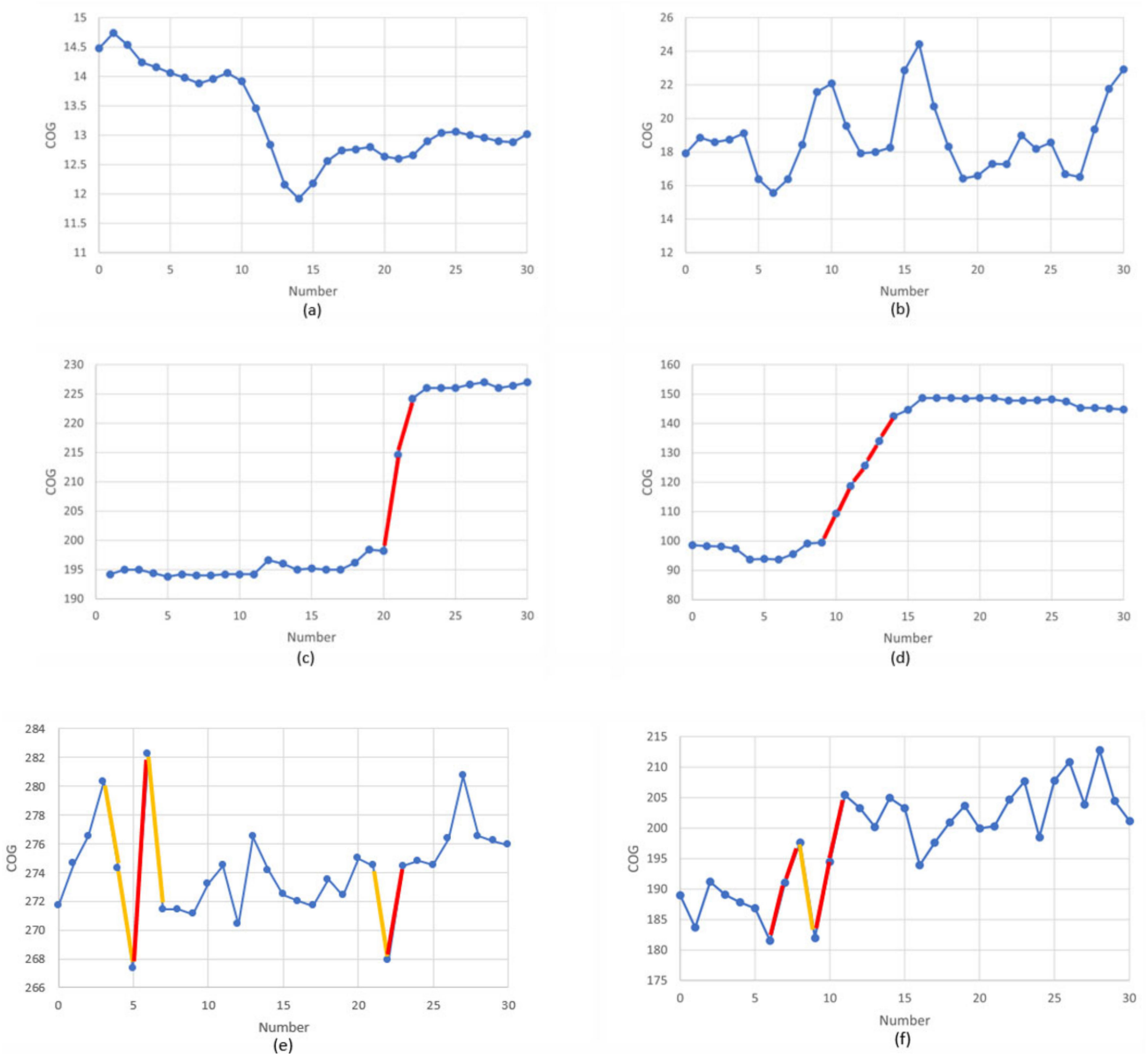


Figure 4. The COG transforms of vessel (**a**) 411747900; (**b**) 215280000; (**c**) 477118100; (**d**) 413484000; (**e**) 413697130; (**f**) 413705580. Red line, left turn; Yellow line, right turn.

Table 1. Score of vessel trajectory after signature transform.

Index	MMSI	Left_Score	Right_Score
1	538006353	0	0
2	538006573	0	1.52
3	538005414	4.448	4.252
4	538004214	1.44	0
5	477946000	39.36	3.84
6	477943000	1.116	6.16
7	477938400	8.948	9.14
8	477800200	20.32	0
9	477698600	1.964	1.916

MMSI, maritime mobile service identity.

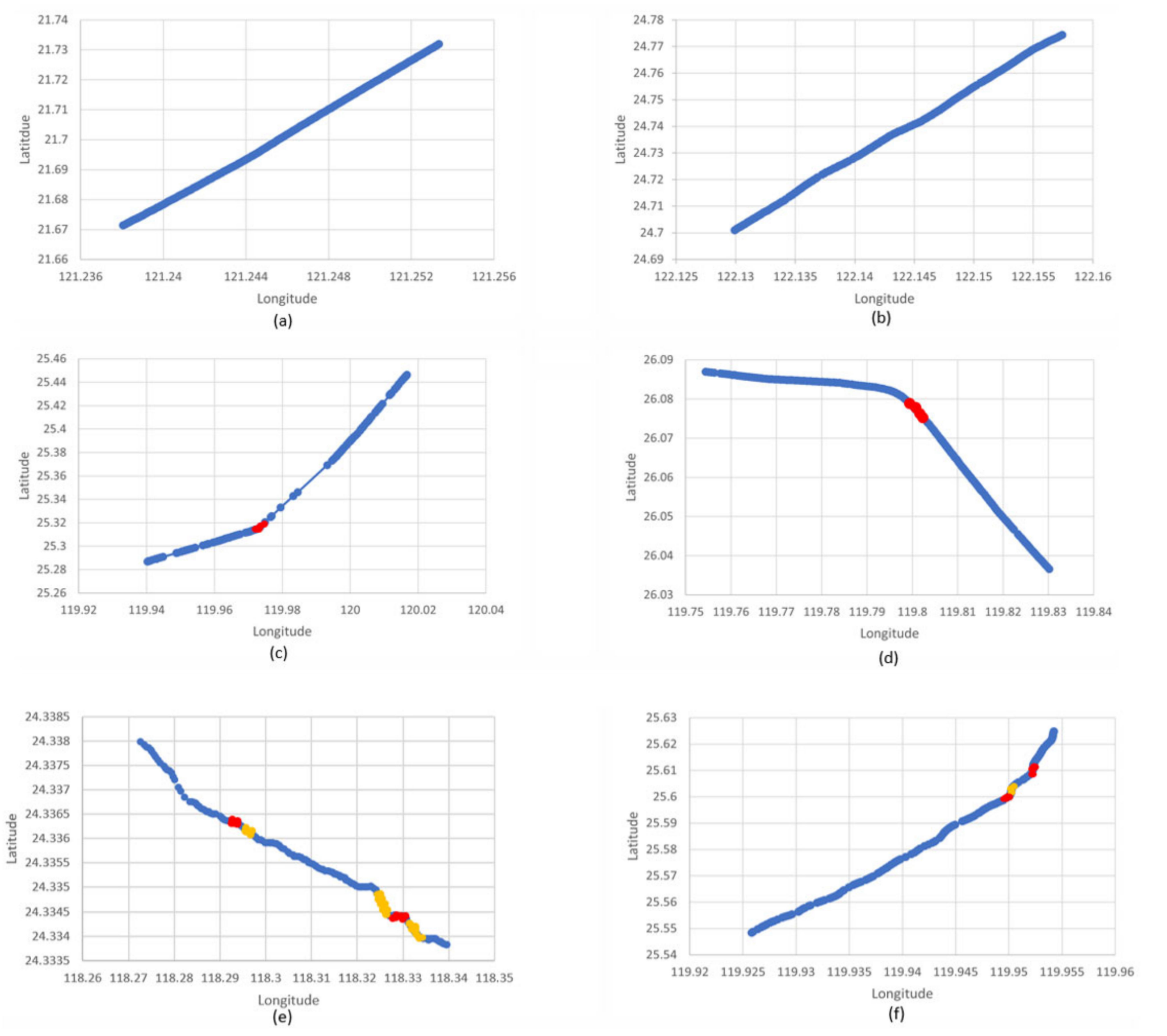


Figure 5. Verify The COG transforms of vessel (**a**) 411747900; (**b**) 215280000; (**c**) 477118100; (**d**) 413484000; (**e**) 413697130; (**f**) 413705580. Red line, left turn; Yellow line, right turn.

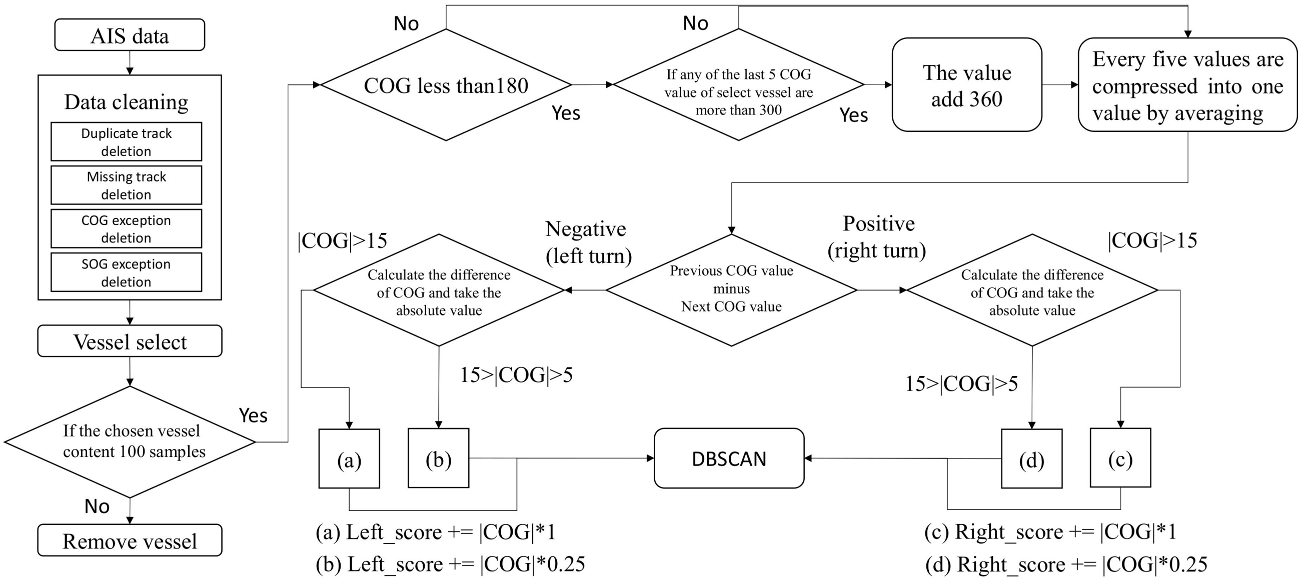

AIS data
Data cleaning
Duplicate track deletion
Missing track deletion
COG exception deletion
SOG exception deletion
Vessel select
If the chosen vessel content 100 samples
Yes
No
Remove vessel
No
COG less than180
Yes
No
If any of the last 5 COG value of select vessel are more than 300
Yes
The value add 360
Every five values are compressed into one value by averaging
|COG|>15
Calculate the difference of COG and take the absolute value
Negative (left turn)
Previous COG value minus Next COG value
Positive (right turn)
Calculate the difference of COG and take the absolute value
|COG|>15
15>|COG|>5
15>|COG|>5
(a)
(b)
DBSCAN
(d)
(c)
(a) Left_score += |COG|*1
(b) Left_score += |COG|*0.25
(c) Right_score += |COG|*1
(d) Right_score += |COG|*0.25

Figure 6. Transform 1D vessel trajectory data into 2D signature.

3.3. DBSCAN

Clustering analysis is a common data mining technique. In this analysis, a set of points are grouped into several clusters such that points in the same cluster are more similar to each other than to those in other clusters. The DBSCAN model can find clusters of arbitrary shapes and remove noisy data. The model requires only one search to obtain the final clustering result; thus, it is highly efficient. The main concept of a density-based algorithm such as the DBSCAN model is that for each point in a cluster, its neighborhood with a given radius must contain at least a minimum number of points. Moreover, the density of the points in the same cluster must be larger than that of points in different clusters [39]. The density in areas that are mostly noise should be lower than the density of any cluster. To identify a cluster, the DBSCAN model begins searching from an arbitrary object p at all search points and retrieves all points that are density-reachable from p with respect to the minimum radius around the point, eps (epsilon), and the minimum number of points in a cluster (MinPts). If p is a core point, then the neighborhood of p can be obtained with respect to eps and MinPts. Points in this neighborhood are used as seeds in the next circle to expand the cluster. If p is a border point and no points are density-reachable from p, p is temporarily determined to be noise. The DBSCAN model then chooses the next point in the database for clustering. The DBSCAN model [40] requires setting the global parameters eps and MinPts in advance. The original DBSCAN algorithm selects the optimal parameters by using the k-nearest neighbors of each point. The pseudocode of the DBSCAN model is presented in Algorithm 1.

3.4. Support Vector Regression (SVR)

Based on the structural risk minimization principle proposed by Vapnik, a loss function ε was derived, and an SVR model was developed to solve nonlinear problems, especially time series prediction problems. SVR was proposed by Vapnik et al. in 1997 [41] and has been used in numerous forecasting tasks such as short-term load forecasting [42] and monthly rainfall forecasting [43]. In order to obtain favorable forecasting performance, all three hyperparameters (C, ε, and σ, a kernel parameter) of the SVR model must be properly selected. These hyperparameters are usually determined through data resampling, which is computationally time-consuming. Thus, an efficient approach to simultaneously determine all parameters is necessary.

Algorithm 1 DBSCAN

```
DBSCAN ($eps$, $minpts$, $D$)
mark all patterns in D as unvisited
$cluid \leftarrow 1$
for each unvisited pattern $x$ in $D$
  do
  $Z \leftarrow$ Find Neighbours ($x$, $eps$, $minpts$)
  if $|Z| < minpts$
  mark $x$ as noise
  else
  mark $x$ and each pattern of Z with cluid
  queue_list ← all unvisited patterns of Z
  until queue_list is empty
  do
  y ← delete a pattern from queue_list
  $Z \leftarrow$ Find Neighbours ($x$, $eps$, $minpts$)
  if $|Z| \geqq minpts$
    for each pattern $w$ in $Z$
    mark $w$ with cluid
    if $w$ is unvisited
    queue_list ← $w$ U queue_list
    end for
  end if
  mark $x$ as visited
  $cluid \leftarrow cluid + 1$
end for
Output all patterns in $D$ marked with cluid or noise
```

3.5. Recurrent Neural Network (RNN)

RNNs were proposed by Rumelhart in 1986 for extracting long-term dependency in sequential data [44,45]. An RNN has a unique memory unit that enables it to be used for short sequence prediction. However, in practical applications, the length of the problem sequence is not known, which may result in gradient vanishing or gradient explosion during the learning process. Therefore, the practical applications of RNNs are limited. However, variants of RNNs exist, such as LSTM [13] and GRU [46]. A simple RNN has only one internal memory unit h_t, which can be expressed as follows:

$$h_t = f\left(Wx_t + U_f h_{t-1} + b\right) \tag{1}$$

where f is the activation function, U and W are the weight matrices of the hidden layer, b is the bias, and x_t is the input vector at time t [47].

3.6. Long Short-Term Memory (LSTM)

LSTM is an extension of RNNs [13]. The LSTM model is versatile in handling parameters with high dimensionality and involves nonlinear activation functions in each layer; therefore, it can capture nonlinear trends in data and remember previous information over long sequences. Accordingly, LSTM was successfully applied to numerous time series problems. The advantage of the LSTM structure is that it contains three types of gates: input, forget, and output gates. As displayed in Figure 7, the LSTM model solves the vanishing gradient problem of RNNs and enables the long-term storage of information.

The symbols $\oplus$ and $\otimes$ denote addition and multiplication in the model, and the arrow denotes the flow of information. The first layer of the memory gate determines the removal of unnecessary information from the cell state and can be expressed as follows:

$$f_t = \sigma\left(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f\right) \tag{2}$$

where f_t denotes forgetting threshold at time t, σ denotes the sigmoid activation function, f_t and U_f denote the weights, x_t denotes the input value, h_{t-1} denotes the output value at time $t-1$, and b_f denotes the bias term.

The input gate determines which information from the current input vector should be stored in the cell state. This includes the decision i_t, which updates the value and *tanh* layer for generating a new state value c_t. The specific expressions are as follows:

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i) \tag{3}$$

$$\widetilde{c}_t = \sigma(W_c \cdot x_t + U_c \cdot h_{t-1} + b_c) \tag{4}$$

where i_t represents the input threshold at time t; W_i, U_i, W_c and U_c represent the weights; and b_i and b_c represent bias terms. The term for updating the state of the cell at time t can be expressed as follows:

$$c_t = \sigma f_t \times c_{t-1} + i_t \times \widetilde{c}_t \tag{5}$$

The third layer is used to produce output information in the current time step and can be expressed as follows:

$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \tag{6}$$

where o_t denotes the output threshold at time t, W_o and U_o denote the weights, and b_o denotes the bias term. Accordingly, the output value of the cell can be expressed as follows:

$$h_t = o_t \times tanh(c_t) \tag{7}$$

where h_t denotes the output value of the cell at time t, *tanh* denotes the activation function, and c_t denotes the state of the cell at time t. After the data pass through the three gates, the effective information is the output, and the invalid information is forgotten.

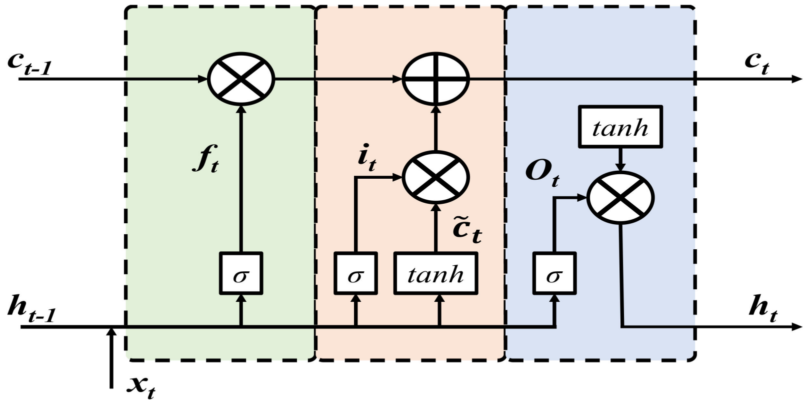


Figure 7. Architecture of LSTM.

3.7. Performance Verification

This study verified the predictive performance of the proposed method by using relevant performance metrics, namely mean absolute percentage error (MAPE), root mean square error (RMSE), p-value, and determination coefficient (R^2) score. Each metric can be expressed as follows:

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - f_i}{y_i}\right| \times 100\% \tag{8}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - f_i)^2} \tag{9}$$

$$p\text{-value} = \frac{(\overline{X} - \mu)}{s/\sqrt{n}} = \frac{\sqrt{n}(\overline{X} - \mu)}{s} \tag{10}$$

where n represents the number of samples, $\overline{X}$ represents the average value, s represents the sample standard deviation, and $\mu = \mu_0$ represents the null hypothesis. For distribution with $n - 1$ degrees of freedom,

$$\mathrm{R}^2 = \left[\frac{\sum_{i=1}^{n} (\eta_{io} - \overline{\eta}_o)(\eta_{iM} - \overline{\eta}_M)}{\sqrt{\sum_{i=1}^{n} (\eta_{io} - \overline{\eta}_o)^2 \sum_{i=1}^{n} (\eta_{iM} - \overline{\eta}_M)^2}} \right]^2, \tag{11}$$

where η_{io} is the observation track point of the ith time step, η_{iM} is the corresponding predicted trajectory point, n is the time step, $\overline{\eta}_o$ is the average of the observations, and $\overline{\eta}_M$ is the average of the predictions.

4. Result and Discussions

4.1. Clustering

Trajectory clustering has attracted growing attention because trajectory data mining plays a critical role in modern intelligent systems for navigation, surveillance, security, abnormal behavior detection, crowd behavior analysis, and traffic control. Similar ship trajectories were identified by first determining the left and right turns of ships in the AIS data. Special trajectories were effectively excluded. If there is a nontrend navigation, COG changes have no research value. After data clustering using the DBSCAN method, this study divided the data into four groups on the basis of their characteristics. Of these four groups, three with obvious characteristics (yellow, blue, and green) were used as the test data (Figure 8). The description of data samples in this study is shown in Table 2.

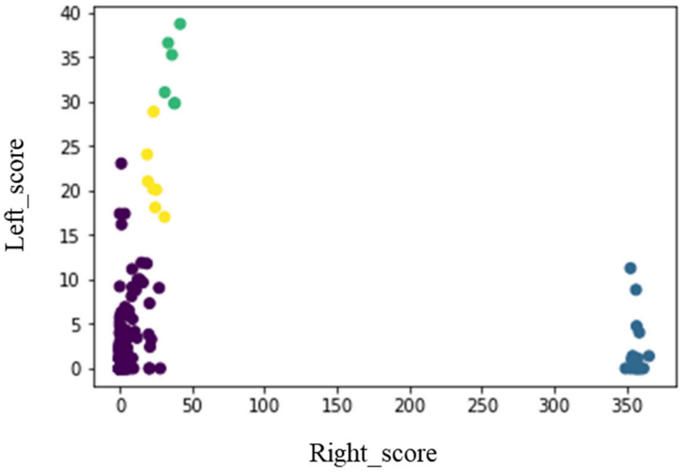


Figure 8. Clustering results of the DBSCAN. Green, cluster 1; Blue, cluster 2; Yellow, cluster 3; Purple, cluster 4.

Table 2. Description of samples in this study.

MMSI	Cluster	Cluster Data	AIS Data	DLSTM Train Data	Train Data	Test Data
477147900 215280000	1	15,372	1819 1188	13,553 14,184	1455 950	364 238
477118100 413484000	2	5662	1500 786	4162 4876	1200 629	300 157
413697130 413705580	3	4004	736 533	3268 3471	589 426	147 107

The DBSCAN model parameters were set as follows: eps = 4 and MinPts = 5. The results are presented in Figure 8, where yellow, blue, green, and purple represent different trajectory clustering results. Left- and right-turn data were used as the basis for the clustering process (Figure 9a). Further cluster analysis of the relevant results revealed 15,372 trajectories in the blue cluster, most of which were characterized by straight lines (Figure 9b). Moreover, a total of 5662 trajectories were observed in the yellow cluster; these trajectories were slightly more curved than the blue cluster trajectories (Figure 9c). Finally, 4004 trajectories were noted in the green cluster; these trajectories had the greatest curvature (Figure 9d).

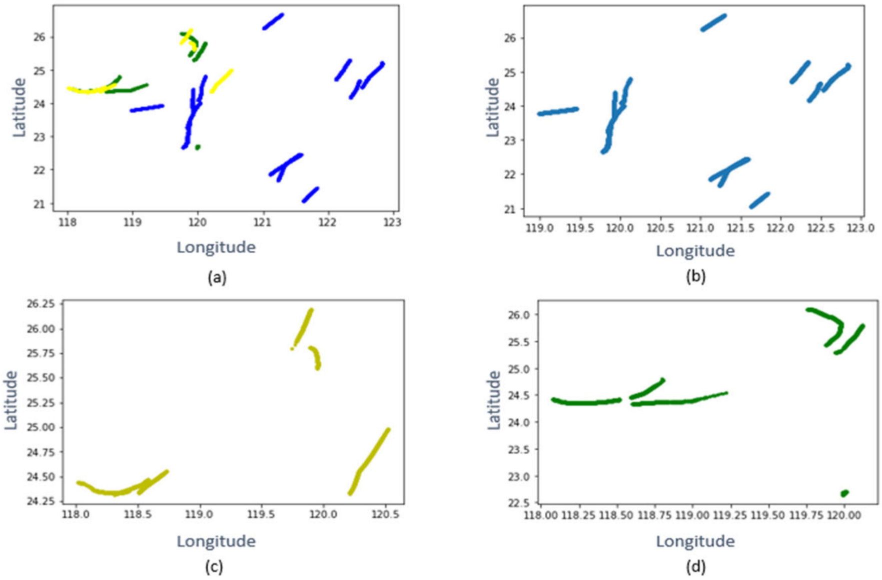


Figure 9. Vessel trajectories through clustering. (**a**) Corresponding trajectory clustering results; Green, cluster 1; Blue, cluster 2; Yellow, cluster 3; (**b**) Vessel trajectory of cluster blue; (**c**) Vessel trajectory of cluster yellow; (**d**) Vessel trajectory of cluster green.

4.2. Parameter Settings

In this study, trial and error were used to optimize the prediction accuracy of the DLSTM. Taking MMSI 413697130 as an example, using the unit numbers 50, 80, 128, and 256 to perform the test, it was found that unit 80 in the first layer obtained MAPE 0.026 and RMSE 0.01 was better than the other three units. Thus, unit 80 is the best value selected in the first layer. Same as the first layer determination, unit 50 has the best performance in the second layer with MAPE 0.019 and RMSE 0.009, which is better than the other three unit combinations. Therefore, for the case of MMSI 413676130, the best units for the first layer and second layer are 80 and 50, respectively, which is significantly better than the other combinations, as shown in Table 3. The parameter setting of DLSTM is shown in Table 4.

Table 3. Vessel trajectory prediction performance on different parameter settings for DLSTM.

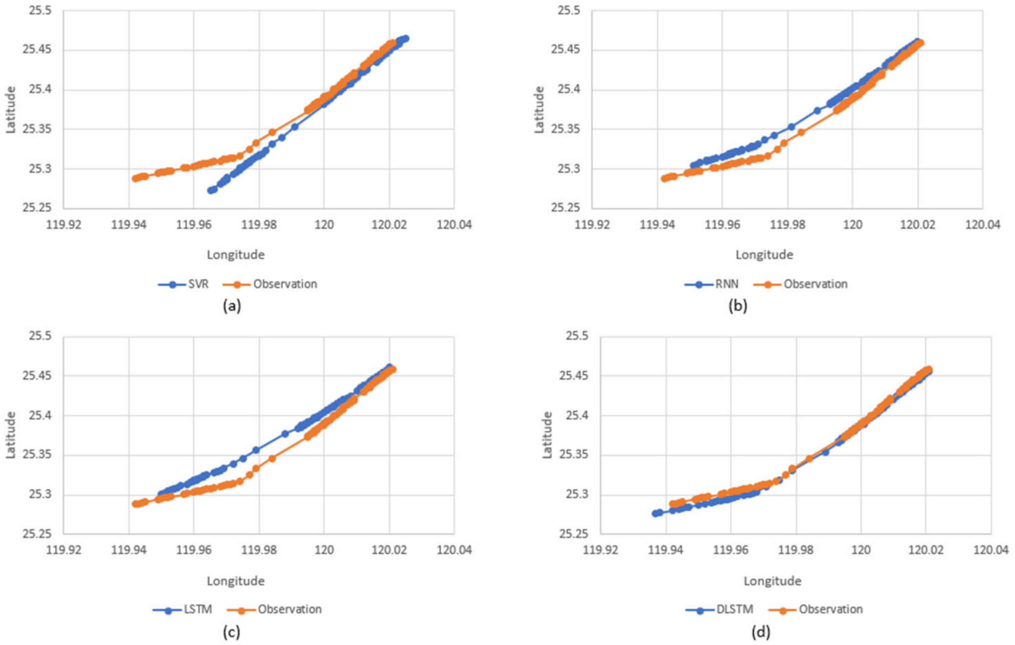
MMSI	Layer1	Layer2	Lr	Epoch	Batch	MAPE	RMSE
413697130	50		0.00001	250	128	0.03	0.04
	80					0.026	0.01
	128					0.04	0.019
	256					0.03	0.01
	80	50				0.019	0.009
		80				0.14	0.04
		128				0.18	0.06
		256				0.03	0.03

Table 4. Parameter settings of the proposed method.

Input	LSTM_1	LSTM_2	Output	Activation Function	Learning Rate	Epochs
238	128	50	1	Adam	1×10^{-5}	300

4.3. Trajectory Prediction Results

In order to clearly understand the relationship between the models and the test set, the vessel with MMSI 413484000 was used as an example for description. The prediction made by the SVR model differed substantially from the observed values (Figure 10). A possible reason for this is that the SVR model did not bend the data due to the unclear characteristics of the ship data when considering hyperparameter. The predictive accuracy of the LSTM model was higher than that of the RNN model; however, the difference in accuracy was small (Figure 10). The predictive accuracy of the proposed DLSTM model was substantially higher than that of the LSTM or RNN model, which can be attributed to its improved feature extraction.



(a)

(b)

(c)

(d)

Figure 10. Models predicted result in vessel 413484000. (**a**) SVR, (**b**) RNN, (**c**) LSTM, (**d**) DLSTM.

Table 5 shows the prediction results using all vessel tracks as the training set. Because the training data contain noise and too many different types of trajectories, the prediction results are quite bad. MAPE is between 0.12 and 0.41, RMSE is between 0.05 and 0.217, and R^2 has a negative correlation ($-0.5 \sim -46$). The results highlight the importance of data preprocessing and clustering.

In order to verify the performance of the DLSTM model, the tracks of six ships were extracted from the AIS data through a clustering process, and the accuracy metrics (RMSE, MAPE, R^2, and p) of the DLSTM, SVR, RNN, LSTM, and other models were evaluated and compared. The results shown in Table 6 revealed that the DLSTM model had the lowest prediction error for the six ships; the average MAPE value derived for the DLSTM model (0.016) was lower than those derived for the SVR, RNN, and LSTM models (0.029, 0.023, and 0.022, respectively). The RMSE value derived for the DLSTM model (0.007) was also lower than those derived for the SVR, RNN, and LSTM models (0.018, 0.010, and 0.008, respectively). Finally, the DLSTM model had the highest R^2 (0.965); the R^2 values derived for the SVR, RNN, and LSTM models were 0.86, 0.924, and 0.942, respectively. Thus, the DLSTM model had a superior performance for all indicators.

Table 5. Vessel trajectory prediction errors based on the whole set for different models on six vessels.

MMSI	Clustering	Criteria	SVR	RNN	LSTM
477147900	1	MAPE	0.14	0.3	0.2
		RMSE	0.06	0.09	0.09
		R^2	0.25	0.07	0.01
215280000	1	MAPE	0.2	0.16	0.34
		RMSE	0.09	0.1	0.15
		R^2	−0.5	−1.36	−3.5
477118100	2	MAPE	0.15	0.16	0.41
		RMSE	0.06	0.15	0.2
		R^2	0.27	−5	−9.7
413484000	2	MAPE	0.18	0.37	0.11
		RMSE	0.065	0.15	0.04
		R^2	0.5	−5	0.19
413697130	3	MAPE	0.11	0.37	0.49
		RMSE	0.07	0.14	0.217
		R^2	−0.6	−24	−46
413705580	3	MAPE	0.12	0.12	0.13
		RMSE	0.05	0.1	0.05
		R^2	−3.6	−3.9	−17.3

Table 6. Vessel trajectory prediction errors for different models for six vessels.

MMSI	Clustering	Criteria	SVR	RNN	LSTM	DLSTM
477147900	1	MAPE	0.036	0.028	0.027	**0.022**
		RMSE	0.022	0.009	0.009	**0.008**
		R^2	0.881	0.941	0.950	**0.963**
215280000	1	MAPE	0.023	0.020	0.016	**0.016**
		RMSE	0.008	0.007	0.007	**0.007**
		R^2	0.889	0.935	0.942	**0.950**
477118100	2	MAPE	0.051	0.047	0.038	**0.022**
		RMSE	0.030	0.018	0.015	**0.009**
		R^2	0.857	0.870	0.910	**0.986**
413484000	2	MAPE	0.023	0.023	0.024	**0.013**
		RMSE	0.012	0.010	0.010	**0.006**
		R^2	0.880	0.964	0.964	**0.986**
413697130	3	MAPE	0.024	0.011	0.021	**0.019**
		RMSE	0.023	0.011	0.008	**0.009**
		R^2	0.901	0.944	0.948	**0.954**
413705580	3	MAPE	0.017	0.011	0.004	**0.003**
		RMSE	0.010	0.004	0.003	**0.001**
		R^2	0.752	0.889	0.938	**0.952**
Average	-	MAPE	0.029	0.023	0.022	**0.016**
		RMSE	0.018	0.010	0.008	**0.007**
		R^2	0.860	0.924	0.942	**0.965**
p-value	-	MAPE	0.006	0.189	0.131	-
		RMSE	0.006	0.028	0.066	-

SVR, support vector regression; RNN, recurrent neural network; LSTM, long short-term memory; DLSTM, DBSCAN-based long short-term memory; Bold face, best value of each row.

The *p*-value was used to determine whether the difference in performance between the DLSTM model and the other models was significant. For the SVR model, the *p*-value was 0.06 for both MAPE and RMSE, indicating that the DLSTM model significantly outperformed the SVR model on both metrics. Similarly, for the RNN model, the *p*-value for RMSE was 0.028, indicating that the DLSTM model also significantly outperformed the RNN model on this metric. For the LSTM model, the *p*-value was >0.05 for both metrics, indicating that the difference in performance between the LSTM and DLSTM models was not significant. However, the R^2 results indicated that the DLSTM model consistently outperformed the LSTM model. For ship 215280000 in cluster 1, ship 477118100 in cluster 2, and ship 413697130 in cluster 3, the R^2 values for the DLSTM model were higher than those for the LSTM model by 0.8%, 8%, and 8%, respectively. Figure 11 presents the predicted ship tracks for each model. For 413484000 and 477118100, the R^2 value for the DLSTM model was 0.986; the predicted tracks were almost identical to the observations. The predicted tracks and indicators observed for the DLSTM model were superior to those observed for the LSTM model. Accordingly, the DLSTM model demonstrated better performance on ship trajectory prediction than the SVR, RNN, and LSTM models. Figure 12 presents a visual comparison of the metrics for each model.

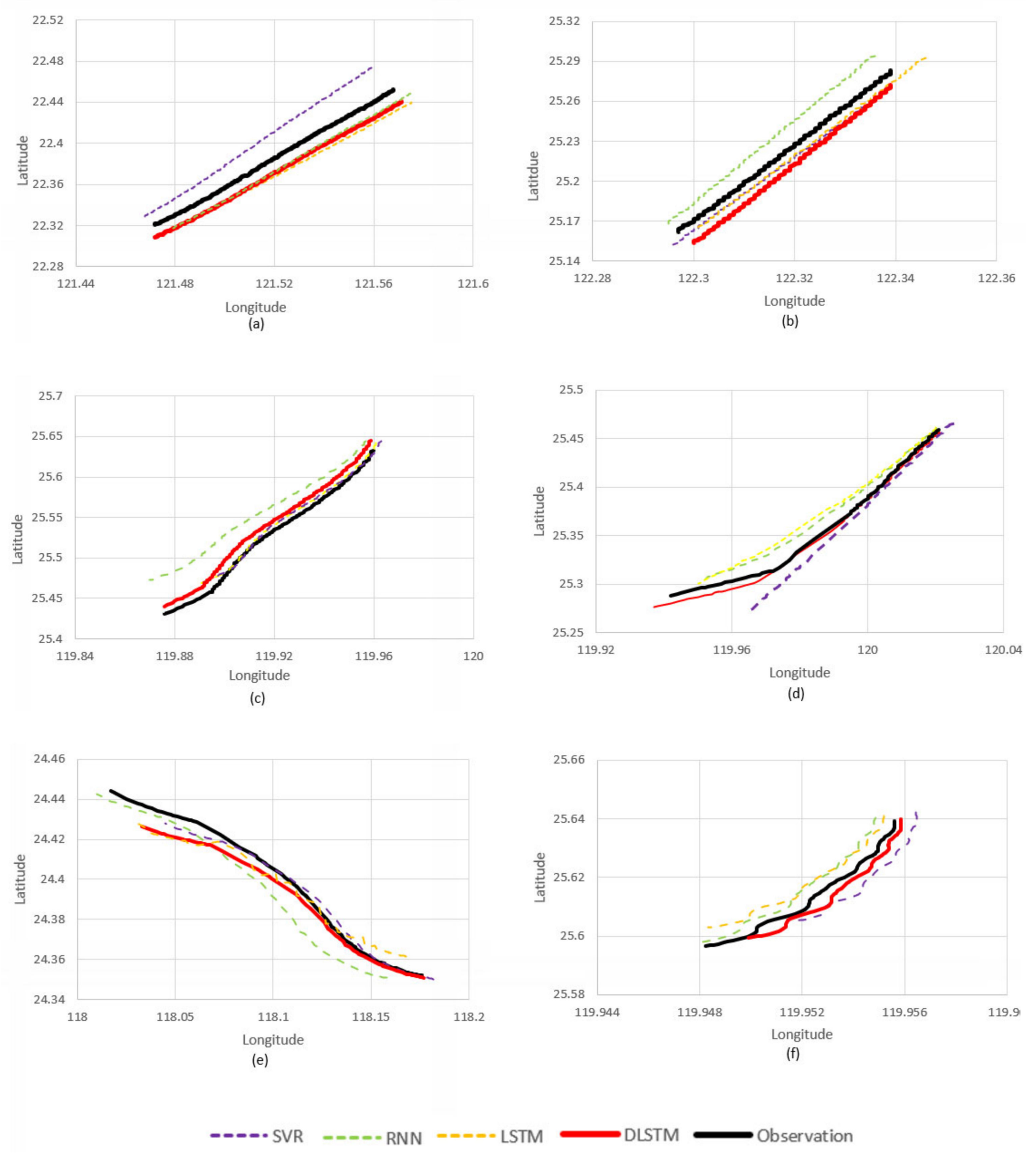


Figure 11. Vessel trajectory predicted result using 4 models. (**a**) 477147900; (**b**) 215280000; (**c**) 477118100; (**d**) 413484000; (**e**) 413697130; (**f**) 413705580.

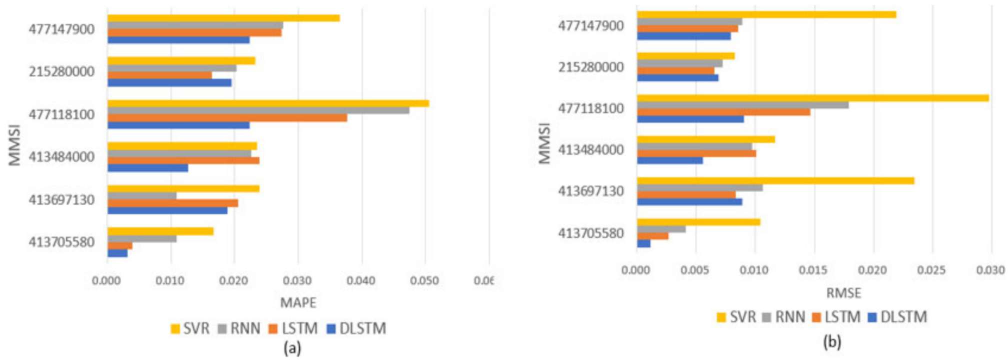


Figure 12. Predicting errors in vessel trajectories. (**a**) MAPE; (**b**) RMSE.

DLSTM shows the best model in this study, which performs better for multi-curved clusters and generally for straight-line clusters. Since DBSCAN is used to calculate the overall curve of the ship, if the clustering is mostly linear, the feature is less obvious, then the performance of DLSTM may not be particularly prominent. Adding loss function and excitation function can improve the performance of the DLSTM model. In addition, using relu as an excitation function and the normalization and adjusting other parameters can prevent overfitting. For example, for 21528000, the MAPE for both the DLSTM and LSTM models was 0.016 because the data features were overly simple, indicating that the models could not capture the features.

The RNN model was originally used to process time series data, but the superposition of previous data during input may have caused gradient explosion or gradient disappearance. However, the RNN model exhibited comparable performance to the LSTM model for simple data. This is due to the fact that the RNN model is simpler than the LSTM model. Nonetheless, the RNN model exhibited poorer performance than the LSTM model for more complex tracks. The calculated p-value for RMSE between the RNN and LSTM models was <0.05, indicating that the LSTM model significantly outperformed the RNN model. The p-values in Table 6 indicate that the performance of the LSTM model was not significantly different from that of the DLSTM model; this result is consistent with expectations.

Identifying similar features is a key task in machine learning. Therefore, this study used the ship COG to assess feature similarity. This method is conducive to eliminating abnormal tracks, such as tracks presented as radar maps, dot matrix maps, or incomplete tracks. The data remaining after DBSCAN exhibited similar characteristics, achieving this goal. The SVR model had the lowest performance among the models in terms of R^2; this result was attributed to the low tolerance of the SVR model for curved tracks. Vessel track data do not exhibit a trend and may change due to ocean conditions. Therefore, a single data point may provide reduced model detection ability for turns; this was attributed to the fact that not all tracks include turns. If the training data set is straight but the test data set is curved, the evaluation index would be lower, and the tolerance for feature values would be greater, enabling it to adapt to tracks with greater curvature. The predictions of the DLSTM model were 2% better than those of the other models, on average; thus, the DLSTM model was determined to be the best for predictions.

Ship track prediction is an important issue for navigation safety. At present, there are many prediction methods of different models, such as LSTM, RNN, and Gru. In order to enhance the accuracy of the prediction model, this study used DBSCAN to group the tracks and then carried out in-depth learning model training for the tracks in the same group. The research results showed that the track prediction using DLSTM was more accurate. Through DBSCAN, the outlier was eliminated, and the tracks with similar characteristics were combined into a data set, greatly improving the capability of capturing track features and prediction accuracy of the deep learning model.

5. Conclusions

In this study, a model combining DBSCAN and LSTM was developed for predicting vessel trajectories. The model can accurately predict trajectory sequences and be used for predictions. The data of the trajectory of six vessels were used to evaluate the model's performance. The results showed that DLSTM was 27.2% and 12.5% better than LSTM in MAPE and RMSE, respectively. The proposed model outperformed the SVR, RNN, and LSTM models, demonstrating its suitability for vessel trajectory prediction tasks. Furthermore, the proposed model can extract significant features and apply them to reduce errors in vessel track point of prediction tasks. Accordingly, the study findings can help maritime traffic controllers predict the accurate trajectories of vessels, enable them to take preventive measures, avoid collisions, and improve the efficiency and safety of maritime traffic.

Author Contributions: C.-H.Y.: conceptualization, project administration. G.-C.L., C.-H.W. and Y.-H.L.: methodology, writing—review and editing. C.-H.W., Y.-C.W. and K.-C.C.: conceptualization, data analyzing. All authors have read and agreed to the published version of the manuscript.

Funding: The funding sources are the Ministry of Science and Technology, Taiwan (under Grant no. 111-2221-E-165 -001 -MY3).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data were obtained from the U.S. Coast Guard through an onboard navigation safety device on 14 March 2022. (https://marinecadastre.gov/ais/).

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

References

1. Tu, E.; Zhang, G.; Rachmawati, L.; Rajabally, E.; Huang, G.-B. Exploiting AIS data for intelligent maritime navigation: A comprehensive survey from data to methodology. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 1559–1582. [CrossRef]
2. Sirimanne, S.N.; Hoffman, J.; Juan, W.; Asariotis, R.; Assaf, M.; Ayala, G.; Benamara, H.; Chantrel, D.; Hoffmann, J.; Premti, A. Review of maritime transport 2019. In Proceedings of the United Nations Conference on Trade and Development, Geneva, Switzerland, 30 October 2019.
3. Harati-Mokhtari, A.; Wall, A.; Brooks, P.; Wang, J. Automatic Identification System (AIS): Data reliability and human error implications. *J. Navig.* **2007**, *60*, 373–389. [CrossRef]
4. Norris, A. AIS implementation–Success or failure? *J. Navig.* **2007**, *60*, 1–10. [CrossRef]
5. Sanchez-Gonzalez, P.-L.; Díaz-Gutiérrez, D.; Leo, T.J.; Núñez-Rivas, L.R. Toward digitalization of maritime transport? *Sensors* **2019**, *19*, 926. [CrossRef] [PubMed]
6. Jurdana, I.; Krylov, A.; Yamnenko, J. Use of artificial intelligence as a problem solution for maritime transport. *J. Mar. Sci. Eng.* **2020**, *8*, 201. [CrossRef]
7. Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; Savarese, S. Social lstm: Human trajectory prediction in crowded spaces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27 June 2016; pp. 961–971.
8. Park, S.H.; Kim, B.; Kang, C.M.; Chung, C.C.; Choi, J.W. Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1672–1678.
9. Xiaopeng, T.; Xu, C.; Lingzhi, S.; Zhe, M.; Qing, W. Vessel trajectory prediction in curving channel of inland river. In Proceedings of the 2015 International Conference on Transportation Information and Safety, Wuhan, China, 25–28 June 2015; pp. 706–714.
10. Wang, L.; Zhang, L.; Yi, Z. Trajectory predictor by using recurrent neural networks in visual tracking. *IEEE Trans. Cybern.* **2017**, *47*, 3172–3183. [CrossRef]
11. 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]
12. Zhang, D.; Ren, Q.; Su, D. A Novel Authentication Methodology to Detect Counterfeit PCB Using PCB Trace-Based Ring Oscillator. *IEEE Access* **2021**, *9*, 28525–28539. [CrossRef]
13. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
14. Ryu, J.; Kamata, S.-I. An efficient computational algorithm for Hausdorff distance based on points-ruling-out and systematic random sampling. *Pattern Recognit.* **2021**, *114*, 107857. [CrossRef]
15. Du, J.; Wu, F.; Yin, J.; Liu, C.; Gong, X. Polyline simplification based on the artificial neural network with constraints of generalization knowledge. *Cartogr. Geogr. Inf. Sci.* **2022**, *49*, 313–337. [CrossRef]

16. Deng, D. DBSCAN clustering algorithm based on density. In Proceedings of the 2020 7th International Forum on Electrical Engineering and Automation (IFEEA), Hefei, China, 25–27 September 2020; pp. 949–953.
17. Drucker, H.; Burges, C.J.; Kaufman, L.; Smola, A.; Vapnik, V. Support vector regression machines. *Adv. Neural Inf. Processing Syst.* **1996**, *9*, 155–161.
18. Nguyen, D.-D.; Le Van, C.; Ali, M.I. Vessel trajectory prediction using sequence-to-sequence models over spatial grid. In Proceedings of the 12th ACM International Conference on Distributed and Event-Based Systems, Hamilton, New Zealand, 25 June 2018; pp. 258–261.
19. Borkowski, P. The Ship Movement Trajectory Prediction Algorithm Using Navigational Data Fusion. *Sensors* **2017**, *17*, 1432. [CrossRef] [PubMed]
20. Meng, S.; Wang, H.; Lu, W.; Wang, Z.; Li, Y. Drift trajectory model of the unpowered vessel on the sea and its application in the drift simulation of the Sanchi oil tanker. *Oceanol. Limnol. Sin.* **2018**, *49*, 242–250.
21. Murray, B.; Perera, L.P. A data-driven approach to vessel trajectory prediction for safe autonomous ship operations. In Proceedings of the 2018 13th International Conference on Digital Information Management, Berlin, Germany, 24–26 September 2018; pp. 240–247.
22. Yan, Z. Traj-ARIMA: A spatial-time series model for network-constrained trajectory. In Proceedings of the Third International Workshop on Computational Transportation Science, San Jose, CA, USA, 2 November 2010; pp. 11–16.
23. Qi, L.; Zheng, Z. Trajectory prediction of vessels based on data mining and machine learning. *J. Digit. Inf. Manag.* **2016**, *14*, 33–40.
24. Zhang, L.; Meng, Q. Probabilistic ship domain with applications to ship collision risk assessment. *Ocean Eng.* **2019**, *186*, 106130. [CrossRef]
25. Wang, C.; Ren, H.; Li, H. Vessel trajectory prediction based on AIS data and bidirectional GRU. In Proceedings of the 2020 International Conference on Computer Vision, Image and Deep Learning, Chongqing, China, 10–12 July 2020; pp. 260–264.
26. Huang, C.; Wang, J.; Chen, X.; Cao, J. Bifurcations in a fractional-order BAM neural network with four different delays. *Neural Netw.* **2021**, *141*, 344–354. [CrossRef] [PubMed]
27. Yang, W.; Yu, W.; Cao, J.; Alsaadi, F.E.; Hayat, T. Almost automorphic solution for neutral type high-order Hopfield BAM neural networks with time-varying leakage delays on time scales. *Neurocomputing* **2017**, *267*, 241–260. [CrossRef]
28. Xu, C.; Zhang, W.; Aouiti, C.; Liu, Z.; Yao, L. Further analysis on dynamical properties of fractional-order bi-directional associative memory neural networks involving double delays. *Math. Methods Appl. Sci.* **2022**. [CrossRef]
29. Xu, Z.; Zeng, W.; Chen, L.; Chu, X. Aircraft Trajectory Prediction Using Social LSTM Neural Network. In Proceedings of the CICTP 2021, Xi'an, China, 16–19 December 2021; pp. 87–97.
30. Semerdjiev, E.; Mihaylova, L. Variable-and fixed-structure augmented interacting multiple model algorithms for manoeuvring ship tracking based on new ship models. *Int. J. Appl. Math. Comput. Sci.* **2000**, *10*, 591–604.
31. Tusell, F. Kalman filtering in R. *J. Stat. Softw.* **2011**, *39*, 1–27. [CrossRef]
32. De Vries, G.K.D.; Van Someren, M. Machine learning for vessel trajectories using compression, alignments and domain knowledge. *Expert Syst. Appl.* **2012**, *39*, 13426–13439. [CrossRef]
33. Kim, J.-S. Vessel target prediction method and dead reckoning position based on SVR seaway model. *Int. J. Fuzzy Logic Intell. Syst.* **2017**, *17*, 279–288. [CrossRef]
34. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
35. Ribeiro, A.H.; Tiels, K.; Aguirre, L.A.; Schön, T. Beyond exploding and vanishing gradients: Analysing RNN training using attractors and smoothness. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Online, 26–28 August 2020; pp. 2370–2380.
36. Zhou, C.; Sun, C.; Liu, Z.; Lau, F. A C-LSTM neural network for text classification. *arXiv* **2015**, arXiv:1511.08630.
37. Chen, K.; Zhou, Y.; Dai, F. A LSTM-based method for stock returns prediction: A case study of China stock market. In Proceedings of the 2015 IEEE International Conference on Big Data (Big Data), Santa Clara, CA, USA, 29 October–1 November 2015; pp. 2823–2824.
38. Yang, C.-H.; Wu, C.-H.; Shao, J.-C.; Wang, Y.-C.; Hsieh, C.-M. AIS-Based Intelligent Vessel Trajectory Prediction Using Bi-LSTM. *IEEE Access* **2022**, *10*, 24302–24315. [CrossRef]
39. Kumar, K.M.; Reddy, A.R.M. A fast DBSCAN clustering algorithm by accelerating neighbor searching using Groups method. *Pattern Recognit.* **2016**, *58*, 39–48. [CrossRef]
40. Tran, T.N.; Drab, K.; Daszykowski, M. Revised DBSCAN algorithm to cluster data with dense adjacent clusters. *Chemom. Intell. Lab. Syst.* **2013**, *120*, 92–96. [CrossRef]
41. Vapnik, V.; Golowich, S.E.; Smola, A.J. Support vector method for function approximation, regression estimation and signal processing. In Proceedings of the Advances in Neural Information Processing Systems, Denver, DC, USA, 2–5 December 1997; pp. 281–287.
42. Ceperic, E.; Ceperic, V.; Baric, A. A strategy for short-term load forecasting by support vector regression machines. *IEEE Trans. Power Syst.* **2013**, *28*, 4356–4364. [CrossRef]
43. Mehr, A.D.; Nourani, V.; Khosrowshahi, V.K.; Ghorbani, M.A. A hybrid support vector regression—Firefly model for monthly rainfall forecasting. *Int. J. Environ. Sci. Technol.* **2019**, *16*, 335–346. [CrossRef]
44. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [CrossRef]

45. Pascanu, R.; Mikolov, T.; Bengio, Y. On the difficulty of training recurrent neural networks. In Proceedings of the International Conference on Machine Learning, Atlanta GA USA, 16 June 2013; pp. 1310–1318.
46. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
47. Kratzert, F.; Klotz, D.; Brenner, C.; Schulz, K.; Herrnegger, M. Rainfall–runoff modelling using long short-term memory (LSTM) networks. *Hydrol. Earth Syst. Sci.* **2018**, *22*, 6005–6022. [CrossRef]