

## Article

# Ship Trajectory Prediction Based on Bi-LSTM Using Spectral-Clustered AIS Data

Jinwan Park <sup>1</sup>, Jungsik Jeong <sup>2</sup> and Youngsoo Park <sup>3,\*</sup>

<sup>1</sup> Department of Maritime Transportation System, Mokpo National Maritime University, Mokpo 58628, Korea; pjinwan2@gmail.com

<sup>2</sup> Division of Maritime Transportation Science, Mokpo National Maritime University, Mokpo 58628, Korea; jsjeong@mamu.ac.kr

<sup>3</sup> Division of Navigation Convergence Studies, Korea Maritime & Ocean University, Busan 49112, Korea

\* Correspondence: youngsoo@kmou.ac.kr; Tel.: +82-51-410-5085

**Abstract:** According to the statistics of maritime accidents, most collision accidents have been caused by human factors. In an encounter situation, the prediction of ship's trajectory is a good way to notice the intention of the other ship. This paper proposes a methodology for predicting the ship's trajectory that can be used for an intelligent collision avoidance algorithm at sea. To improve the prediction performance, the density-based spatial clustering of applications with noise (DBSCAN) has been used to recognize the pattern of the ship trajectory. Since the DBSCAN is a clustering algorithm based on the density of data points, it has limitations in clustering the trajectories with nonlinear curves. Thus, we applied the spectral clustering method that can reflect a similarity between individual trajectories. The similarity measured by the longest common subsequence (LCSS) distance. Based on the clustering results, the prediction model of ship trajectory was developed using the bidirectional long short-term memory (Bi-LSTM). Moreover, the performance of the proposed model was compared with that of the long short-term memory (LSTM) model and the gated recurrent unit (GRU) model. The input data was obtained by preprocessing techniques such as filtering, grouping, and interpolation of the automatic identification system (AIS) data. As a result of the experiment, the prediction accuracy of Bi-LSTM was found to be the highest compared to that of LSTM and GRU.

**Keywords:** ship trajectory prediction; intelligent collision avoidance; maritime accidents; spectral clustering; Bi-LSTM; GRU



**Citation:** Park, J.; Jeong, J.; Park, Y. Ship Trajectory Prediction Based on Bi-LSTM Using Spectral-Clustered AIS Data. *J. Mar. Sci. Eng.* **2021**, *9*, 1037. <https://doi.org/10.3390/jmse9091037>

Academic Editors: Marco Cococcioni and Kostas Belibassakis

Received: 11 August 2021

Accepted: 17 September 2021

Published: 21 September 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

According to statistics compiled by Korea Maritime Safety Tribunal (KMST), a total of 13,687 marine accidents occurred over the last five years (2016–2020), and among them, collision accidents between ships account for 45% of the total, which account for the largest proportion [1]. Since 2016, the number of collision accidents has increased by an average of 4% annually. As a result of the analysis on the causes of collision accidents in the last 5 years, 95% of all collision accidents were caused by human factors, and 70% of them were known as negligence on look-out [1]. This situation is due to the erroneous judgment of the risk of collision despite the provision of the navigation information from the radio detection and ranging (RADAR), electric chart display and information system (ECDIS), and the automatic identification system (AIS). When two ships are meeting, predicting the future path of own ship and a target ship is important to make the right decision and avoid collision. McLane and Wolf proved through their experiments that the trajectory prediction helps to prevent the collisions [2].

Inoue et al. and Fossen predicted the ship trajectory using a ship hydrodynamic model that can calculate the ship motion based on the actual ship data, such as the principal particulars of ship hull, rudder, propulsion system, ocean current velocity, and wind

force [3,4]. Passenier presented a track predictor for ships based on a relatively simple mathematical model that can adapt to continuously changing navigation conditions and applied the extended-Kalman filtering technique as a method for online identification and adaptation to the disturbances and the parameters of the prediction model [5]. Czapiewska and Sadowski compared the performance of the Kalman filtering technique and the linear algorithm as a method of predicting the trajectory of a ship [6]. The linear algorithm is a very simple extrapolation method that calculates the future position based on the observed information, on the assumption that the state information such as the speed, course, and position of the vessel observed in the most recent time period will remain constant in the future [7–10]. Breda and Passenier compared the three different path predictors through the ship simulation experiments [11]. The first of those was based on the hydrodynamic model of the ship; the second predictor was based on a speed and rate of turn extrapolator; the third predictor was based on a linear speed and course extrapolator. Laxhammar predicted the patterns of the ship traffic for anomaly detection using the unsupervised clustering method. The gaussian mixture model (GMM) is used as cluster model and the expectation maximization (EM) algorithm is used as clustering algorithm [12]. Ristic et al. extracted ship motion patterns from the historic AIS data of the confined ports and waterways using statistical analysis, which were then used for a motion anomaly detection through the kernel density estimation (KDE) and particle filter [13]. Aarsæther and Moan also estimated the patterns of the ship trajectory based on the computer vision technique using AIS data [14]. The navigation patterns that were recognized by their work can be applied for providing navigation plans in a confined area. Tang et al. showed the ship trajectory obtained from the AIS data on a grid plane and predicted the future path of the ship based on the probabilistic directed graph model and the extrapolation method [15]. The neural network, which is an artificial intelligence model in the form of a neuron in a biological brain, was also used to predict the trajectory of the ship. Łacki presented the intelligent ship maneuvering prediction system using the neuroevolution and the evolutionary algorithms [16]. Xu et al. and Zhou et al. adopted the back-propagation neural network algorithm to predict the ship trajectory [17,18]. The trajectory prediction based on such a neural network can derive relatively accurate results through the learning process of the observed data of parameters of the ship navigation without applying the hydrodynamic model of the ship or collecting the accurate disturbance data at sea. Zhao and Shi adopted the density-based spatial clustering of applications with noise (DBSCAN) to cluster the ship trajectory and predicted the ship trajectory based on the long short-term memory (LSTM), which is one of the recurrent neural network (RNN) models [19]. Since the ship trajectory data are time series data, it is necessary to consider not only the trajectory data of the current time step but also previously observed trajectory data to predict future trajectory data. A representative neural network that can predict future data by applying time series data is the RNN [20]. However, since a back-propagation algorithm is basically used for training RNN, when error information is propagated back through time, the vanishing gradient problem generally occurs. Models that can manage this problem are the LSTM [21] and the gated recurrent unit (GRU) [22]. In the field of aviation, Shi et al. also adopted the LSTM to predict aircraft trajectory [23]. However, Graves and Schmidhuber and Siami-Namini et al. proved that the bidirectional LSTM (Bi-LSTM) that is an extended model of the LSTM provides better performance than the LSTM [24,25]. Therefore, it is necessary to apply the Bi-LSTM in this study to increase the accuracy of the prediction model. As a part of the methodology applied by Zhao and Shi, the DBSCAN is a density-based clustering algorithm that is frequently used for trajectory pattern recognition [26]. However, since the trajectory of a ship is a nonlinear curve composed of several position data, the clustering algorithm based on the density of the data point has limitations in clustering the ship trajectory. To solve this problem, it is necessary to apply the spectral clustering that can reflect the similarity between individual trajectories on the result.

The purpose of this study is to propose the methodology of ship trajectory prediction that can be used for the intelligent ship collision avoidance algorithm. Based on the results

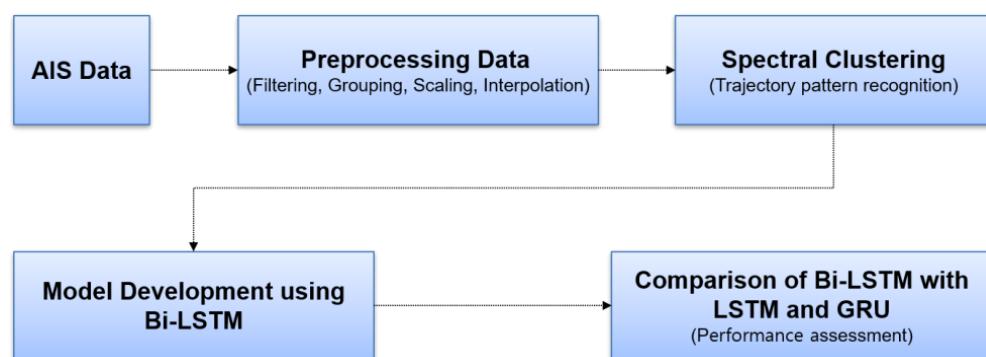
of trajectory pattern recognition through the spectral clustering, we developed the models to predict the ship's trajectory by applying the Bi-LSTM, and its performance was compared with the LSTM and the GRU. For the experiment, the AIS data collected in confined coastal waters were used as input data.

The remainder of this article is organized as follows: Section 2 proposes the method for predicting the ship trajectory using the spectral clustering and the extended RNNs. Section 3 shows the results of the experiment using AIS data. Section 4 present the conclusions that include the limitations and further works of this study.

## 2. Methodology

Machine learning is the methods that make it possible to identify the properties and patterns of a defined problem through the data training. Because the navigation of a ship is fraught with too many uncertainties, in order to deal with this uncertainty problem such as the trajectory prediction, a proper way is recognizing patterns through the training of observed data using a machine learning method rather than a method for deriving results through theoretical correlations between the various factors. Therefore, the ship trajectories were predicted by unsupervised and supervised learning tools using the historic trajectory data that were extracted from the AIS data in the confined area. The AIS data were preprocessed using several methods such as filtering, grouping, scaling, and interpolation. In order to accurately predict the trajectories, it is necessary to first analyze the patterns of them in the target area because that area has a dense pattern of trajectories generated by many ships. As mentioned earlier, the DBSCAN [27] used by Zhao and Shi [19] to recognize the pattern of the trajectory is a clustering algorithm based on the density of data points, so there is a limit to clustering the trajectories that can be curved, straight, or wavy. This is because the similarity between the tracks was not considered. Therefore, we clustered the trajectories by applying the spectral clustering technique that can consider the similarity between the trajectories. The similarity between trajectories measured based on the longest common subsequence (LCSS) [28] in consideration of the characteristics of the ship trajectories having different shapes and lengths.

After the trajectory clustering procedure, we prepared the trajectory data with the same pattern for using as input data into the regression models of neural networks, the supervised learning tools, to predict the future trajectory. The trajectory data are time series data composed of data points indexed in time sequence. Since the values of previous time step affect the current value, it is reasonable to apply the RNN that can reflect past information on the outputs of current information. We can obtain the trajectory data of future time step for each time step by training the RNN. However, since the gradient of the error function tends to vanish quickly in the general RNN training, the Bi-LSTM was applied to alleviate this phenomenon. In order to assess the performance of the developed model through the Bi-LSTM, it was compared with the other models applied by the LSTM and the GRU using the value of the root mean square error (RMSE). Figure 1 generally shows the process diagrams of the methodology mentioned above.



**Figure 1.** The proposed method for ship trajectory prediction.

### 2.1. Preprocessing AIS Data

The AIS data were used to develop the model for predicting the ship trajectory. Before being used as input data, they require appropriate preprocessing to ensure reliability, accuracy, and availability of them. First, we extracted necessary information, which consisted of the following variables: the maritime mobile service identity (MMSI) number, time stamps, latitude, longitude, course, and speed, from the AIS data of ships in the confined coastal waters. Using the MMSI number, the AIS data were grouped by ship to perform the trajectory interpolation. According to the ship's AIS performance standards required by the international maritime organization (IMO), the static information should be updated every 6 min or on demand, and the dynamic information should be reported at intervals of at least 2 s and at most 3 min depending on the speed of the ship [29]. However, since most of the ship dynamic information among the collected AIS data was not updated according to the performance standards, the procedure for interpolating the missing ship dynamic information was required. The observed trajectory data that consisted of the latitude, the longitude, the course, and the speed of the target vessel were interpolated at 1 s intervals by applying a cubic spline function. The performance of this interpolation method had proven in [30,31]. Then, the data reduction was performed in the light of moving speed of the ships, so that the intervals of the time stamps were changed from 1 s to 2 min. Therefore, if the interval of data points exceeds 2 min, we considered that they are different from each other and separated them at that position. Based on this method, the AIS data were grouped by trajectory as follows:

$$Tr_n = \{Lat_t, Long_t, Co_t, V_t\} \quad (1)$$

where  $Tr$  denotes individual trajectory dataset,  $n$  denotes the index of the trajectory dataset,  $Lat$  denotes the latitude,  $Long$  denotes the longitude,  $Co$  denotes the ship course over ground,  $V$  denotes the ship speed over ground, and  $t$  denotes the time stamps. Lastly, the scaling process was conducted on the input data of the Bi-LSTM, the LSTM, and the GRU. Each variable was scaled to the distribution centered on 0, with a standard deviation of 1.

### 2.2. Application of Spectral Clustering

Since the trajectories have various patterns and lengths in the confined coastal waters, it would be difficult to expect a result with high prediction accuracy if the trajectory data have been applied to the neural network model as they stand [32]. To increase the accuracy of the trajectory prediction, trajectory clustering is necessary to recognize their pattern. Through the process, we can identify various trajectory groupings that have different patterns. The trajectories in the same group are more similar to each other than to those in other groups. There is the dynamic time warping (DTW) and the longest common subsequence (LCSS) methods that can measure the similarity between two trajectories regardless of their length. As a result of comparing the performance of the two methods, it is known that the LCSS is superior to the DTW in the noise and outliers [33]. Therefore, the LCSS was adopted in this study, and the LCSS distances between each pair of trajectories were calculated according to [28].

Let  $\{a_1, a_2, \dots, a_l\}$  and  $\{b_1, b_2, \dots, b_m\}$  be the data sets of trajectories  $A$  and  $B$ , respectively. The LCSS distance between  $A$  and  $B$ ,  $D_{LCSS}(A, B)$  is obtained by

$$D_{LCSS}(A, B) = 1 - \frac{L(A, B)}{\min(l, m)} \quad (2)$$

where

$$L(A, B) = \begin{cases} 0, & n = 0 \text{ or } m = 0 \\ 1 + L(H(A), H(B)), & d_E(a_l, b_l) \leq \epsilon \text{ and } |l - m| \leq \delta \\ \max(L(H(A), B), L(A, H(B))), & \text{otherwise} \end{cases} \quad (3)$$

In Equation (3),  $H(A) = \{a_1, a_2, \dots, a_{l-1}\}$ ,  $H(A) = \{b_1, b_2, \dots, b_{m-1}\}$ , and  $d_E$  denotes the Euclidean distance. The constant  $\epsilon$  denotes the matching threshold, and constant  $\delta$  controls the range of time intervals for matching two trajectories. In this study,  $\epsilon$  and  $\delta$  were set to 1 and 60 respectively, through iterative experiments. Based on the LCSS distances through pairwise calculation between the trajectories, a similarity matrix  $S$  can be obtained by

$$(S_{ij})_{i,j=1,\dots,n} = \exp\left(-\left(D_{LCSS}^2(Tr_i, Tr_j)/2\sigma^2\right)\right), \quad (4)$$

where  $\sigma$  represent a kernel width that affects the performance of the spectral clustering. We estimated the best  $\sigma$  as 0.93 through iterative experiments. Each trajectory has a similarity of 1 to itself, and as the LCSS distance increases, the value of similarity decreases and gets closer to 0. This measured similarity was used to calculate a graph Laplacian matrix in the process of the spectral clustering. The spectral clustering is a method of grouping data into arbitrary clusters using eigenvectors of the graph Laplacian matrix that contains graph information [34,35]. The graph Laplacian matrix  $L$  can be obtained by

$$L = D - S \quad (5)$$

where  $D$  denotes a degree matrix that can be obtained as follows:

$$D = \text{diag}\left(\sum_{j=1}^n S_{1,j}, \dots, \sum_{i=1}^n S_{n,i}\right) \quad (6)$$

However, in most cases, a normalized Laplacian matrix is used to improve the performance of the spectral clustering. There are two types of the normalized graph Laplacian matrix: the random-walk Laplacian matrix,  $L_{rw}$  [34] and the normalized symmetric Laplacian matrix,  $L_{sym}$  [36]. They can be calculated as follows:

$$L_{rw} = D^{-1}L = I - D^{-1}S \quad (7)$$

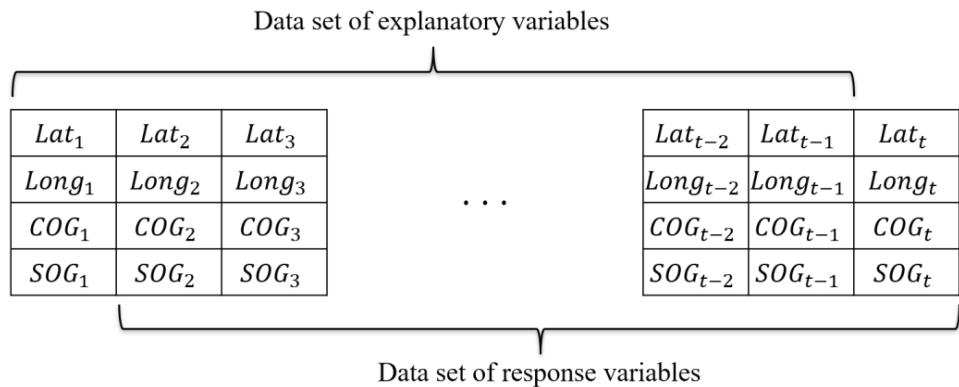
$$L_{sym} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}SD^{-1/2} \quad (8)$$

In the spectral clustering method, it is recommended to use  $L_{rw}$  rather than  $L_{sym}$  as the normalized graph Laplacian matrix [35]. Therefore, in this study,  $L_{rw}$  was used to construct the graph Laplacian matrix. Based on the obtained graph Laplacian matrix, clustering of trajectory was performed using the  $k$ -means [37] clustering algorithm. As the  $k$ -means process, the first step is to choose  $k$  data points from all data and set each as the center of the initial cluster. The second step is to calculate the distance between each data point and each cluster centroid. The third step if to assign each data to the closest cluster. The fourth step is to calculate the average of all assigned data in each cluster to obtain  $k$  new cluster center. It repeats the second step through fourth step until there is not a change in the cluster compared to the previous state. To perform the  $k$ -means clustering, it is necessary to estimate the value of  $k$  denotes the number of clusters in the data. We used a similarity graph to estimate the number of clusters according to [36] in this study. The similarity graph can be created from the similarity matrix, and we can find the number of connected components in the graph as the value of  $k$ . Through the iterative process, we limited the similarity values to 0.7 and estimated the value of  $k$  to 29.

### 2.3. Application of Recurrent Neural Networks

Based on the results of the spectral clustering, the trajectory data with the same pattern were prepared for the model development. Considering that the trajectory data corresponds to the time series data indexed in time order, the RNNs such as Bi-LSTM, LSTM, and GRU were applied to the model development. To predict the value of future time step of each the trajectories, as represented in Equation (1), the explanatory and response variables were equally specified as the latitude (*Lat*) and the longitude (*Long*)

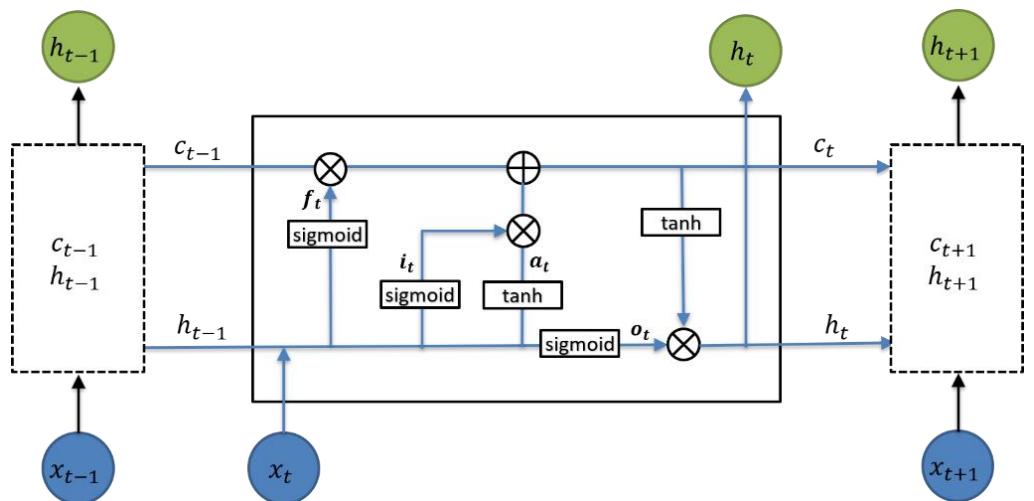
of the ship position, the ship course over ground (COG), and the ship speed over ground (SOG). The trajectory data of response variables were shifted forward by one time step from that of explanatory variables, which was illustrated in Figure 2. Thus, the RNNs learns the prepared data to predict the value of the next time step. In the prediction step, we updated the RNNs state with the observed value of time steps between predictions in consideration of the actual navigation environment in which the input can be obtained in real time through navigation equipment such as AIS or RADAR.



**Figure 2.** Structure of trajectory data for training of RNNs.

### 2.3.1. LSTM

The LSTM was proposed by Hochreiter and Schmidhuber in 1997 [21]. They managed the vanishing gradient problem using the LSTM that has the memory cells, input, and output gates. In 1999, Ger et al. improved the initial LSTM by introducing a forget gate that enables the LSTM to learn to reset itself [38]. The modified LSTM model was shown in Figure 3 [21].



**Figure 3.** LSTM model.

In Figure 3 [38],  $x_t$  and  $h_t$  are the input and the hidden state at time  $t$ , respectively. Unlike the general RNN model, it has a cell state  $c_t$ , and the three gates that are the forget gate  $f_t$ , the input gate  $i_t$ , and the output gate  $o_t$  in the hidden state. The forget gate  $f_t$  decides how many rates should be maintained from the value of the previous cell state  $c_{t-1}$  at the time  $t$ . It can be obtained as follows [21]:

$$f_t = \sigma(U_f x_t + W_f h_{t-1} + b_f) \quad (9)$$

where  $\sigma$  is the sigmoid function,  $U_f$  and  $W_f$  are the weight values, and  $b_f$  is the bias value. The sigmoid function is the most commonly used as an activation function of the neural network along with the hyperbolic tangent function. Since they are differentiable function, the optimization algorithms such as the gradient descent can be adopted as a learning method in the neural networks. In the LSTM and GRU models, the sigmoid function is applied to each gate as follows:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (10)$$

The hyperbolic tangent function  $\tanh$  is applied to update the cell or hidden state in the LSTM and GRU model as follows:

$$\tanh(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (11)$$

The input gate  $i_t$  at time  $t$  decides how much of the processing result of the input  $x_t$  should be reflected in the cell state  $c_t$ . It can be obtained as follows [21]:

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \quad (12)$$

where  $U_i$  and  $W_i$  are the weight values, and  $b_i$  is the bias value.

The output gate  $o_t$  at time  $t$  adjusts the output of the value stored in the cell state  $c_t$ , and it can be obtained as follows [21]:

$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o) \quad (13)$$

where  $U_o$  and  $W_o$  are the weight values, and  $b_o$  is the bias value.

The cell state  $c_t$  at time  $t$  can be obtained as follows [21]:

$$c_t = i_t \circ a_t + f_t \circ c_{t-1} \quad (14)$$

where  $a_t$  and  $\circ$  are the new cell state at time  $t$  and an element-wise product, respectively.  $a_t$  can be obtained as follows: [21]:

$$a_t = \tanh(U_c x_t + W_c h_{t-1} + b_c) \quad (15)$$

where  $U_c$  and  $W_c$  are the weight values, and  $b_c$  is the bias value. Finally, the hidden state  $h_t$  can be obtained as follows [21]:

$$h_t = o_t \circ \tanh(c_t) \quad (16)$$

### 2.3.2. Bi-LSTM

Bi-LSTM is a neural network using the LSTM model for each hidden node of a bidirectional RNN [39], in which the output value is affected by the values of the input and hidden state at both the previous and later time as the hidden layer of that is separated in forward and backward directions [24]. Figure 4 [39] shows the structure of this Bi-LSTM model, and the output  $y$  is calculated based on the hidden states of both the forward layer and backward layer. At time  $t$ , the forward hidden state  $\overset{\rightarrow}{h}_t$  can be obtained as follows [39]:

$$\overset{\rightarrow}{h}_t = \sigma(U_{\rightarrow} x_t + W_{\rightarrow} \overset{\rightarrow}{h}_{t-1} + b_{\rightarrow}), \quad (17)$$

where  $U_{\rightarrow}$  and  $W_{\rightarrow}$  are the weight values, and  $b_{\rightarrow}$  is the bias value.

The backward hidden state  $\overset{\leftarrow}{h}_t$  at time  $t$  can be obtained as follows [39]:

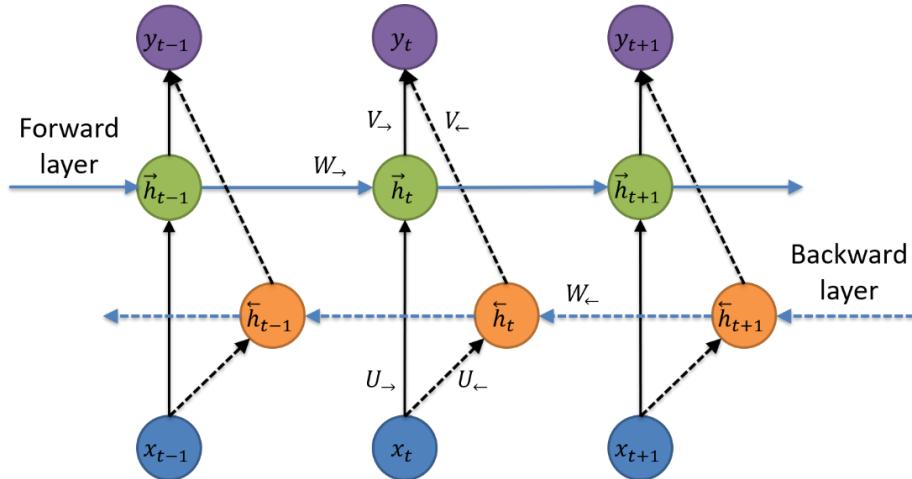
$$\overset{\leftarrow}{h}_t = \sigma(U_{\leftarrow} x_t + W_{\leftarrow} \overset{\leftarrow}{h}_{t+1} + b_{\leftarrow}), \quad (18)$$

where  $U_{\leftarrow}$  and  $W_{\leftarrow}$  are the weight values, and  $b_{\leftarrow}$  is the bias value.

The output  $y_t$  at time  $t$  can be obtained as follows [39]:

$$y_t = V_{\rightarrow} \vec{h}_t + V_{\leftarrow} \overset{\leftarrow}{h}_t + b_o, \quad (19)$$

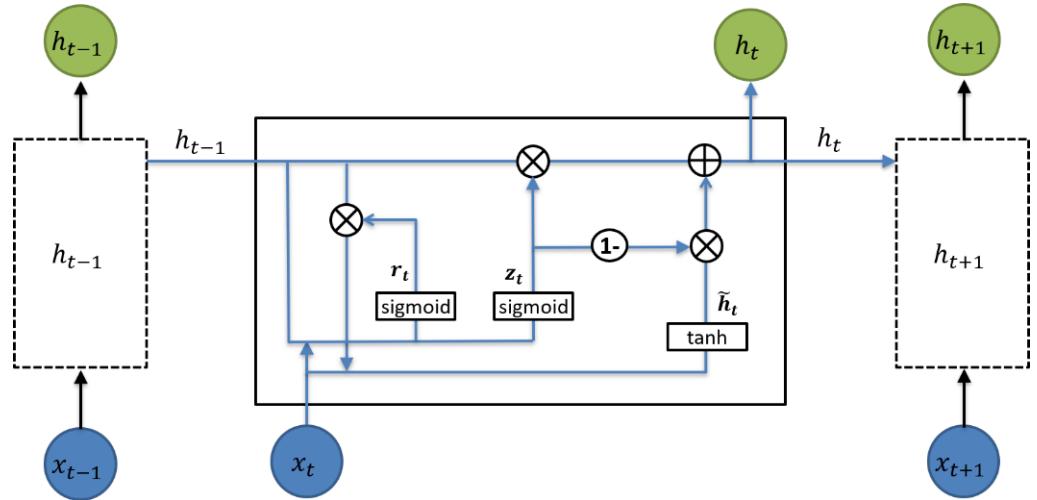
where  $V_{\rightarrow}$  and  $V_{\leftarrow}$  are the weight values, and  $b_o$  is the bias value.



**Figure 4.** Bi-RNN model.

### 2.3.3. GRU

The GRU is a neural network model proposed in [22] (2014), compared with the LSTM, the internal operation is simple. The LSTM consists of the three gates, whereas the GRU consists of a reset gate  $r_t$  and an update gate  $z_t$  as shown in Figure 5 [22].



**Figure 5.** GRU model.

In Figure 5,  $x_t$  denotes the input to the hidden state at time  $t$ , and  $h_t$  denotes the output of the hidden state at time  $t$ . The reset gate  $r_t$  decides how to combine the input  $x_t$  and the previous hidden state  $h_{t-1}$ , and it is computed by [22]:

$$r_t = \sigma(U_r x_t + W_r h_{t-1} + b_r) \quad (20)$$

The new hidden state  $\tilde{h}_t$  at time  $t$  is computed by [22]:

$$\tilde{h}_t = \tanh(U_h x_t + W_h (r_t \circ h_{t-1})) \quad (21)$$

The update gate  $z_t$  decides how much rate of the previous hidden state  $h_{t-1}$  is to be updated with the new hidden state  $\tilde{h}_t$ . Based on the  $z_t$ , the hidden state  $h_t$  can be obtained as follows [22]:

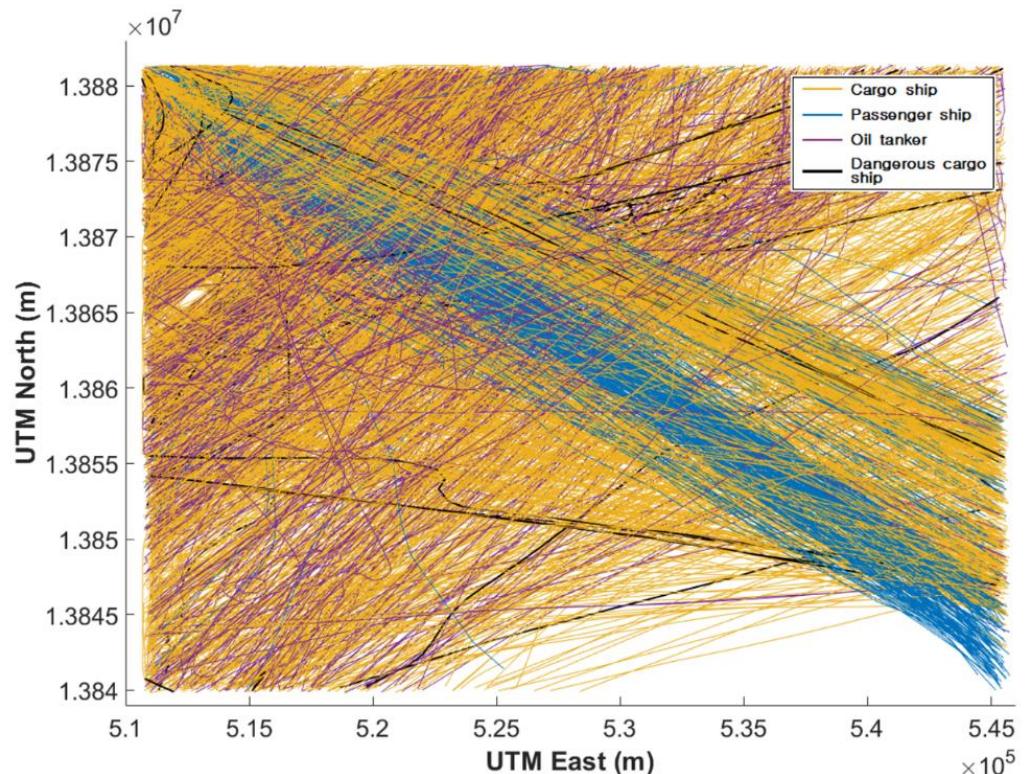
$$z_t = \sigma(U_z x_t + W_z h_{t-1} + b_z) \quad (22)$$

$$h_t = z_t \circ h_{t-1} + (1 - z_t) \circ \tilde{h}_t \quad (23)$$

### 3. Simulations and Results

#### 3.1. Data Collection

To generate neural network models for predicting the ship trajectory, the actual AIS data were used as input data. The AIS data were collected for 14 days in the coastal waters near the entrance to the port of Busan in Korea [40]. The selected sea area had the highest number of vessel traffic and marine accidents in Korea over the past five years (2016–2020) [1,41]. The collected AIS data includes a total of 1351 ships and 2816 trajectories with four types of ships: cargo ship, passenger ship, oil tanker, dangerous cargo ship. Figure 6 shows the ship trajectories classified by ship type in the universal transverse Mercator (UTM) coordinate system. The entrance to the port of Busan is located at the top left corner in Figure 6. As shown in Figure 6, it can be seen that ship trajectories of various patterns are concentrated in the target area. If these trajectory data are directly used to develop a predictive model, it will be difficult to expect good performance.



**Figure 6.** Total trajectories detected from the AIS data.

#### 3.2. Results of Ship Trajectory Clustering

The ship trajectories were grouped together by similar patterns using the spectral clustering. Table 1 shows the results of the ship trajectory clustering, which include the label, the quantity, and the ratio of each cluster. As shown in Table 1, a total of 2816 trajectories were grouped into 29 clusters. It was found that the cluster of label 12 contains the maximum number of trajectories (7.63%), and the cluster of label 2 contains the minimum number of trajectories (1.17%). Based on these results, each trajectory cluster was

illustrated in Figure 7. Although some abnormal trajectories were included in the overall results, it can be seen that there were clear differences in the patterns among each cluster.

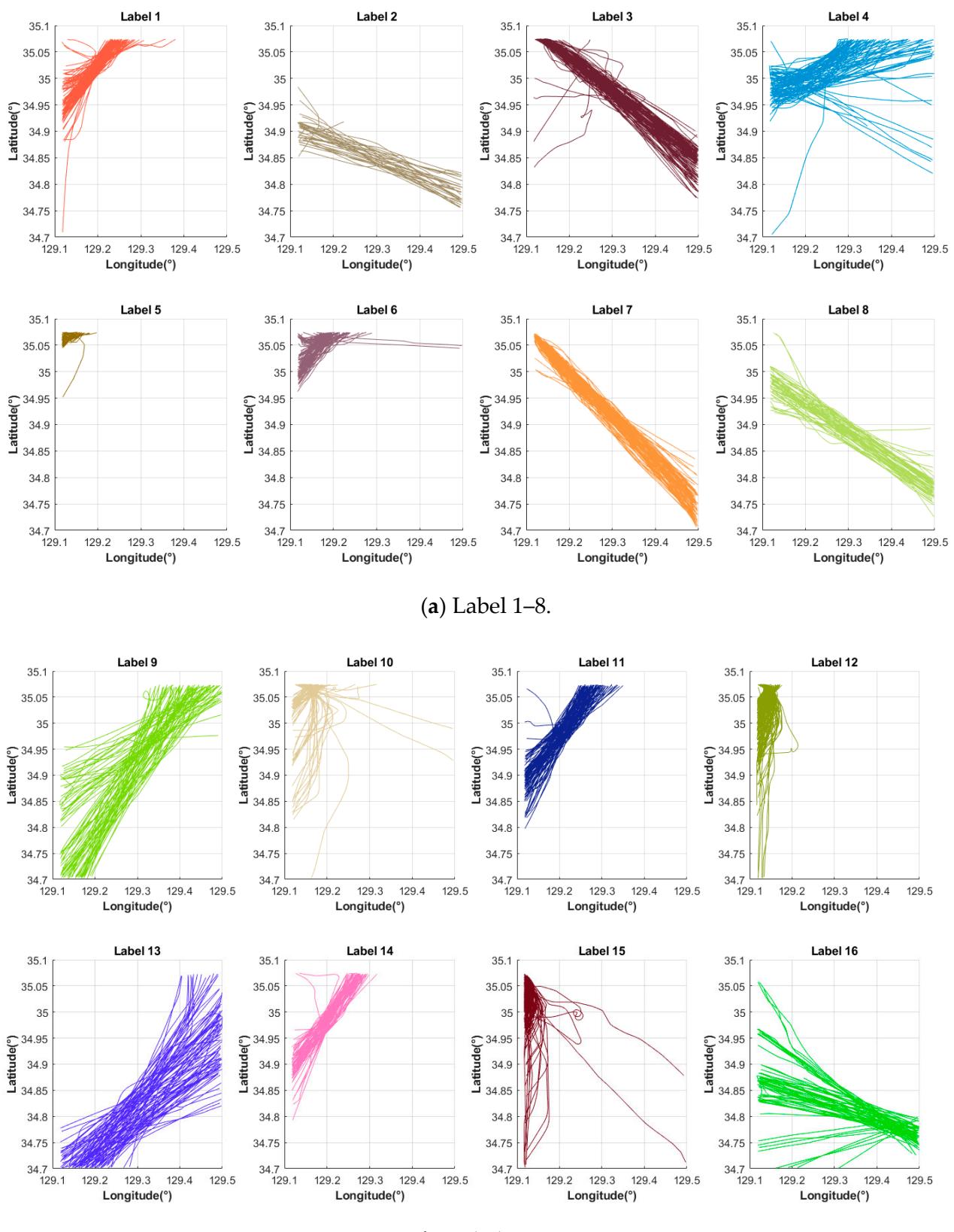
**Table 1.** Result of ship's trajectory clustering.

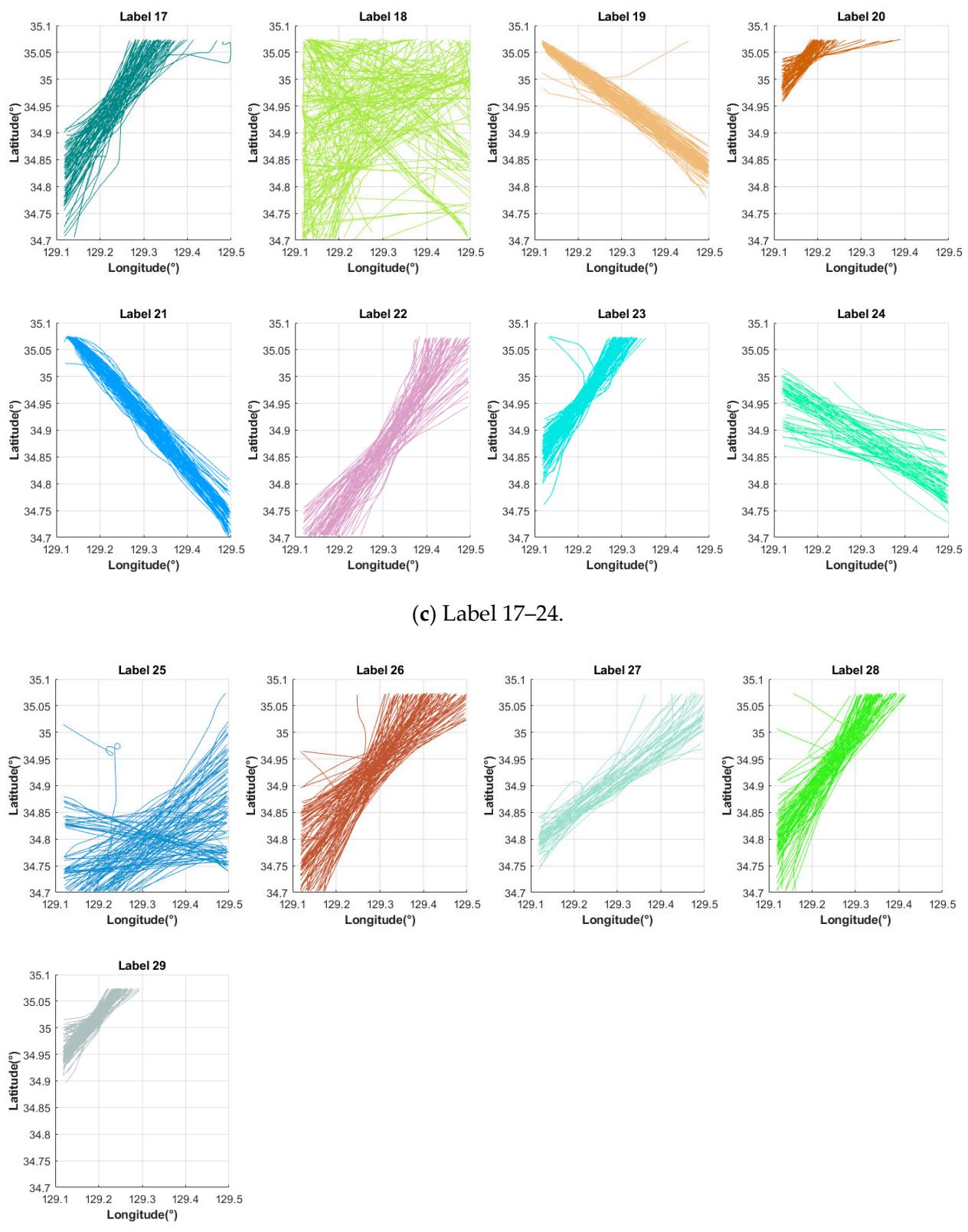
Label	Quantity	Ratio (%)	Label	Quantity	Ratio (%)	Label	Quantity	Ratio (%)
1	104	3.69	11	104	3.69	21	89	3.16
2	33	1.17	12	215	7.63	22	75	2.66
3	124	4.40	13	94	3.34	23	88	3.13
4	102	3.62	14	80	2.84	24	62	2.20
5	77	2.73	15	195	6.92	25	100	3.55
6	108	3.84	16	64	2.27	26	102	3.62
7	101	3.59	17	75	2.66	27	51	1.81
8	58	2.06	18	165	5.86	28	81	2.88
9	87	3.09	19	101	3.59	29	121	4.30
10	85	3.02	20	75	2.67	-	-	-

In Figure 7, since the entrance to the port of Busan is located at the top left corner in the trajectory plot of each cluster, the clusters of label 3, 7, 10, 12, 15, 19, 21 represent the inbound or outbound ship trajectories, and the clusters of label 1, 4, 9, 11, 13, 14, 17, 20, 22, 23, 25–29 represent passing ship trajectories in front of the port of Busan. According to the above clustering results, the patterns of trajectory were found to have two major patterns: northeast or southwest direction, northwest or southeast direction, which accounted for the largest proportion of the total trajectories, neglecting the abnormal trajectories. Therefore, two corresponding patterns were designated as group A and group B, respectively, and the trajectory data were extracted and classified by the groups as Table 2. Thus, the amount of sample data was 565 and 535 for group A and B, respectively.

**Table 2.** Classification of ship's trajectory pattern.

Group	Cluster Label	Direction	Quantity
A	9, 13, 17, 22, 26, 27, 28	Northeast/Southwest	565 (20.1%)
B	3, 7, 8, 19, 21, 24	Northwest/Southeast	535 (19.0%)

**Figure 7. Cont.**



(d) Label 25–29.

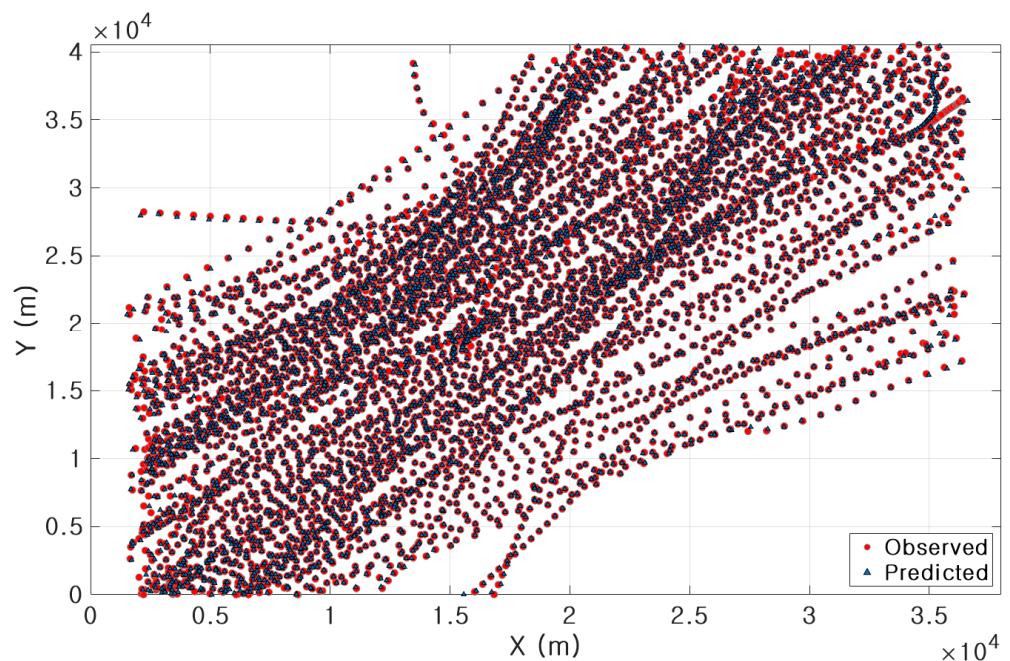
**Figure 7.** Ship trajectory plot of each cluster.

### 3.3. Results of Ship Trajectory Prediction

The trajectory prediction is performed by the two groups based on the Bi-LSTM, the LSTM, and the GRU. To avoid the data overfitting, we used a 5-fold cross-validation for training sample data, which is a method of training a model by using 4 equal parts, as training data among the entire data were partitioned into 5 parts, and evaluating the accuracy of the model by using the remain 1 equal parts as test data. The model is built 5 times with each equal part being used as test data once, and the accuracy of each model is calculated. As previously stated, the input data of each group were scaled so that we can make an appropriate fit and prevent the training from diverging. The accuracy of the final model is estimated as the average of the accuracy values calculated during cross-validation. To measure the accuracy of the model, the root mean square error (*RMSE*) value was used. The *RMSE* measures the prediction error that indicates the difference between the predicted trajectory ( $\hat{Tr}$ ) of the model and the observed value ( $Tr$ ) by inputting the test data to the trained model. The *RMSE* is defined as:

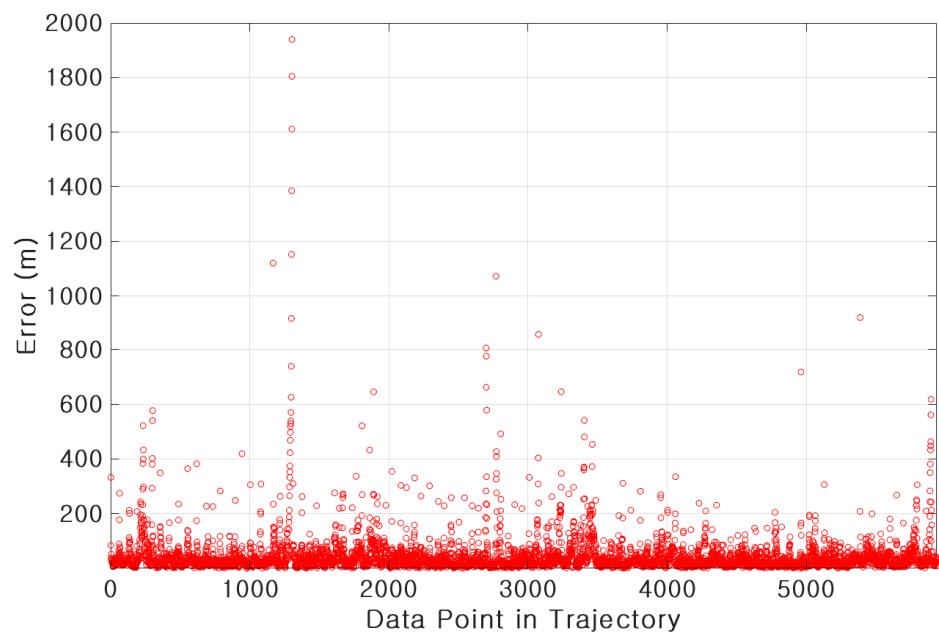
$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\hat{Tr}_i - Tr_i)^2}{n}} \quad (24)$$

A model that has the lower value of the *RMSE* on new data has the better generalization performance and can solve the problem of overfitting. To compare the performance of the Bi-LSTM, the LSTM, and the GRU, they were constructed under the same conditions [42]. The hidden layer was specified to have 200 hidden units. The adaptive moment estimation (ADAM) which is a method of training weights while adjusting the learning rate for each weight was used. The train epoch was set to 300, and the gradient threshold was set to 1. The initial learning rate was set to 0.005 and the learning rate drop after 120 epochs by multiplying by a factor of 0.2. The results of each prediction model are shown in Figure 8 using the 5th test data among cross-validation partitions.

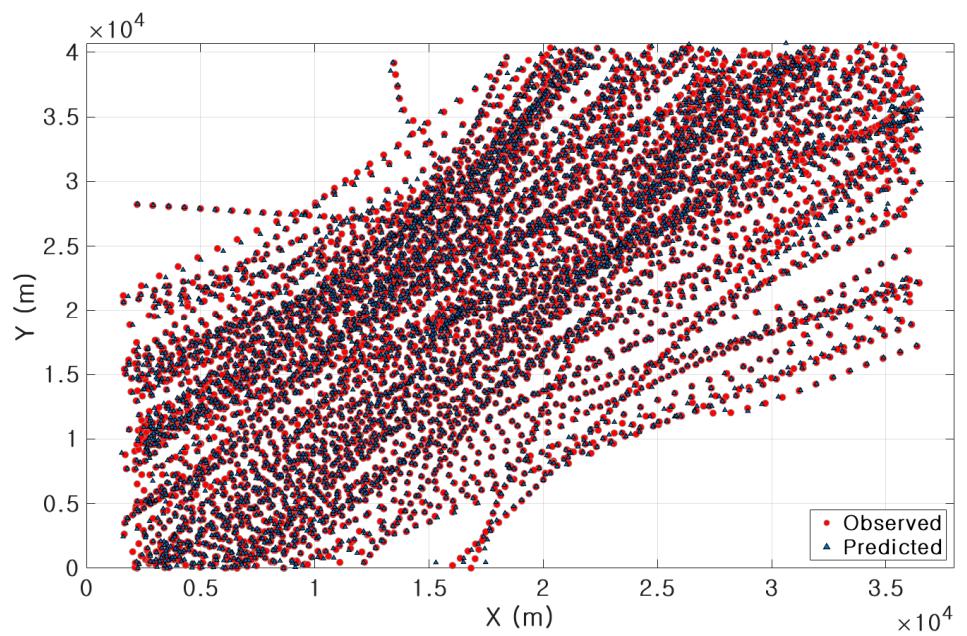


(a) Predicted trajectory for group A using Bi-LSTM.

**Figure 8.** Cont.

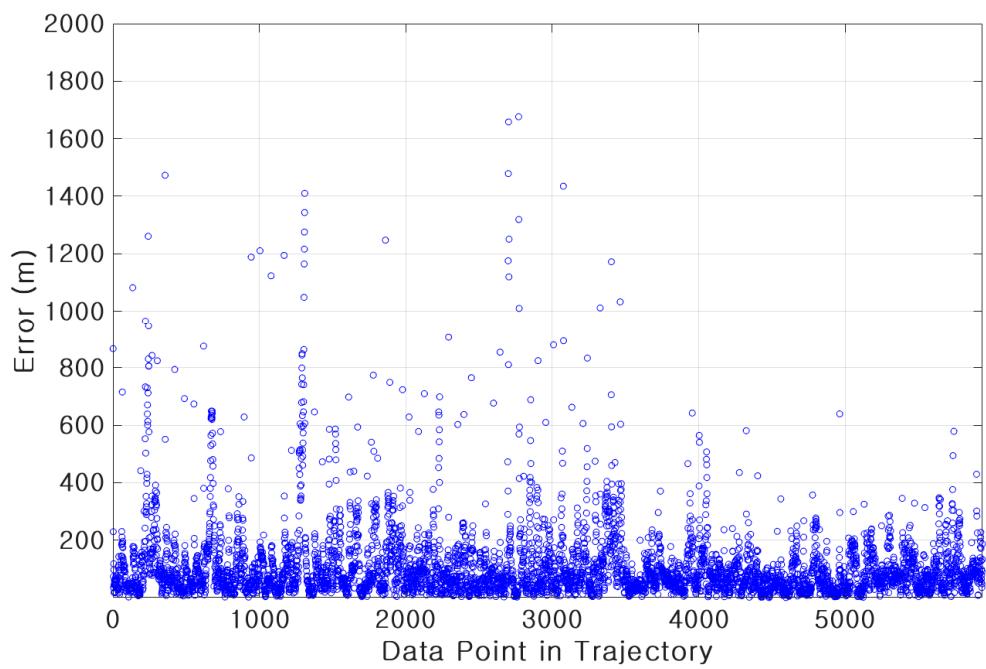


(b) Error scatter plot for group A using Bi-LSTM.

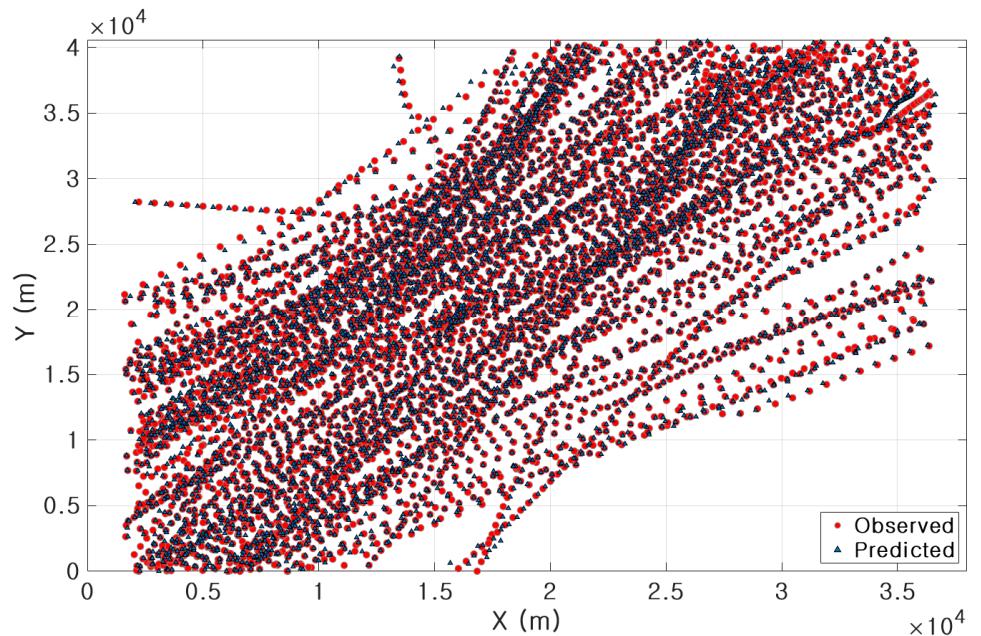


(c) Predicted trajectory for group A using LSTM

**Figure 8. Cont.**

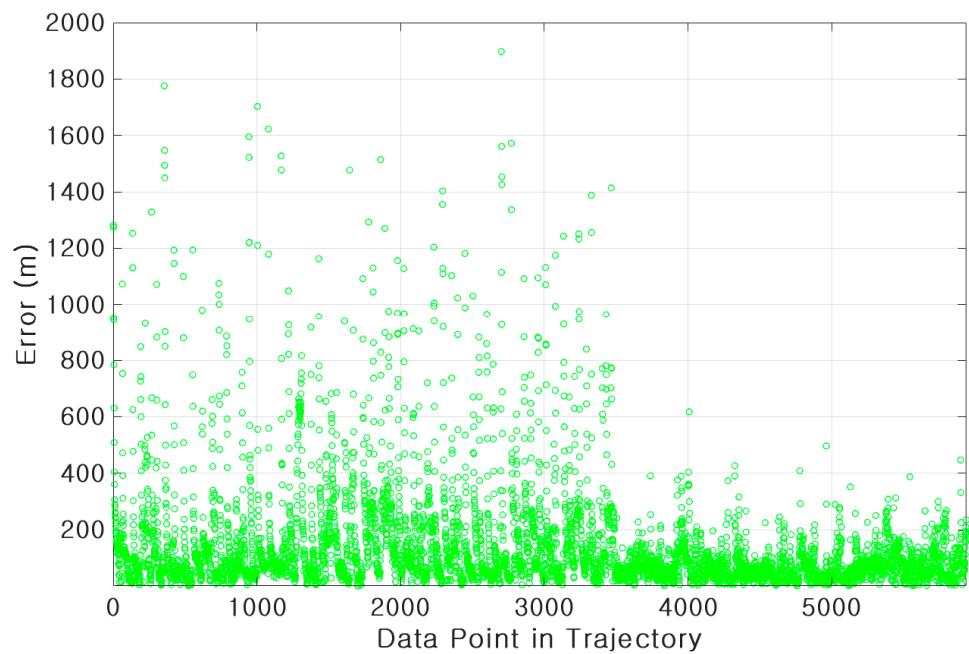


(d) Error scatter plot for group A using LSTM.

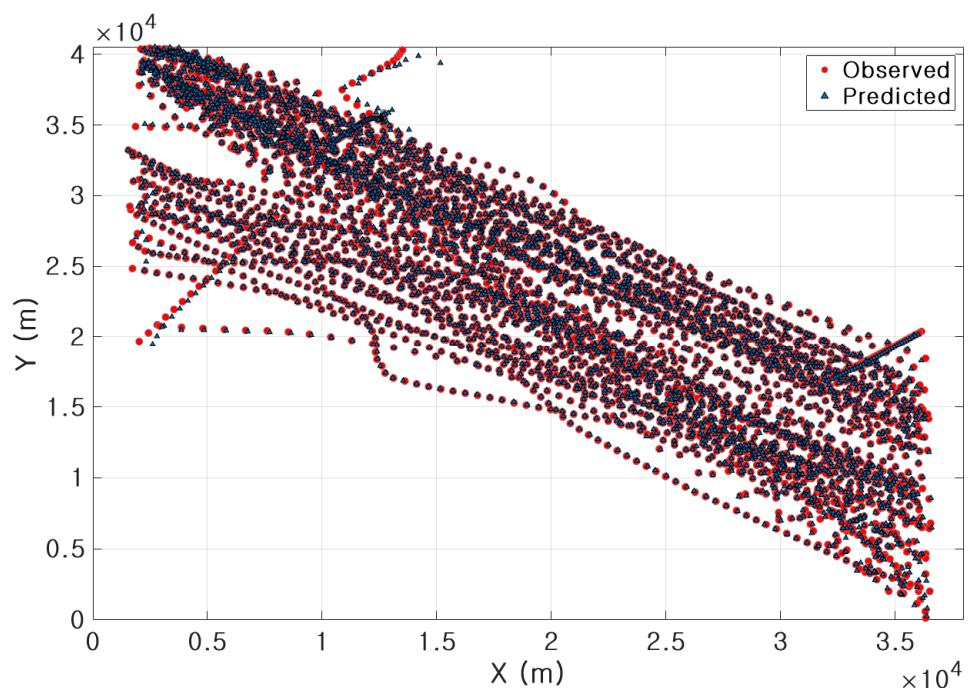


(e) Predicted trajectory for group A using GRU.

**Figure 8. Cont.**

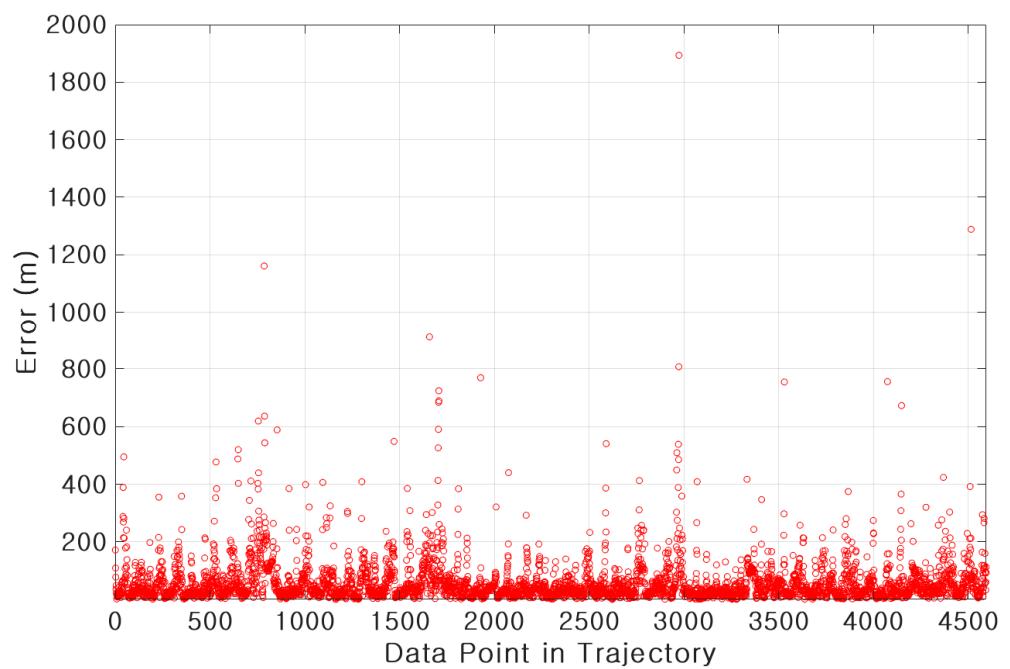


(f) Error scatter plot for group A using GRU.

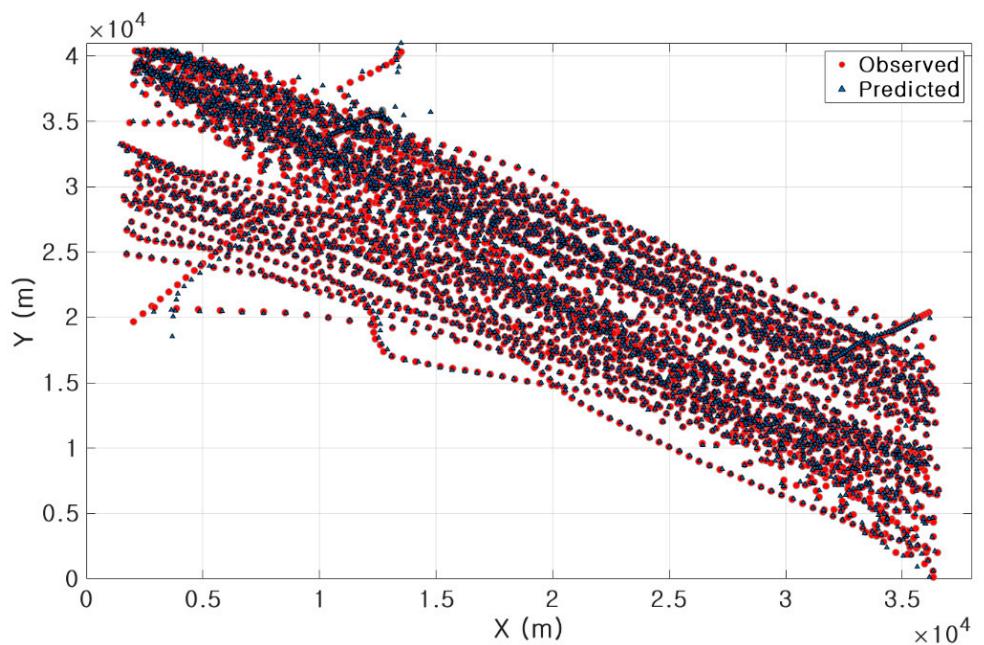


(g) Predicted trajectory for group B using Bi-LSTM.

**Figure 8. Cont.**

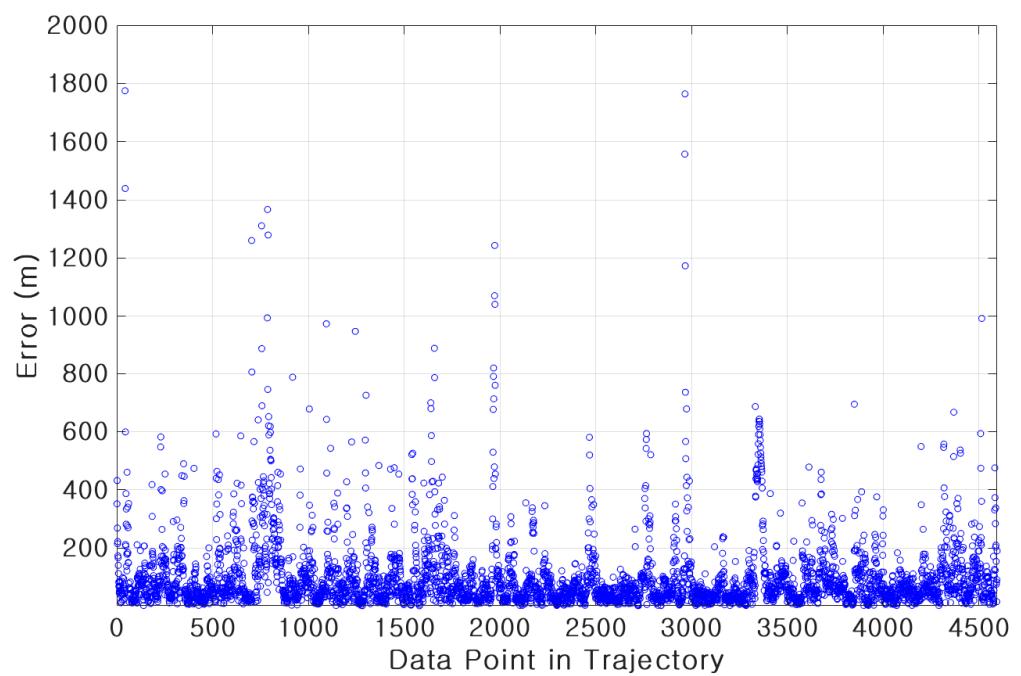


(h) Error scatter plot for group B using Bi-LSTM.

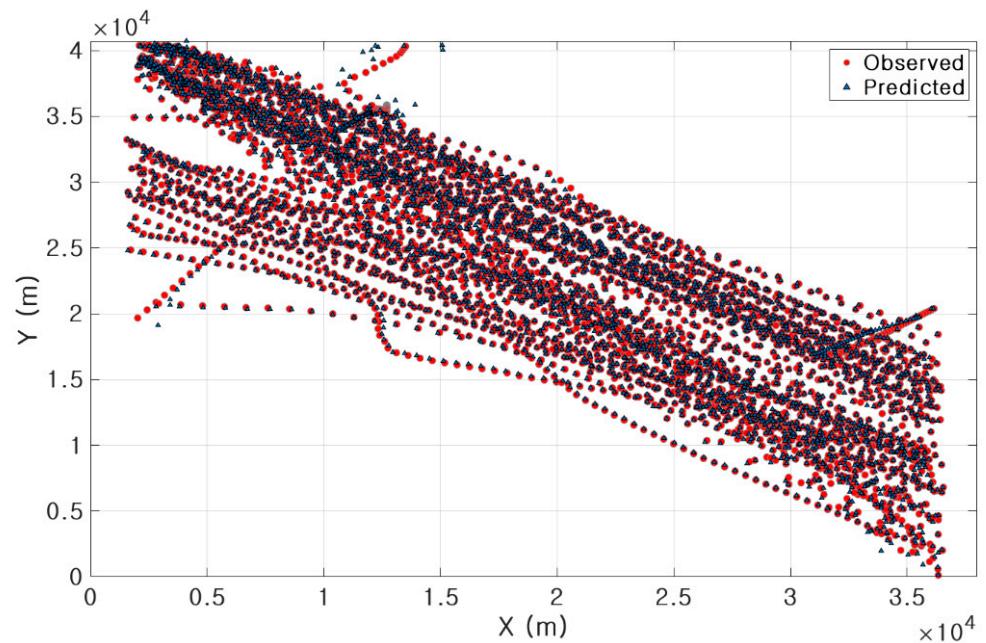


(i) Predicted trajectory for group B using LSTM.

**Figure 8. Cont.**

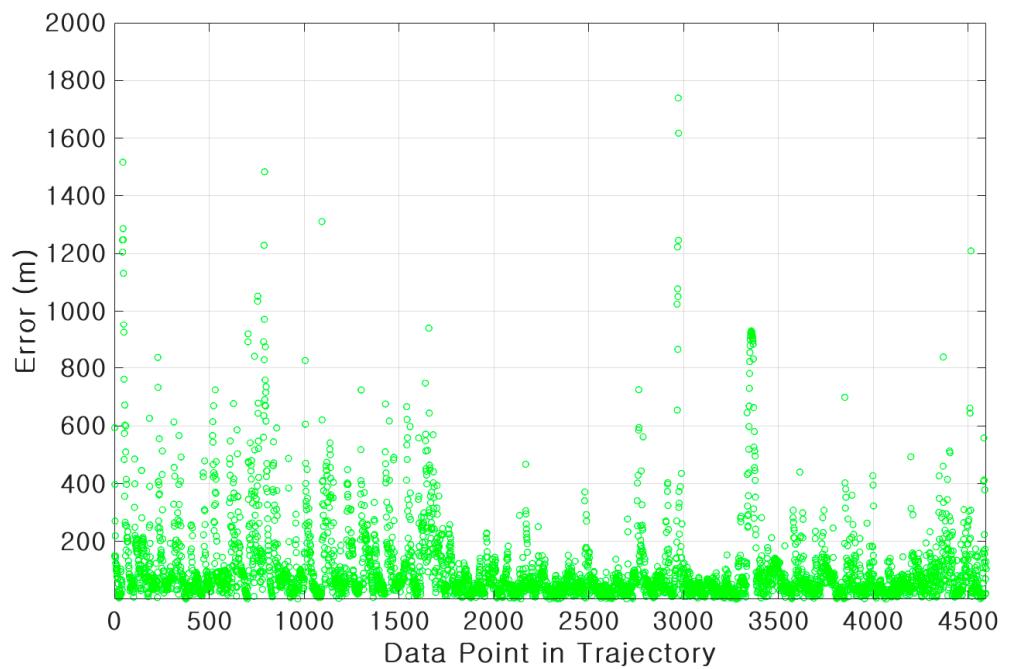


(j) Error scatter plot for group B using LSTM.



(k) Predicted trajectory for group B using GRU.

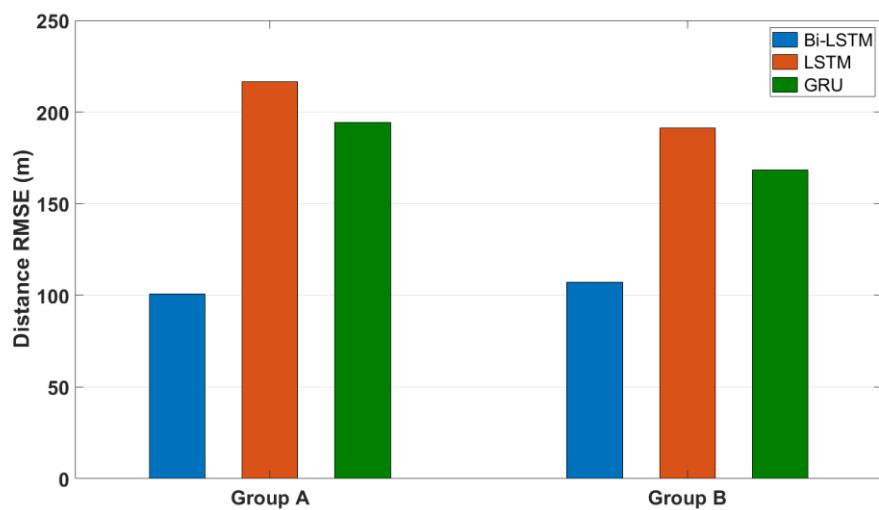
**Figure 8. Cont.**



(I) Error scatter plot for group B using GRU.

**Figure 8.** Results of trajectory prediction.

Figure 8 consists of the trajectory plots (a), (c), (e), (g), (i), (k) and the error scatter plots (b), (d), (f), (h), (j), (l). In the trajectory plots, the red trajectories represent the actual data observed by the AIS, and the blue trajectories represent the predicted data by each RNN model. The error scatter plots display the distance error between the observed data and the predicted data for each trajectory. According to each trajectory plot in Figure 8, the Bi-LSTM models accurately predict the trajectory in comparison with other models since the blue trajectories of Bi-LSTM were distributed closer to red trajectories than other models, and the LSTM and the GRU models showed a similar performance. The error scatter plots also showed that the Bi-LSTM models have the lower prediction errors for each data compared to the other two models. Moreover, according to (g), (i), and (k) of Figure 8, the Bi-LSTM showed superior prediction performance compared to the LSTM and the GRU models for the trajectory with a lot of changes in the course of the ship. The results of trajectory prediction applying the Bi-LSTM, the LSTM, and the GRU models to the group A and B are summarized in Figure 9 and Table 3. As shown in Figure 9, we can easily recognize that the prediction of the Bi-LSTM model is more accurate than the LSTM and the GRU in both group A and B. According to the normalized RMSE average in Table 3, the accuracy of the prediction model applied with the Bi-LSTM was higher than that of the LSTM and the GRU models, and the difference between the LSTM and the GRU was small. The prediction accuracy for the position was the highest among the position, course, and speed. In the comparison of the training times for the three models, the GRU model took less training time than the Bi-LSTM and the LSTM models, and the Bi-LSTM recorded the longest training time in both groups. The LSTM and the GRU take a longer training time compared to the general RNN because they have more weights and bias terms to be trained. In addition, since the Bi-LSTM is a structure in which backward training is added to the structure of the LSTM, it takes longer to learn than the LSTM.



**Figure 9.** Comparison of distance RMSE for each model.

**Table 3.** Results comparison among BLSTM, LSTM, and GRU.

Group	Method	Avg. Elapsed Training Time	RMSE			Normalized Avg. RMSE
			Distance	Course	Speed	
A	Bi-LSTM	22 min 1 s	0.0104 (101 m)	0.0345 (3.1°)	0.0275 (0.1 knot)	0.26
	LSTM	9 min 8 s	0.0224 (217 m)	0.1495 (13.4°)	0.1042 (0.3 knot)	1.00
	GRU	8 min 20 s	0.0202 (194 m)	0.1464 (13.1°)	0.1041 (0.3 knot)	0.98
B	Bi-LSTM	22 min 26 s	0.0113 (107 m)	0.0219 (2.0°)	0.0219 (0.2 knot)	0.27
	LSTM	9 min 45 s	0.0201 (191 m)	0.1242 (11.4°)	0.0583 (0.6 knot)	1.00
	GRU	8 min 24 s	0.0177 (169 m)	0.1251 (11.49°)	0.0583 (0.6 knot)	0.99

#### 4. Conclusions

In this study, we proposed a methodology for predicting the ship trajectory that can be used for an intelligent collision avoidance algorithm at sea. The Bi-LSTM, known as the RNN, was applied to predict the future trajectory of the ship by using the spectral clustered AIS data in the confined coastal waters. The results of the Bi-LSTM were compared to ones of the LSTM and the GRU models. For preparing the input data of the three models, the ship trajectories with the similar patterns were extracted by the spectral clustering method. The spectral-clustered AIS data were learned by using the RNN models to predict the trajectories of future time steps. The RMSEs of ship parameters—i.e., position, course, and speed—were calculated to compare with each other. It concludes that the Bi-LSTM model presents the better accuracy, compared with the LSTM and the GRU models. It has been proven that the performance of the Bi-LSTM is better than the LSTM applied in previous studies. In the future, performance comparison through quantitative evaluation between the DBSCAN and the spectral clustering should be implemented. Moreover, in the procedure of the trajectory clustering, it is necessary to define the major traffic patterns based on the clear evidence. In relation to the considerable low accuracy of prediction for some trajectories, the cause should be identified. This may be due to the absence of an optimization process for the hyper-parameters required for model development. Regarding

the evaluation of the developed model, the *RMSE* alone cannot sufficiently evaluate and compare the model. The application of various evaluation metrics can block qualitative evaluations, such as comparison of graphs with large amounts of data. Furthermore, despite the collection of large amounts of data, only 1098 out of 2816 trajectories were used in this study. Considering the scalability of future research, the inefficiency of data utilization must be addressed. We constructed the prediction model for the ship trajectory based on the spectral clustering and the Bi-LSTM. It is expected that the proposed model contributes to developing an intelligent collision avoidance algorithm, which can reduce the human error in determining the risk of collision between ships and take collision avoidance actions at an early stage.

**Author Contributions:** Conceptualization, J.P., J.J. and Y.P.; data curation, J.P.; formal analysis, J.P. and J.J.; investigation, J.P. and J.J.; methodology, J.P., J.J. and Y.P.; resources, J.J. and Y.P.; supervision, Y.P.; validation, J.J. and Y.P.; visualization, J.P. and Y.P.; writing—original draft, J.P.; writing—review & editing, J.J. and Y.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The datasets analyzed or generated in this study are available from the corresponding author upon reasonable request.

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

## References

1. KMST (Korean Maritime Safety Tribunal) 2020 Annual Report of Marine Accident Statistics. Available online: <https://www.kmst.go.kr> (accessed on 3 May 2021).
2. McLane, R.C.; Wolf, J.D. Symbolic and Pictorial Displays for Submarine Control. *IEEE Trans. Hum. Factors Electron.* **1967**, HFE-8, 148–158. [[CrossRef](#)]
3. Inoue, S.; Hirano, M.; Kijima, K.; Takashina, J. Practical Calculation Method of Ship Maneuvering Motion. *Int. Shipbuild. Prog.* **1981**, *28*, 207–222. [[CrossRef](#)]
4. Fossen, T.I. *Handbook of Marine Craft Hydrodynamics and Motion Control*; John Wiley & Sons: Chichester, UK, 2011; ISBN 9781119991496.
5. Passenier, P.O. An Adaptive Track Predictor for Ships. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 1989.
6. Czapiewska, A.; Sadowski, J. Algorithms for Ship Movement Prediction for Location Data Compression. *TransNav Int. J. Mar. Navig. Saf. Sea Transp.* **2015**, *9*, 75–81.
7. Schöller, C.; Aravantinos, V.; Lay, F.; Knoll, A. What the constant velocity model can teach us about pedestrian motion prediction. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1696–1703. [[CrossRef](#)]
8. Johansen, T.A.; Perez, T.; Cristofaro, A. Ship collision avoidance and COLREGS compliance using simulation-based control behavior selection with predictive hazard assessment. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3407–3422. [[CrossRef](#)]
9. Last, P.; Bahlke, C.; Hering-Bertram, M.; Linsen, L. Comprehensive Analysis of Automatic Identification System (AIS) Data in Regard to Vessel Movement Prediction. *J. Navig.* **2014**, *67*, 791–809. [[CrossRef](#)]
10. Sang, L.; Yan, X.; Wall, A.; Wang, J.; Mao, Z. CPA calculation method based on AIS position prediction. *J. Navig.* **2016**, *69*, 1409–1426. [[CrossRef](#)]
11. van Breda, L.; Passenier, P.O. Effect of path prediction on navigational performance. *J. Navig.* **1998**, *51*, 216–228. [[CrossRef](#)]
12. Laxhammar, R. Anomaly detection for sea surveillance. In Proceedings of the 2008 11th International Conference on Information Fusion, Cologne, Germany, 30 June–3 July 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 1–8.
13. Ristic, B.; La Scala, B.; Morelande, M.; Gordon, N. Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction. In Proceedings of the 2008 11th International Conference on Information Fusion, Cologne, Germany, 30 June–3 July 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 1–7.
14. Aarsæther, K.G.; Moan, T. Estimating navigation patterns from AIS. *J. Navig.* **2009**, *62*, 587. [[CrossRef](#)]
15. Tang, H.; Wei, L.; Yin, Y.; Shen, H.; Qi, Y. Detection of abnormal vessel behaviour based on probabilistic directed graph model. *J. Navig.* **2020**, *73*, 1014–1035. [[CrossRef](#)]
16. Łącki, M. Intelligent prediction of ship maneuvering. *TransNav Int. J. Mar. Navig. Saf. Sea Transp.* **2016**, *10*, 511–516. [[CrossRef](#)]
17. Xu, T.; Liu, X.; Yang, X. Ship Trajectory online prediction based on BP neural network algorithm. In Proceedings of the 2011 International Conference of Information Technology, Computer Engineering and Management Sciences, Nanjing, China, 24–25 September 2011; IEEE: Piscataway, NJ, USA, 2011; Volume 1, pp. 103–106.

18. Zhou, H.; Chen, Y.; Zhang, S. Ship trajectory prediction based on BP neural network. *J. Artif. Intell.* **2019**, *1*, 29. [[CrossRef](#)]
19. Zhao, L.; Shi, G. Maritime anomaly detection using density-based clustering and recurrent neural network. *J. Navig.* **2019**, *72*, 894–916. [[CrossRef](#)]
20. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [[CrossRef](#)]
21. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
22. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
23. Shi, Z.; Pan, Q.; Xu, M. LSTM-Cubic A\*-based auxiliary decision support system in air traffic management. *Neurocomputing* **2020**, *391*, 167–176. [[CrossRef](#)]
24. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [[CrossRef](#)]
25. Siami-Namini, S.; Tavakoli, N.; Namin, A.S. The performance of LSTM and BiLSTM in forecasting time series. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 3285–3292.
26. Riveiro, M.; Pallotta, G.; Vespe, M. Maritime anomaly detection: A review. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1266. [[CrossRef](#)]
27. Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Portland, OR, USA, 2–4 August 1996; Volume 96, pp. 226–231.
28. Vlachos, M.; Kollios, G.; Gunopulos, D. Discovering similar multidimensional trajectories. In Proceedings of the 18th International Conference on Data Engineering, San Jose, CA, USA, 26 February–1 March 2002; IEEE: Piscataway, NJ, USA, 2002; pp. 673–684.
29. IMO (International Maritime Organization). *Adoption of New and Amended Performance Standards for Navigational Equipment*; IMO: London, UK, 1998; Volume 86, pp. 13–16.
30. Sang, L.Z.; Yan, X.P.; Mao, Z.; Ma, F. Restoring method of vessel track based on AIS information. In Proceedings of the 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science, Guilin, China, 19–22 October 2012; pp. 336–340. [[CrossRef](#)]
31. Zhang, D.; Li, J.; Wu, Q.; Liu, X.; Chu, X.; He, W. Enhance the AIS data availability by screening and interpolation. In Proceedings of the 2017 4th International Conference on Transportation Information and Safety (ICTIS), Banff, AB, Canada, 8–10 August 2017; pp. 981–986. [[CrossRef](#)]
32. Shi, Z.; Xu, M.; Pan, Q.; Yan, B.; Zhang, H. LSTM-based flight trajectory prediction. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–8.
33. Morris, B.T. Understanding Activity from Trajectory Patterns. Ph.D. Thesis, University of California San Diego, La Jolla, CA, USA, 2010.
34. Shi, J.; Malik, J. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 888–905.
35. Von Luxburg, U. A tutorial on spectral clustering. *Stat. Comput.* **2007**, *17*, 395–416. [[CrossRef](#)]
36. Ng, A.; Jordan, M.; Weiss, Y. On spectral clustering: Analysis and an algorithm. *Adv. Neural Inf. Process. Syst.* **2001**, *14*, 849–856.
37. Lloyd, S. Least squares quantization in PCM. *IEEE Trans. Inf. Theory* **1982**, *28*, 129–137. [[CrossRef](#)]
38. Gers, F.A.; Schmidhuber, J.; Cummins, F. Learning to forget: Continual prediction with LSTM. *IEE Conf. Publ.* **1999**, *2*, 850–855. [[CrossRef](#)]
39. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [[CrossRef](#)]
40. Park, J.; Jeong, J.S. An Estimation of Ship Collision Risk Based on Relevance Vector Machine. *J. Mar. Sci. Eng.* **2021**, *9*, 538. [[CrossRef](#)]
41. Ministry of Oceans and Fisheries. Statistics of Vessels Arrival and Departure at Major Port of Korea. Available online: <http://www.mof.go.kr> (accessed on 3 May 2021).
42. Park, J. A Study on the Estimation of Ship Collision Risk Using Machine Learning and Its Optimal Path Finding. Ph.D. Thesis, Mokpo National Maritime University, Mokpo, Korea, 2021.