Preprints.org

Article

A Ship State Recognition Method Based on Graph Convolutional Neural Network

Lin Ma , Hao Cao [*] , Guo-You Shi , Kaige Zhu , Weifeng Li , Shengyan Qin

Article

A Ship State Recognition Method Based on Graph Convolutional Neural Network

Lin Ma [1,2,*], Hao Cao [1,2], Guo-You Shi [1,2], Kaige Zhu [3], Weifeng Li [1,2] and Shengyan Qin [1]

[1] Navigation College, Dalian Maritime University, Dalian 116026, Liaoning, China;
     m13109809082@163.com (H.C.); sgydmu@dlmu.edu.cn (G.-Y.S); zkg@dlmu.edu.cn (K.Z.);
     dmulwf@dlmu.edu.cn (W.L.); q2220224068@dlmu.edu.cn (S.Q.)
[2] Key Laboratory of Navigation Safety Guarantee of Liaoning Province, China
[3] Ningbo University
*   Correspondence: malindmu@dlmu.edu.cn

Abstract: To address the challenge of recognizing ship states accurately amidst the complexities of marine environments, this study proposes a novel ship state recognition approach leveraging a graph convolutional neural network (GCNN). Initially, the method extracts canonical and efficient ship motion trajectories from AIS data. Subsequently, a state recognition network tailored for ship motion trajectories is devised and implemented employing graph convolution. Notably, the accuracy of this model is enhanced through the introduction of novel weights and optimization of the Adj parameter. Experimental evaluations conducted on a ship state dataset demonstrate significant performance improvements. Specifically, the proposed recognition network achieves a recognition accuracy of 98.3% for regulated ship trajectories, marking an impressive 8.4% enhancement over traditional convolutional neural networks. This advancement holds promise for enhancing ship state recognition accuracy across diverse maritime applications including maritime supervision, navigation safety, and ship management.

Keywords: maritime transportation networks; graph convolutional neural networks; AIS data

1. Introduction

Shipping holds a paramount position in global goods transportation. In the era of big data and smart navigation, constructing and enhancing maritime traffic networks is imperative for effective maritime traffic planning and management. This is crucial to meet the escalating and intricate shipping demands and to furnish ship operators with optimized navigational recommendations. Zhe et al. (2020) conducted a review of maritime traffic network research, underscoring its significance and vast potential for enhancing maritime traffic safety and intelligence. They emphasized the importance of leveraging advancements in pattern mining, traffic prediction, and integrating technologies such as the Internet of Things (IoT), Artificial Intelligence (AI), Knowledge Engineering (KE), and Big Data Computing (BDC) solutions in maritime transportation network optimization[1].

The evolution of maritime transportation networks provides ample data and communication resources, fostering the development of smart navigation systems. This contributes to heightened safety, efficiency, and sustainability while bolstering the operation and management of the entire maritime transportation industry[2–5]. For instance, Po-Ruey et al. conducted analyses of ship collision accidents using actual AIS data to inform encounter posture judgment and collision avoidance decision-making, thereby enhancing maritime traffic safety control[5]. Similarly, Dominik et al. utilized evolutionary algorithms to develop a methodological tool facilitating the automated discovery of waypoints and optimal route definition, with the simulated maritime traffic network closely resembling real-world counterparts[3]. These studies confirm that AIS data can be used as an accurate data source when modeling the maritime traffic network and can be mined for effective ship motion state information.

The identification of vessel states is pivotal in the construction of maritime transportation networks[6–9]. Real-time monitoring and analysis of ship position, speed, and heading, among other parameters, empower ship state recognition systems to avert collision accidents and enhance ship safety. Moreover, by analyzing ship state data, traffic flow can be optimized, enhancing overall network efficiency and alleviating congestion. Ship-state recognition systems foster collaboration among various stakeholders, facilitating the development of a more intelligent, secure, and efficient maritime transportation network. Consequently, ship state recognition emerges as a vital technology for modern maritime transportation network construction, profoundly impacting operational efficiency, safety, and sustainability across the maritime transportation system.

Previous research has enlisted numerous scholars to aid in the construction of maritime transportation networks through the processing of vast AIS data. Po-Ruey et al. analyzed ship collision accidents based on actual AIS data, utilizing it to assess rendezvous postures and inform collision avoidance decision-making, thereby enhancing maritime transportation safety under the purview of transportation managers[5]. Research has also affirmed the high reliability of AIS data for studying maritime traffic safety. Dominik et al. developed a methodological tool using evolutionary algorithms to assist navigators in planning maritime routes by automatically discovering waypoints and defining optimal routes. During the study of AIS data, the simulated maritime traffic network using this tool exhibited significant overlap with the actual maritime traffic network[3]. This underscores the feasibility of extracting ship state characteristics from AIS data to aid in constructing maritime transportation networks. To efficiently identify ship state information, this study plans to utilize a graph convolutional neural network for feature extraction from AIS data to facilitate state identification.

A graph convolutional neural network is a type of convolutional neural network capable of capturing associations between nodes of image data and extracting features based on interrelationships among each node of AIS data. Sravan et al. employed a Multi-Relational Graphic Convolutional Neural Network (MR-GCN) for the state recognition of moving vehicles, with the experimental structure of the model demonstrating strong accuracy and robustness[10]. Given the significant error inherent in the original ship AIS data regarding state information, this study aims to employ a graph convolutional neural network to train the model and perform feature extraction on the AIS data to achieve accurate ship state recognition.

This paper aims to construct a graph convolutional neural network to extract key features of ship movement and identify the ship's state based on ship navigation information such as ship position, time differences, and changes in ship heading provided by the ship's AIS data. The objective is to provide effective navigation reference information for the maritime transportation network.

2. Literature Review

2.1. Subsection

Ship's AIS data encompasses both static information, such as the ship's IMO (International Maritime Organization) number, timestamp, latitude, longitude, heading, bow direction, etc., and dynamic information including speed, sailing time, etc. Different ship statuses exhibit distinct dynamic and static characteristics. This data serves various purposes, from real-time monitoring of a ship's position and status to applications like accident investigation, waterway planning, and ship traffic management. The accuracy of AIS data is paramount as it enhances the safety, efficiency, and sustainability of maritime transportation networks, promoting organized ship transportation between seas and ports while providing effective navigation guidance to ship decision-makers. AIS communications between ships and shore-based facilities contribute to collision risk reduction and bolster the efficiency and safety of maritime transportation. A comprehensive analysis of a ship's AIS data can significantly aid ship navigation[11–13]. V. Fernandez et al. introduced a method to automatically generate an integrated maritime traffic network from historical ship AIS data. This approach offers a structured characterization of maritime traffic flow, paving the way for real-time automated monitoring, anomaly detection, and posture prediction of maritime traffic[2]. Mao et al. evaluated an Extreme Learning Machine (ELM)--based path prediction method using an AIS

database. The results demonstrated the database's utility as a standardized training resource for various trajectory prediction algorithms and other AIS data mining applications[13].

2.2. Conventional Ship Condition Recognition Methods

Ship state recognition plays a crucial role in maritime traffic management and ship safety. Traditional methods are generally classified into two categories: rule-based and machine-learning-based approaches. Rule-based methods rely on predefined navigation rules, which can be static or dynamic. Static rules encompass predetermined navigational criteria such as speed limits and restricted areas, while dynamic rules utilize real-time ship states like speed and heading to enforce specific navigation rules. Some researchers have proposed methods for extracting ship-stopping information by integrating trajectory features with geographic scene semantics. This approach accurately identifies a ship's docking point. Building upon this concept of feature extraction, our study combines AIS data features to identify ship trajectory data and distinguish various ship states, thereby enhancing maritime transportation network efficiency. While effective in simpler scenarios, this method may struggle to adapt to complex maritime environments.

Machine learning-based methods, on the other hand, identify ship states by analyzing extensive trajectory data. Supervised learning employs labeled datasets, comprising trajectories and corresponding state labels, to train models. In contrast, unsupervised learning recognizes similar trajectory patterns through clustering algorithms without relying on pre-labeled data. However, traditional methods face limitations and challenges. Firstly, they are sensitive to trajectory noise and prone to misclassification due to sensor errors and environmental factors. Secondly, these methods often treat individual ship states independently, making it challenging to capture complex interaction effects between ships in congested waters. Additionally, models may struggle to generalize across different regions and conditions, necessitating significant domain knowledge and manual adjustments. Traditional methods may also perform poorly when encountering new, emerging, or uncommon ship states. Chen et al. introduced a maritime image ship detection framework based on integrated YOLO, demonstrating accurate ship state recognition across consecutive frames[6]. While this method efficiently identifies the state of the ship, it imposes high requirements on dynamic image data and may struggle to effectively capture ship operations in complex maritime environments.

To address these challenges, contemporary research endeavors to integrate traditional methods with novel data-driven approaches like deep learning. By leveraging deep learning's capability to learn complex spatiotemporal patterns from large-scale data, models can enhance the understanding and recognition of ship states in intricate maritime settings, thereby providing more reliable support for maritime traffic management and ship safety. Building upon this background, our study adopts a graph convolutional neural model to extract features from AIS data. We iteratively optimize model parameters through multiple rounds of training to achieve higher accuracy rates for ship states, thereby constructing a comprehensive ship state recognition model.

2.3. Graph Convolutional Neural Network

Graph Convolutional Neural Networks (GCNs) have demonstrated exceptional performance in modeling and analyzing graph-structured data across various domains. They have yielded promising results in applications such as social networks, bioinformatics, and recommender systems. In social networks, GCNs are commonly employed to discern intricate relationships between nodes, thereby enhancing the understanding of user states.

An end-to-end graph neural network called MR-GNN has been introduced for multi-resolution bi-graph neural networks, facilitating the prediction of structured entity interactions [14], Furthermore, a study explored semi-supervised classification utilizing graph convolutional networks for learning from graph-structured data in a semi-supervised manner[15]. Additionally, a spatio-temporal multi-graph convolutional network (STMGCN)-based trajectory prediction framework has been developed, leveraging the Mobile Edge Computing (MEC) paradigm for ship trajectory prediction [16]. Another novel deep learning framework has been proposed for ship speed prediction, integrating k-hop graph convolutional networks (k-GCN) and long short-term memory

(LSTM) models[17]. Furthermore, a deep attention-aware spatiotemporal graph convolutional network (DAA-SGCN) based on AIS data has been introduced for predicting future ship trajectories [18]. These advancements highlight the versatility and efficacy of GCNs in addressing complex challenges across diverse domains.

In ship state recognition, Graph Convolutional Networks (GCNs) offer significant advantages over traditional methods by considering interactions between ships, thereby enhancing modeling capabilities in congested waters. GCNs exhibit adaptability to new state types and can automatically adjust to diverse maritime environments across regions and conditions. They effectively handle trajectory data noise and enhance state accuracy by leveraging global information learning. The integration of GCNs presents new opportunities for ship state recognition, enabling a more comprehensive and dynamic understanding of relationships between ships, thus improving recognition accuracy and robustness.

Unlike conventional Convolutional Neural Networks (CNNs), GCNs synthesize connections between each node in image data, facilitating better extraction of state information from AIS data and yielding richer feature sets. The model comprehensively considers connections within AIS data, exploring feature relationships during model optimization. In ongoing research, the application of GCNs in ship state identification promises to provide advanced technical support for maritime traffic management and ship safety.

Within processed AIS data, each feature data exhibits diverse associations across different time intervals. Relationships between non-adjacent data rows can represent various ship state features. For instance, conducting feature extraction with 10-state time intervals facilitates the identification of motion characteristics and position changes. By consecutively extracting features with state time intervals, instant motion characteristics of the data become apparent. Graph Convolutional Neural Network feature extraction should consider connections between feature values across different data types within a set of states. Furthermore, understanding the mutual influence of these data associations is essential for effective model development.

3. Methodology

In this chapter, a state identification model utilizing an optimized graph convolutional neural network (GCN) will be proposed. The initial section will focus on cleaning and processing AIS data to extract cleaned ship trajectory segments. Following this, the structure and content of a conventional GCN will be introduced. Subsequently, the features of the AIS data and processed structural information will be combined to optimize the model, aligning it with the experiment's data structure to achieve initial model construction.

In the third section, the model's key parameters will be optimized by leveraging the connections between AIS data features to extract data features more efficiently. This optimization process will lead to the development of a fully optimized graphical convolutional neural network model.

3.1. AIS Data Processing

3.1.1. Cleaning of AIS Data

In traditional ship state identification, the ship state in AIS data serves as the primary determinant, with reliable state data exhibiting continuous stability before and after state changes. However, upon randomly selecting several sets of ship databases and analyzing their ship state change differences along with their neighboring state differences, as depicted in Figure 1 below, it becomes evident that the data exhibits significant errors and large amplitude changes. This observation underscores the inadequacy of the original AIS data in providing effective ship state reference data.

As a result, this study opts to select trajectory segments with stable ship states for characterization during data processing, ensuring the utilization of reliable ship state data.
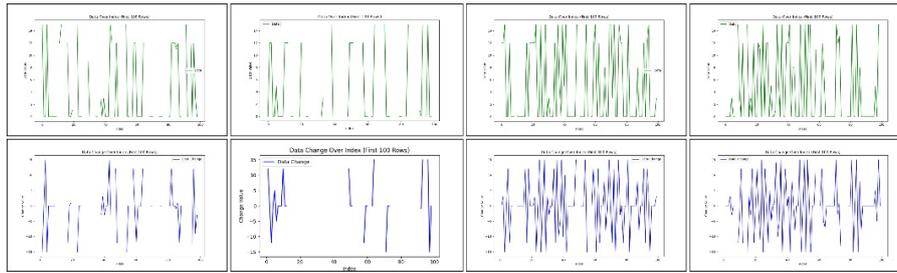
Figure 1. Trends in the "status" of AIS data.

The raw AIS data provided by the ship has a large error and needs to be cleaned and processed to minimize the experimental process error. Table 1. below shows the initial processing of the main parameters:

Table 1. Preliminary process for the main parameters.

Main parameters	Preliminary treatment
MMSI	Unique number for each ship to distinguish between different ships
BaseDateTime	Time in UTC format, converted to general time format
LAT\LON	The latitude and longitude of the ship, increased by a factor of 600,000, used as sailing distances in the study
SOG	Vessel speed to ground, value 102.3 is invalid and should be deleted.
COG	course to ground
Heading	Vessel heading, the value of 511 is invalid data, deleted.
Status	Vessel sailing status, as the primary label, for training and validation of data sets
Other data	No significant correlation with vessel status and not used in this study

The study will commence by processing AIS data, initially cleaning all AIS data to eliminate erroneous and highly erroneous data. Subsequently, stable ship trajectory segments will be selected, with each segment comprising every 50 consecutive data points. Labels will be assigned to each group of data based on the initial ship state. These labeled data groups will then undergo feature extraction using a preliminary graph convolutional neural network.

Following feature extraction, each group's states will be distinguished based on the extracted features. Bayesian optimization will be integrated into the model to optimize hyperparameters, enhancing model performance. Additionally, the weight of each AIS data feature will be incorporated into the optimization loop.

Given the temporal characteristics of AIS data, the Adj parameter in the graph convolutional neural network, representing data correlation, will be optimized to derive the optimal recognition model. This optimization process aims to achieve efficient ship state recognition.

3.1.2. AIS Data Characterization

To ensure the accuracy and validity of all data during model training, short data segments are insufficient to effectively capture the ship's motion characteristics. Therefore, every 50 AIS data points belonging to the same ship are grouped. Each data point within a group is continuous in the initial dataset, sharing the same state and with a time interval of no more than 10 minutes.

The original AIS data comprises only six-dimensional features: latitude, longitude, time, speed to ground, heading to ground, and bow direction. To enable the model to capture both static information features and motion features of the data while utilizing the continuity of the ship's motion, adjacent rows of AIS data are batch-calculated. This process provides the model with more

extractable features and diverse training information. Through batch calculation of data features, the original six dimensions are expanded to 16 dimensions. Notably, time difference and ground speed have broader coverage and are inherited between groups when processing the data.
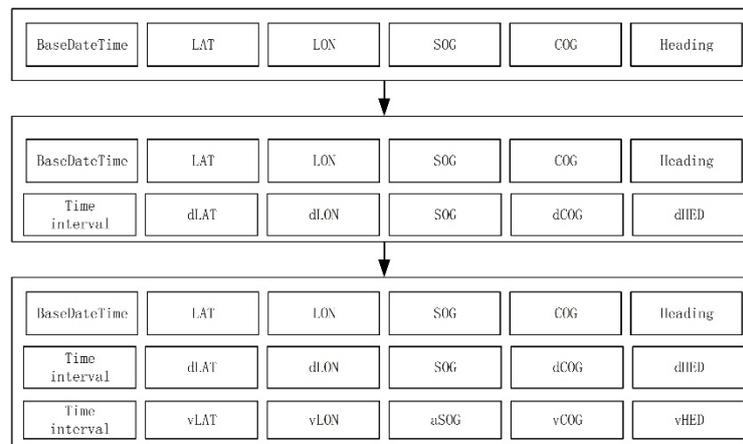
BaseDateTime	LAT	LON	SOG	COG	Heading

BaseDateTime	LAT	LON	SOG	COG	Heading
Time interval	dLAT	dLON	SOG	dCOG	dHED

BaseDateTime	LAT	LON	SOG	COG	Heading
Time interval	dLAT	dLON	SOG	dCOG	dHED
Time interval	vLAT	vLON	aSOG	vCOG	vHED

Figure 2. Elemental processing of AIS data.

After processing, the AIS data is converted from a large amount of irregular jumbled data into n sets of normalized 50×18 feature data sets for input to the model, as shown in Figure 3 below:
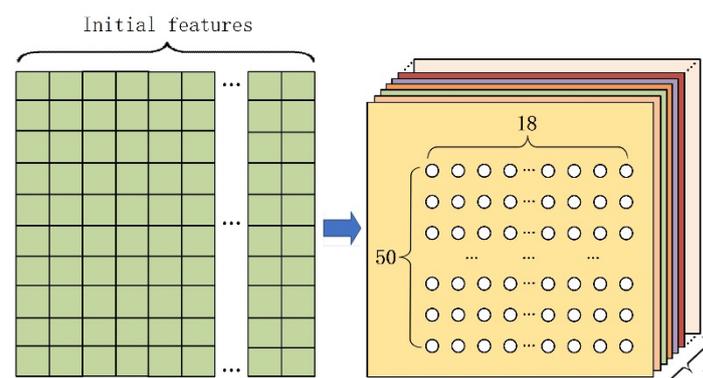
Figure 3. Comparison of AIS data before and after processing.

3.2. Optimizing Graph Convolutional Neural Network Models

The conventional graph convolutional neural network (GCN) is a deep learning model tailored for processing graph-structured data. Graph-structured data comprises a complex network of nodes and edges, where nodes depict entities and edges signify connections between entities. In GCNs, neighbor information of nodes is aggregated through convolution operations and adjacency matrices. Subsequently, an activation function is applied, allowing for repeated convolution operations to capture varying levels of information. Finally, useful feature representations are extracted. The structure and operation of GCNs enable efficient learning and representation of graph data features, making them particularly effective for image data state recognition. The conventional GCN structure, as depicted in Figure 4, serves as a reference for understanding GCNs. However, considering the lack of clear structural connections between original AIS data information, and aiming to recognize the state of the dataset, the data processing structure can be illustrated as shown in Figure 4, II:
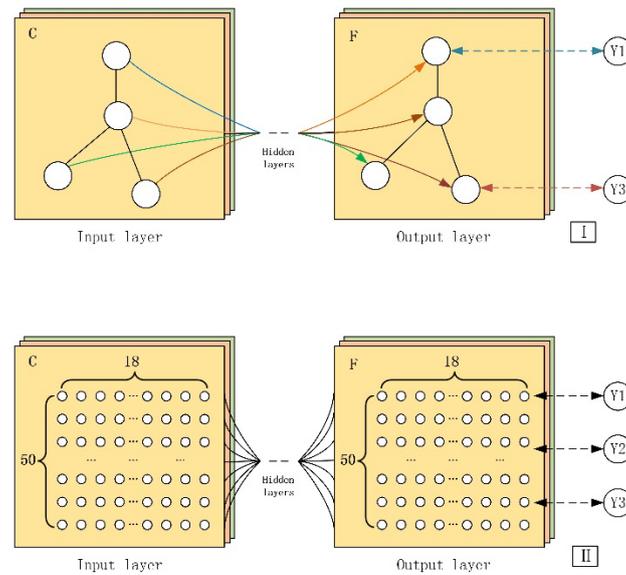
Figure 4. Graph Convolutional Structure Diagram.

In this study, we propose to use a graph convolutional neural network for state identification and feature extraction of a ship's AIS data, to convert a large amount of AIS data information with ambiguous features into clear and reliable ship state information. The experimental process of this study is shown in Figure 5 below, Figure Ⅰ organizes the acquired messy AIS data into data groups containing each feature of Figure 2. The complete data group ensures the efficiency of the experiment and the reliability of the model training. Table 1. reads each data set and plots it into trajectory points. Each state has its different characteristics but can not directly identify the different states, the need for feature extraction, and plotted latitude and longitude coordinates are just convenient for the next data selection visualization, the model is running when reading each node has multi-dimensional feature information. Figure Ⅲ In the selection of a fixed format of the data group, the experiment is 50 nodes per group, (Figure 10 data points per group is to facilitate the presentation). Figure IV and V are extracted from a set of data connecting data points, on the way to present each node containing M features (the number of M is changed according to the actual situation). Figure VI extracts the data through the graph convolutional network to extract the features and then weights them into three types of states. Figure VII shows the association of each node with other nodes, in the model all the associations are combined as a neighbor matrix, and the value of the matrix is further explored during the model training process. Figure VIII shows the convolutional feature extraction demonstration where the features are extracted from each set of data by the yellow convolutional kernel.
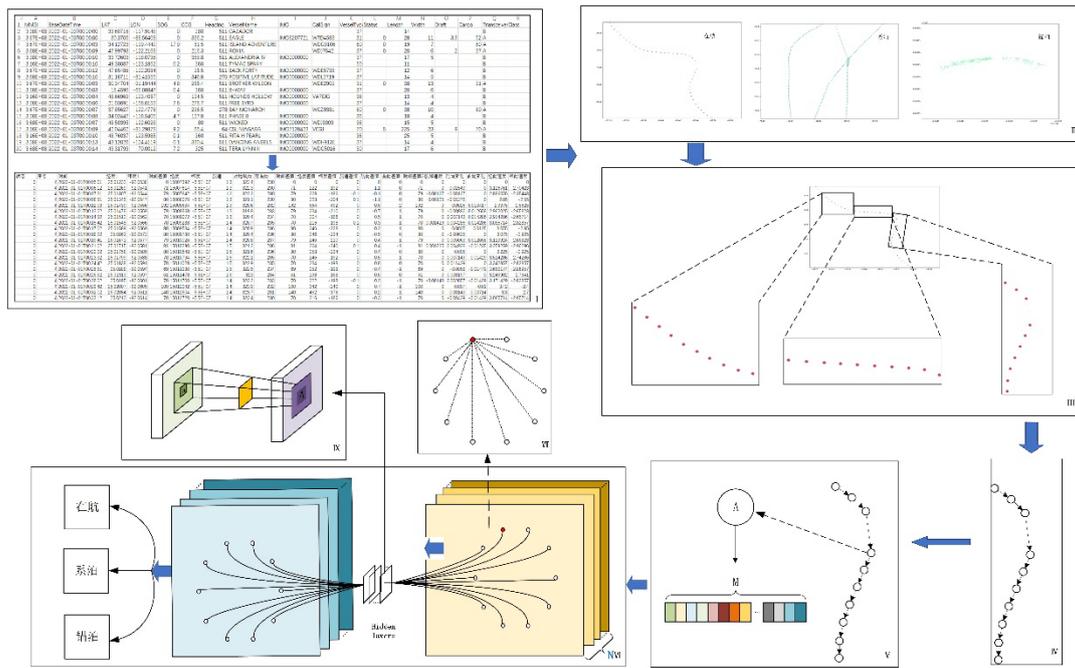
Figure 5. Model Structure Diagram.

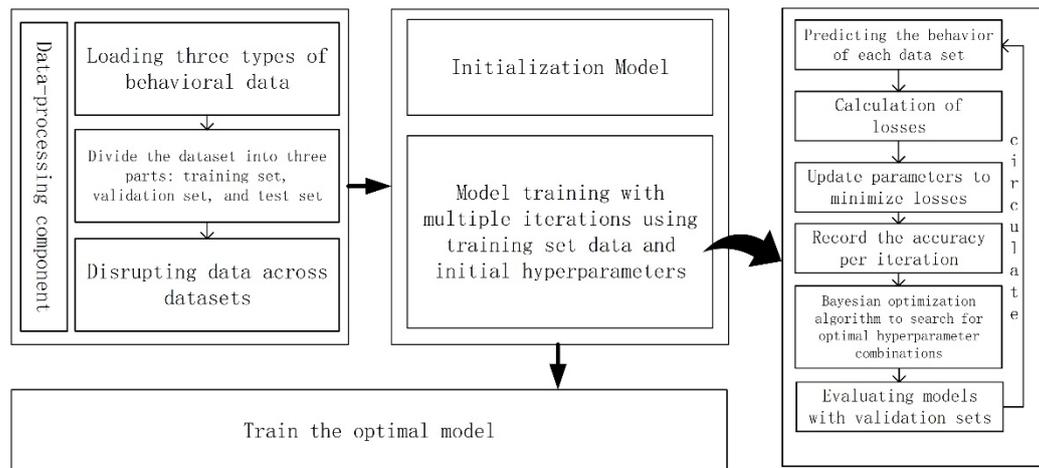The exact steps to optimize the model are shown below:

Figure 6. The main structure of the model.

3.3. Model Important Parameters

3.3.1. Setting of General Parameters

In the process of graph convolutional neural network training, there are many important parameters involved in the operation of the model, some of the regular parameters need to go through the pre-training process to select the optimal value, and the learning rate (Learning_rate) is used to optimize the model parameters of the hyperparameter, used in the model to control the step size of the parameter update, which can affect the model's convergence speed and performance. In the process of model training, the learning rate is too high and the step size of the model parameter update is too large, which may lead to the oscillation or dispersion of the loss function during the training process and make the model unable to converge. A learning rate that is too low, on the other hand, results in small step sizes for model parameter updates, making the training process slow and making it difficult for the model to escape from the local optimum point, thus affecting model

convergence. Weight_decay is a hyperparameter used to control weight decay (L2 regularization), which helps to reduce the risk of overfitting in deep learning models by penalizing larger model weights. It is set in the code as a hyperparameter for use in the Adam optimizer. When its value is small, the model penalizes the parameter size less, allowing the parameter to take larger values; a larger Weight_decay, on the other hand, causes the model to tend to choose smaller parameter values, which reduces the complexity of the model. Nconv is a hyperparameter used to control the number of convolutional layers in the GCN model, which can be adjusted according to the complexity of the problem and the size of the training data. In the model, dropout controls the probability that each neuron is randomly discarded with a certain probability during the training process, which is used to mitigate the overfitting problem. In general, when the dataset is small or the model is prone to overfitting, a larger Dropout ratio can be chosen. For larger datasets or simpler models, a smaller Dropout ratio can be chosen. Too high Dropout may also lead to too much information loss, thus affecting the performance of the model. According to the characteristics of the convolutional neural network, the range of the main parameters of the model is set as shown in Figure 8, and in the subsequent process of model operation, Bayesian optimization is used to come up with the optimal value collocation of the model.

Table 2. Setting range of values for each feature of the model.

Parameters	Setting range
Learning_rate	(0.1, 0.01, 0.001)
Weight_decay	(0.001, 0.01, 0.1)
dropout	(0.1, 0.5)

3.3.2. Innovative Parameter Analysis and Setup

In addition to the above several conventional model parameters, considering the characteristics of the AIS data and the specificity of the model, this study innovatively adds the optimization of two parameters, weights, and adjacency matrix, to optimize the model.

Each feature has a different degree of influence on the state recognition pairs, to explore the importance of various features when reading data, the weights parameter is introduced, and the weights are initialized as a one-dimensional array of length n (matching the number of features), which is used to adjust the importance of each feature in the process of extracting data features by the model. The values of the elements of the weights are optimized in the code using the gradient descent method.
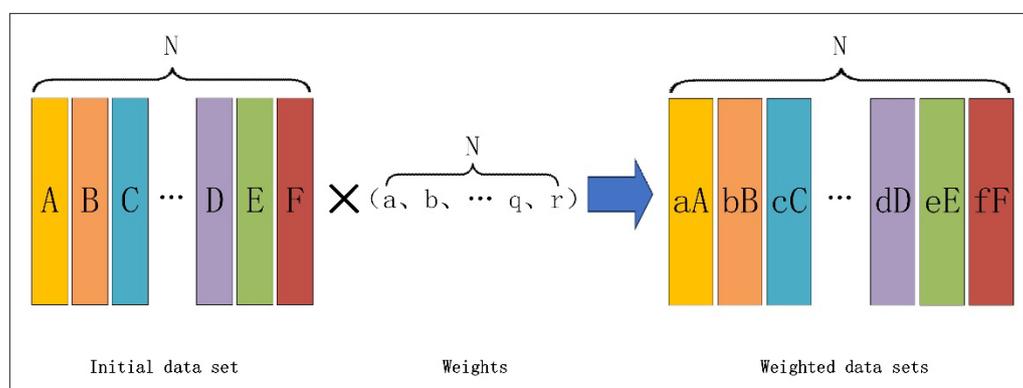

Figure 7. Schematic diagram of data set changes.

When initializing a GCN model, the adjacency matrix (Adj or A) is used to define the topology of the graph data to ensure that the model can perform graph convolution operations correctly. During model initialization, Adj is passed to the model so that the model can understand the

connection relationships between nodes. In a graph convolutional neural network, layers are propagated from layer to layer as:

$$H^{(l+1)} = \sigma(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{1}$$

In the equation, where A represents the initial adjacency matrix, and I is the identity matrix. The addition of the identity matrix allows the model to consider both inter-node relationships and the influence of individual nodes simultaneously. In this model, there is no reasonable basis for setting the values of the adjacency matrix Adj to represent relationships between features corresponding to different data rows within a data group.

To address this issue, assuming that the relationship coefficient between adjacent data points in a data group is 4, and the coefficient for points at a distance of 2 is 2, considering the self-feature in the Graph Convolutional Neural Network (GCN) where the self-relationship coefficient is set to 1, and assuming there are five data groups with consideration up to a distance of 2, the feature relationship structure is illustrated in the diagram below:
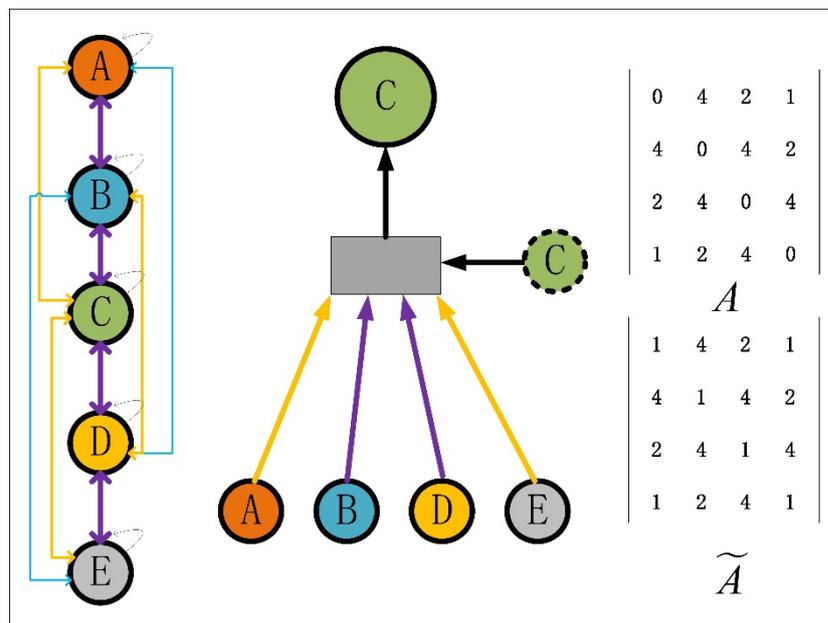

Figure 8. Schematic diagram of feature association structure.

To align with the feature extraction for 50 sets of data, this study initially initializes the adjacency matrix Adj as a 50x50 identity matrix. Subsequently, it is incorporated into the model for optimization. The optimization of Adj holds a crucial position in the new model, as each parameter's variation in the matrix corresponds to the transformation of relationships between respective rows of data. This implies that changes in the parameters consider the associations between them and can provide a clearer exploration of the AIS data group association structure through the study of Adj data features.

4. Results Analysis and Discussion

This section describes the whole process of the experiment, the first part describes the structure of the model, the second part deals with the processing of the AIS data, and the third part carries out the experimental process as well as analyzes the results of the experiment.

4.1. Experimental Procedure

This experiment collects publicly available AIS data from AccessAIS-MarineCadastre.gov from January 2021 through December 2022. To avoid experimental chance, the data of all coastal waters range of the U.S. were selected as the research object, and this study targeted the data of the three states of underway, anchored, and moored as the research object. After cleaning and organizing the AIS data, the original data of 321,485KB are processed to obtain 182,383KB of valid data that meets the model training. 5,000 sets of data for each of the three states of underway, mooring, and anchoring were obtained, with 50 data in each set, and the AIS data for each state were divided into three parts, namely, training set, validation set, and test set, in the ratio of 3:1:1. The model is trained using the training set to train the model, and the accuracy of the initially trained model on the validation set is used as a feedback to optimize the parameters for the loop.

Table 3. Distribution of data sets.

DATA SET	ACTS OF VESSELS		
	underway	moorings	anchor
TRAINING SET	3000	3000	3000
VALIDATION SET	1000	1000	1000
TEST SET	1000	1000	1000

Before the model is trained, the data from the read test and validation sets are put together and disrupted before being put into the convolutional network for training.

4.2. Analysis and Comparison Of Results

4.2.1. Analysis of Model Results

The model code is written in Python, and PyTorch is used to build and train the deep-learning model. To select the appropriate parameters for the construction of the convolutional neural network model, pre-experimentation is conducted before the formal experiments to select the main parameters of the model, pre-experimentation selects three kinds of behavior each of five hundred groups of data for training. The maximum number of iterations of the model is set to 10, 20, and 30, respectively, when training, record the time of each round of training during the experiment, the total time consumed for model training stabilization, and the highest accuracy rate under the accuracy stabilization, and record the information as shown in the following table:

Table 4. Pre-experiment 1 experimental results.

Maximum number of iterations (epochs)	10	20	30
Training time per round (s/trial)	183.4	244.21	278.01
Total time (min)	30:34	40:42	47:40
Highest accuracy	0.75253	0.78737	0.78719

Pre-experiment results show that the model works best when the number of training rounds is 20, under the premise that the model is adequately trained, the recognition accuracy is guaranteed, so the model is set to a maximum number of iterations of 20 times.

Pre-experiment 2 is set, Weight_decay is set to take the value of 0.1, 0.001, 0.001, the model is trained and the change in accuracy rate of different groups with the number of training rounds is recorded in Figure 9.
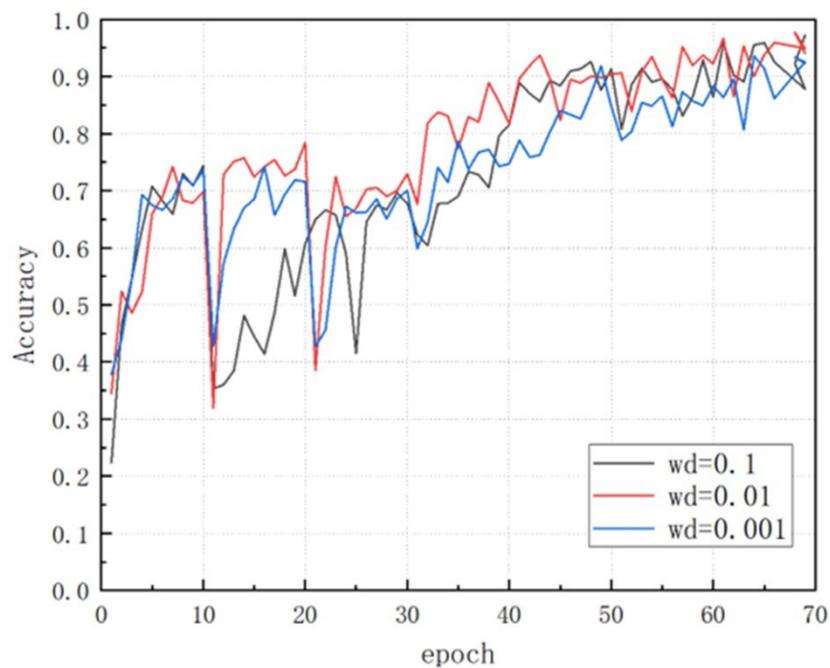
Figure 9. Pre-experiment 2 experimental results.

Pre-experiment 3 is set up and the learning rate is set to take the values of 0.1, 0.001, 0.001 and the model is trained and the change in accuracy with the number of training rounds for different groups is recorded in Figure 10.
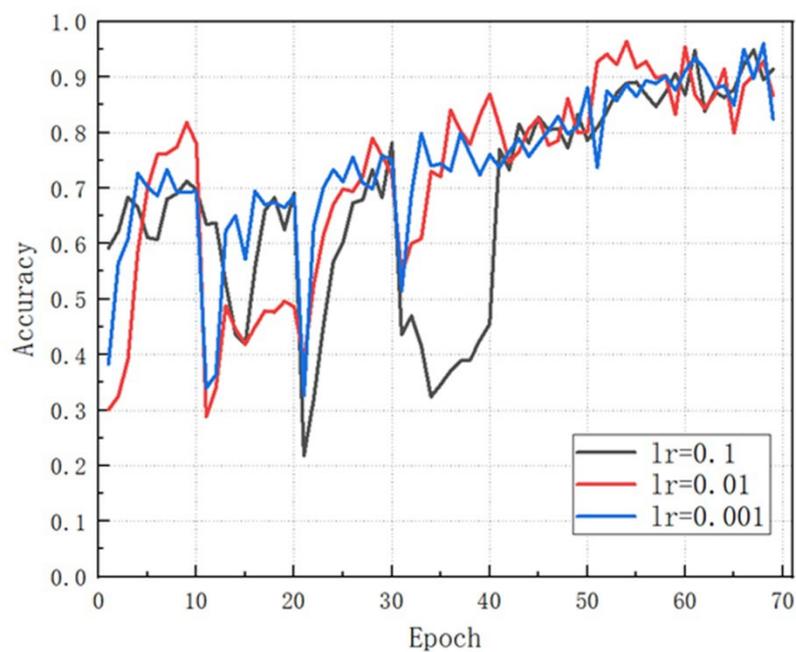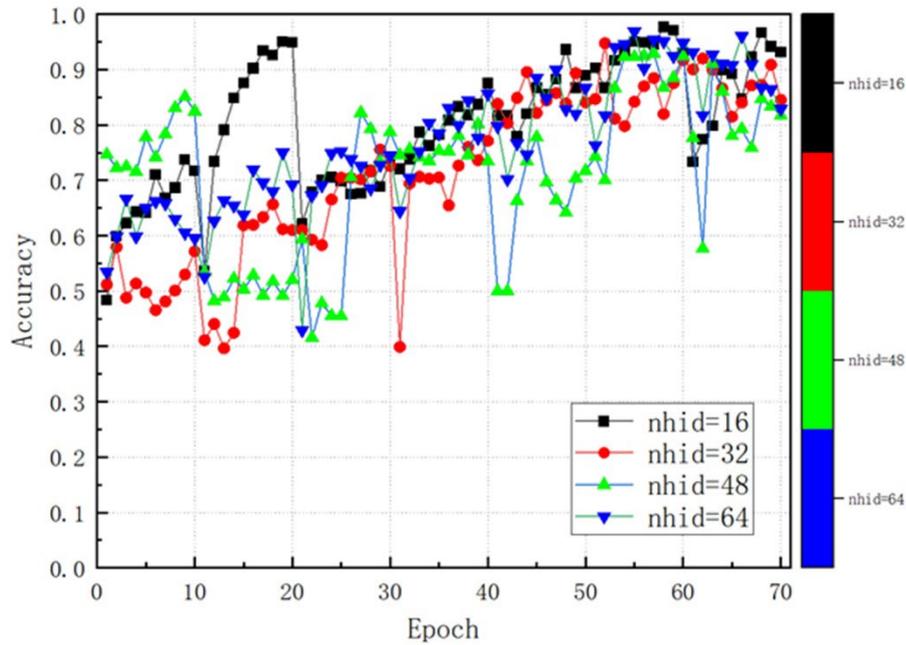

Figure 10. Pre-experiment 3 experimental results.

Pre-experiment 4 is set up, the number of hidden layers is set to take the values of 16, 32, 48, and 64, the model is trained and the change in accuracy with the number of training rounds for different groups is recorded in Figure 11.

Figure 11. Pre-experiment 4 experimental results.

Pre-experiment 5 is set up and Weight_decay is set to take the values of 3, 5, 10, and 20, the model is trained and the change in accuracy with the number of training rounds for different groups is recorded in Figure 12.
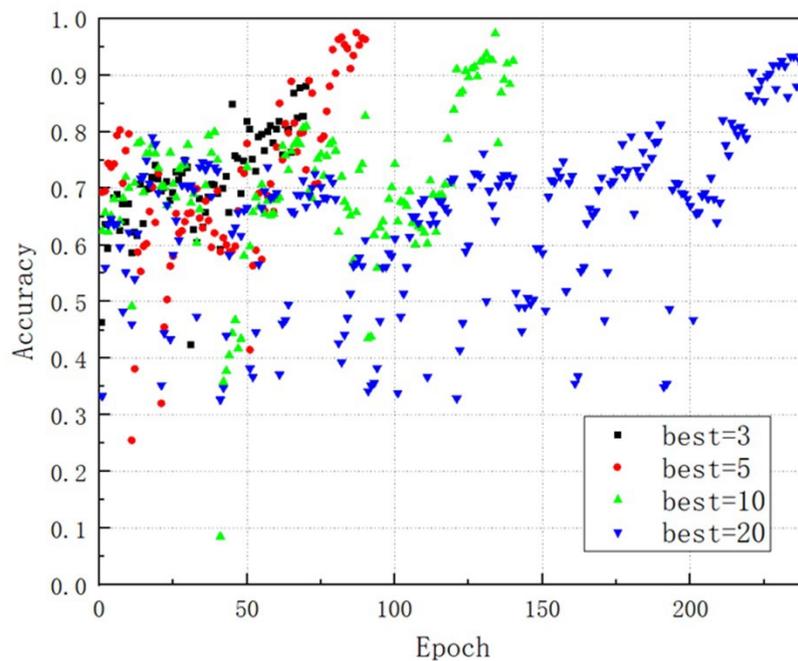

Figure 12. Pre-experiment 5 experimental results.

After the above pre-factoring, it is found that the model's performance is optimal when the maximum number of training iterations is taken to be 20, the Weight_decay setting is taken to be 0.01, the learning rate is taken to be 0.01, the dimensionality of the hidden layer is taken to be 16, and the number of Bayesian optimization rounds is taken to be 5 and 10. To ensure sufficient training and parameter search, the maximum number of rounds of Bayesian optimization is set to 10, and Dropout, Weights, and Adj are all searching for optimal values in the hyperparameter space.

After the screening and processing of the data, the data format read by the GCN model is 50 rows and 16 columns of ship AIS feature data in each group, and this study aims to explore the feature combinations that can effectively respond to the ship state when performing ship state recognition, so pre-experiments are set up to perform the feature recognition and compare the accuracy of different feature combinations. Pre-experiment selected all features, " Ⅰ ", " Ⅱ ", " Ⅲ " were tested respectively, the results are shown in Figure 13 below:
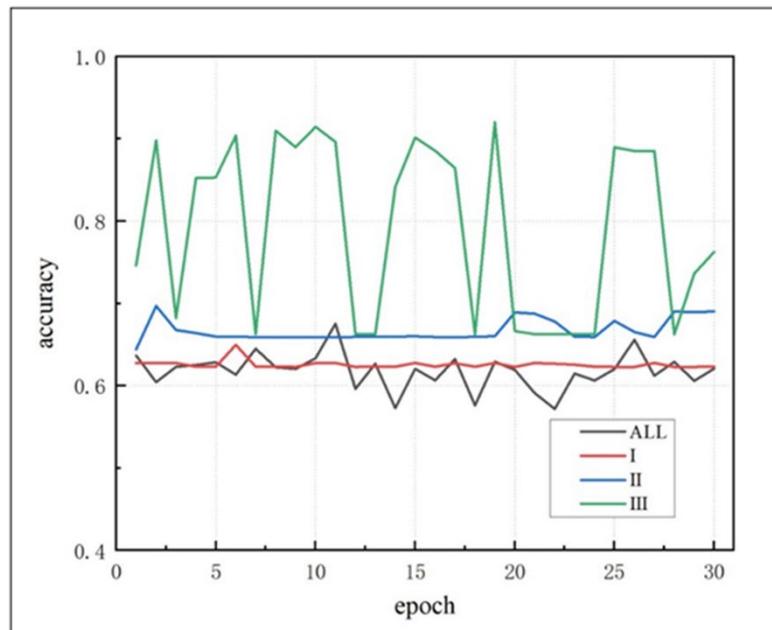

Figure 13. Model Pre-Experiment Comparison Chart.

By comparing the model optimization effect and recognition accuracy of these feature combinations, it is found that the group of feature combinations of 'longitudinal velocity', 'latitudinal velocity', 'ship acceleration', 'time difference', 'heading change', and 'bow direction change' in Figure 2. has a better performance in the training of state recognition. The reason is that this combination can effectively respond to the ship's motion characteristics as well as take into account the ship's static characteristics, which can be more effective for state model training. The search for the optimal parameters of all the features takes a longer time, and in the case of limited computing power, this six-feature combination fits the model better.

To conduct a comparison experiment, this study set up a control group, some scholars used a convolutional neural network to recognize the ship's docking state, this experiment also used a convolutional neural network for the same dataset to train the accuracy varies with the number of rounds of training as shown in Figure 14 below:
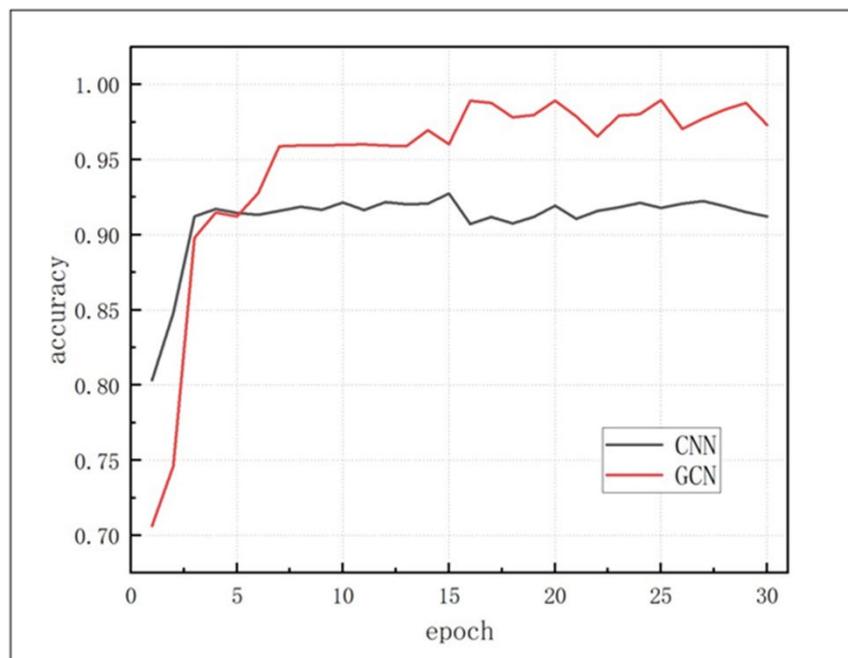
Figure 14. Modeling experimental results.

Several simulations were conducted and it was found that the model recognition accuracy was stable at around 98.3%, with the model taking around 523.43 seconds per round of training, and the whole training being carried out in 17 minutes and 30 seconds. The model trained using a convolutional neural network was able to achieve 89.9% recognition accuracy after the training was stabilized, the model took about 797.07 seconds per round of training, and the whole training was conducted in 26 minutes and 35 seconds. The model has high recognition accuracy after optimizing both Adj and weights, and the recognition structure against other groups proves that the optimized model has better performance in state recognition.

4.3.2. Evaluation of Training Results

The chapter conducts a comprehensive experimental process, beginning with the processing of AIS data to obtain a complete and reliable AIS data segment. A preliminary experiment is conducted, wherein four experimental groups are established based on the number of features. Different combinations of feature data groups are utilized for state recognition, revealing that the combination of features corresponding to Group III is more responsive to the comprehensive characteristics of the data. This suggests that dynamic features are more pertinent for ship state recognition in this experiment. Conversely, the group containing all features performs poorly, indicating inefficiency in extracting feature data when the model is overloaded with feature data.

Building upon these findings, a complete experiment is conducted employing a convolutional neural network for training. The results demonstrate that the recognition accuracy of the group optimizing both Adj and weights can reach 98.3%, underscoring the graph convolutional neural network's capability to achieve high accuracy in recognition through optimized association between elements. In comparison, the same data is subjected to state recognition using a conventional convolutional neural network, achieving a recognition accuracy of 89.9%. This highlights the superior experimental outcomes of the graph convolutional neural network when considering the characteristics of graph data.

Upon completion of model training, the model is utilized for state recognition of AIS data, extracting specific state information of ships from complex AIS data segments. In segments where ships are correctly recognized, the extracted state information can offer reliable sailing advice to incoming and outgoing ships. Furthermore, by categorizing ship state information based on the

model's recognition, AIS data of ship sailing areas can be transformed into an effective and reliable state information library. This categorized state information can aid in effectively dividing areas into underway channel areas, anchorage areas, and mooring areas, providing valuable navigation advice for navigating ships and enhancing marine traffic safety.

In unfamiliar sea areas, ships can conduct navigation planning based on passing ships' sailing trajectories, paying attention to ships in anchorage areas, and planning navigation accordingly. The graph convolutional neural network demonstrates strong data mining ability for graph-structured data, achieving high accuracy in recognizing AIS data trajectories in a fixed format. This approach is particularly applicable in waters with abundant historical AIS data, offering significant implications for ship decision-making and path planning.

4.3.3. Discussion

In this study, an optimized algorithm based on the graph convolutional neural network is proposed to identify the shipping state. Leveraging the capability of the graph convolutional neural network to consider the connection between each data point, the proposed algorithm achieves an 8.4% improvement in accuracy compared to the traditional convolutional neural network algorithm. This heightened accuracy enables more informed sailing suggestions for decision-makers. Moreover, the model can be seamlessly integrated into ship facilities, enabling real-time identification of shipping status based on acquired AIS data, thereby providing a basis for relevant departments to respond and make decisions promptly.

Regarding data processing, the model calculates original AIS data through time intervals and neighboring data differences to derive various combinations of data features. A pre-experiment is conducted to compare the recognition accuracy results of different combinations. The findings indicate that the combination of features corresponding to Group III demonstrates superior recognition accuracy, underscoring the effectiveness of feature information changes in state recognition. Conversely, the inferior performance of the group containing all features reflects that an excessive number of features diminishes the optimization efficiency of the model during graph convolutional network training.

In terms of model construction, this study innovatively employs a graph convolutional neural network for ship state recognition. Ship trajectory data typically entails spatial coordinates, making GCNs well-suited to capture spatial relationships in the graph structure. Each node represents a location point, while the edges represent spatial relationships between these locations, facilitating effective learning of spatial patterns in the trajectory data. Moreover, the GCN can flexibly adapt to different trajectory data samples and time intervals, efficiently transferring information through the graph structure crucial for interactions between position points in the trajectory data.

Through multi-layer graph convolution, the model captures relationships between global and local aspects, enhancing understanding of the dynamic characteristics of trajectory data. Additionally, the model adeptly accommodates dynamic changes in ship trajectory data, such as changes in speed and direction, facilitating real-time state recognition.

Compared to conventional convolutional neural networks, the introduction of weights and adjacency matrices allows the model to explore associations between each data point within a group more comprehensively. When utilizing Bayesian optimization parameters, this enhanced model structure can more flexibly adapt to different data structures, thereby optimizing accuracy and robustness. Empirical evidence demonstrates that the model achieves high accuracy after introducing adjacency matrix optimization.

5. Conclusions

In this study, the acquired AIS data undergo initial cleaning and screening to select stable navigation data segments, ensuring the reliability of the experimental data sources. Subsequently, a graphical convolutional neural network (GCN) model is constructed to establish a correlation graph of AIS data in a digital format based on intra-group connections. To enhance the accuracy of the model, optimization of the weights of each feature and the adjacency matrix is introduced. This

enables the model to better explore the correlation among the features of the AIS data and efficiently identify the shipping state in a targeted manner.

The model provides real-time ship status and state information based on known AIS data segments, offering navigation decision-makers more reliable reference information. However, the model imposes higher requirements on the fixed format of AIS data. Subsequent research should focus on optimizing the model to reduce its reliance on specific data formats and better adapt to changing maritime navigation conditions. Enhancing the model's real-time monitoring function to instantly obtain the status of other ships can contribute to building a more comprehensive maritime transportation network.

This study proposes a ship state recognition method based on AIS data and GCN, aiming to improve maritime traffic management and ship navigation safety. By effectively processing a large amount of AIS data, the model accurately recognizes states such as sailing, anchoring, and mooring, providing regulators with a valuable tool to manage and maintain maritime traffic order.

The innovation of this study lies in transforming error-prone AIS data into useful ship state information, which is valuable for decision-makers and planners. However, the study has limitations. The model performs well with complete data segments but struggles with fragmented and mutilated AIS data. Moreover, it focuses on individual vessel states without fully considering navigation conditions and various influencing factors. Future research should address these limitations by focusing on data restoration to reduce AIS data requirements and building scene-aware models that consider different maritime traffic scenarios and environmental factors. Real-time monitoring and decision support integration and international data sharing and cooperation are also crucial for improving maritime traffic management and safety on a global scale.

In conclusion, continuous model improvements and expanded research areas will enhance maritime traffic management, improve navigation safety, and contribute to the construction of future maritime traffic networks. This will positively impact global maritime traffic and navigator safety.

References

1. Xiao Z.; Fu X.; Zhang L. et al. Traffic Pattern Mining and Forecasting Technologies in Maritime Traffic Service Networks: A Comprehensive Survey. IEEE Transactions on Intelligent Transportation Systems 2020, 21(5):1796-1825.
2. Fernandez Arguedas V.; Pallotta G.; Vespe M. Maritime Traffic Networks: From Historical Positioning Data to Unsupervised Maritime Traffic Monitoring. IEEE Transactions on Intelligent Transportation Systems 2018, 19(3):722-732.
3. Filipiak D.; Węcel K.; Stróżyna M. et al. Extracting Maritime Traffic Networks from Ais Data Using Evolutionary Algorithm. Business & Information Systems Engineering 2020, 62(5):435-450.
4. Liu Z.; Gao H.; Zhang M. et al. A Data Mining Method to Extract Traffic Network for Maritime Transport Management. Ocean & Coastal Management 2023, 239.
5. Lei P.R.; Tsai T.H.; Wen Y.T. et al. A Framework for Discovering Maritime Traffic Conflict from Ais Network. Asia-Pacific Network Operations and Management Symposium (APNOMS) 2017,2017: 1-6.

6. Chen X.; Qi L.; Yang Y. et al. Video-Based Detection Infrastructure Enhancement for Automated Ship Recognition and Behavior Analysis. Journal of Advanced Transportation 2020, 2020:1-12.
7. Pan X.; He Y.; Wang H. et al. Mining Regular Behaviors Based on Multidimensional Trajectories. Expert Systems with Applications 2016, 66:106-113.
8. Suo Y.; Ji Y.; Zhang Z. et al. A Formal and Visual Data-Mining Model for Complex Ship Behaviors and Patterns. Sensors (Basel) 2022, 22(14).
9. Zhou Y.; Daamen W.; Vellinga T. et al. Review of Maritime Traffic Models from Vessel Behavior Modeling Perspective. Transportation Research Part C: Emerging Technologies 2019, 105:323-345.
10. Mylavarapu S.; Sandhu M.; Vijayan P. et al. Towards Accurate Vehicle Behaviour Classification with Multi-Relational Graph Convolutional Networks. 2020 IEEE Intelligent Vehicles Symposium (IV) 2020, 2020: 321-327.
11. Gao M.; Shi G.-Y. Ship-Handling Behavior Pattern Recognition Using Ais Sub-Trajectory Clustering Analysis Based on the T-Sne and Spectral Clustering Algorithms. Ocean Engineering 2020, 205.
12. Wang S.; Li Y.; Xing H. A Novel Method for Ship Trajectory Prediction in Complex Scenarios Based on Spatio-Temporal Features Extraction of Ais Data. Ocean Engineering 2023, 281.
13. Mao S.; Tu E.; Zhang G. et al. An Automatic Identification System (Ais) Database for Maritime Trajectory Prediction and Data Mining. Proceedings of ELM-2016: 2018// 2018; Cham: Springer International Publishing 2018,2018: 241-257.
14. Xu N.; Wang P.; Chen L. et al. Mr-Gnn: Multi-Resolution and Dual Graph Neural Network for Predicting Structured Entity Interactions. International Joint Conference on Artificial Intelligence 2019, 2019.
15. Zhang H.; Lu G.; Zhan M. et al. Semi-Supervised Classification of Graph Convolutional Networks with Laplacian Rank Constraints. Neural Processing Letters 2021, 54(4):2645-2656.
16. Liu R.W.; Liang M.; Nie J. et al. Stmgcn: Mobile Edge Computing-Empowered Vessel Trajectory Prediction Using Spatio-Temporal Multigraph Convolutional Network. IEEE Transactions on Industrial Informatics 2022, 18(11):7977-7987.
17. Zhao J.; Yan Z.; Chen X. et al. K-Gcn-Lstm: A K-Hop Graph Convolutional Network and Long–Short-Term Memory for Ship Speed Prediction. Physica A: Statistical Mechanics and its Applications 2022, 606.
18. Wang S.; Li Y.; Xing H. et al. Vessel Trajectory Prediction Based on Spatio-Temporal Graph Convolutional Network for Complex and Crowded Sea Areas. Ocean Engineering 2024, 298.