THE DESTINATION PORT PREDICTION FOR TRAMP SHIPS BASED ON AIS TRAJECTORY DATA MINING: A CASE STUDY OF VLCC

CHEN DONG
MPhil

The Hong Kong Polytechnic University

2022

The Hong Kong Polytechnic University

Department of Logistics and Maritime Studies

The Destination Port Prediction for Tramp Ships Based on AIS Trajectory Data Mining: A Case Study of VLCC

CHEN Dong

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Philosophy

April 2022

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

_____ (Signed)

_____CHEN Dong_____(Name of student)

Abstract

Tramp shipping accounts for more than 75% of the total tonnages of ships in the market. Different from the liner shipping, tramp shipping has no fixed route, schedule, and destination port. These characteristics lead to the supply and demand imbalance that is recognized as the spatial-temporal heterogeneity problem in transportation. Destination port prediction is fundamental and significant to solve this problem. Even though AIS (Automatic Identification System) can provide the destination port information, about 70% of the information is wrong. Hence, the vessel trajectory-based method has risen to prominence, which is available for any stage of sailing.

Port calls are important to extract the voyages and semantic information. To recognize port calls rapidly and correctly, we develop an optimized CB-SMoT algorithm with less time complexity. Compared with other algorithms, our algorithm can correctly identify 84.6% of port calls for bulk carriers and 90.63% for tanker ships.

Grounded on the identified port calls of VLCCs, we construct a framework of three models as follows:

Model 1 is a high order sequence of port calls model. The definition of order is the number of previous ports. We build different order sequences with port names as semantic information. Then we train the random forest (RF) classifier with the feature X of high order sequences and the label Y of destination port. The accuracy increases with the growth of order, which means richer previous ports information is beneficial for destination classification. When the order is larger than 3, the accuracy can reach 0.80 and above.

Model 2 is a trajectory similarity model. We adopt the TRACLUS algorithm to produce representative trajectories. The representative trajectory is an extracted standard trajectory for a route but does not really exist. We calculate the similarity between the sailing trajectory and representative trajectories by SSPD (Symmetrized Segment-Path Distance) and convert them to probability matrices as semantic information. In our model, the similarity probability matrix, IMO number and DWT

(Deadweight tonnage) can be the features X and the destination port can be the label Y. We train the tree-based models and find GBDT (Gradient Boosting Decision Tree) achieves the best performance. The accuracy increases along with days and exceeds 0.70 after 25 days. We also find the common segments of sailing trajectories has a negative effect on the prediction.

Model 3 is a neural network model. We predict voyages for three frequent routes, respectively. The results show LSTM (Long Short-Term Memory) has the minimum RMSE (root mean squared error) of longitude and latitude for the predicted last few days' trajectory. When the time length for prediction is shorter, the correct rate is higher. The correct rate can reach 0.83 by predicting the last-48h trajectory. The reliable results can be provided two days in advance before arriving.

Our work provides an innovative integrated framework of models for destination port prediction covering different stages of a voyage and gives the application guidelines of these models. In the future, based on our study, the routing optimization problem for tramp shipping will be studied.

Keywords: Destination port prediction; Tramp shipping; Port calls recognition; VLCC trajectory data mining; High order sequence of port calls; Trajectory similarity; Neural networks

Acknowledgements

My journey in The Hong Kong Polytechnic University brings me many precious memories. During the past one and a half years, the Department of Logistics and Maritime Studies (LMS) gave me a lot of help, and countless people supported my efforts on this study. I would like to express my gratitude and appreciation to all of them.

The first and biggest appreciation goes to my supervisor Dr. YANG Dong. His brilliant, great feedback and guidance enriched my study throughout the learning period. I greatly appreciate him for providing opportunities to make me grow professionally. The project experiences are invaluable for my future research.

My sincere thanks also go to the committee members: Prof. Zhuo Sun, Prof. Paul Tae-Woo LEE and the BoE chair Prof. Luo Meifeng, for their participation in my oral defense and constructive comments.

Very special thanks go to all the academic staff, research students, and administrative staff in business school.

Last, but not least, my family deserves endless gratitude: wherever and whenever my mother encouraged me, and my father told me to be brave and persistent. They forever can be my strong backing.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Research Background

Shipping has been the most economical transportation mode in international trade. The International Marine Organization (IMO) reports more than 90% of commodities and goods in the world are transported by shipping (Sirimanne et al., 2019). As the growth of the global economy, the tramp ship capacity in the market increases rapidly. According to the statistics of the United Nations Conference on Trade and Development (UNCTAD), the tonnage of tramp ships accounts for about 75% of the total tonnages of all ship types. Tramp shipping business plays an important role in transporting cargo such as coal, crude oil and iron ore.

Different from the liner shipping, in tramp shipping, the cargo owners are concentrated while the carriers are scattered. The route mode of tramp shipping is a point-to-point mode. Hence, tramp shipping has no fixed route, schedule, and destination port. Moreover, the sailing time under heavy ballast condition accounts for a high portion in tramp shipping. Taking the Capesize dry bulk carriers as the example, in iron ore trade, the time ratio of full loaded status is only 48.8% for China's flag fleet and 52% for Panama’s flag fleet. For shipping companies, the cost for operation management and the loss of profits are huge. In recent years, increasing researchers pay attention to this phenomenon.

As most studies show, imbalance of supply and demand exists in the industry of tramp shipping. The tramp shipping market can be regarded as an unstable market. Volatility is the most important characteristic for tramp shipping. However, the tramp

ships' behaviors under the volatility are rarely researched. Because, in the real world, the spot market information is opaque and is hard to collect. The shipowners usually assign the ships to move toward the port with potential demands in a limited time window, aiming to conclude a transportation contract before other competitors. But the shipowners cannot know how many tramp ships will arrive at the same port. As a result, many tramp ships wait for loading in one port, but few tramp ships to load cargo in another port. The distribution of tramp ships is not homogeneous for the market. We call this phenomenon a spatial-temporal heterogeneity problem in tramp shipping.

To solve this problem, the significant and fundamental work is the destination port prediction. How to predict the destination port accurately becomes the cornerstone to build the operations research model or the optimization model. As the development of Automatic Identification System (AIS), the destination port information can be provided directly. However, only 30% of the destination information in AIS is credible. Because the destination port item is submitted by shipowners. Some shipowners intend to hide the real information to take advantage of the market, while some shipowners cannot decide which port to load cargo.

Even though most destination port information is wrong, the AIS system still brings the opportunity to mine the vessel trajectories. Based on these trajectories, the destination port can be predicted. One popular method is combining vessel trajectories with the machine learning models.

Besides, VLCCs make up more than 60% of the crude oil transportation and contribute most of the profit elasticity for shipping companies. The ocean routes of

VLCCs are simple and the variations of destinations are limited. Therefore, the VLCC fleet can be a good research objective to ensure the feasibility of our study. It is important to develop a comprehensive framework of different models to predict the VLCCs' destination ports. This framework can be extended and used for other tramp ship types in the future.

1.2 Research Status of Related Work

1.2.1 AIS data application

AIS is a kind of self-reporting surveillance system and originally designed for traffic services to collect vessel information at a high frequency. Any vessel over 300 GT needs to be equipped with AIS on board (Jia et al.,2019). AIS data mainly records the static information and dynamic information, including MMSI/IMO, timestamp, speed, draft, ship's heading, ship's name, longitude, latitude etc. AIS data has important research values and has been widely employed in maritime studies.

For vessel trajectory, the path planning is one of the popular research topics. A suitable vessel path is important for shipowners to save fuel cost and keep safe. Different from the path planning for vehicles, sea weathers and speed variations make this work even more complicated for ships. Wen et al. (2014) designed a Route Miner System with AIS data to detect the movement behaviors in a free space and output a group of ship routes for planning. Zaccone and Martelli (2018) proposed a random sampling algorithm to avoid the obstacle in ship's path. Yu et al. (2021) developed an A* algorithm for unmanned surface vehicles (USVs) to plan the routes and avoid other dynamic ship movements.

1.2.2 Destination prediction

(1) Vessel destination prediction

The designation information is hard to know because the vessel may change its target port during any stage of sailing. Vessel destination prediction is a kind of decision support tool to capture dynamic information. It is important for shipping companies to manage the operation and change the schedule, although the uncertainty of vessel destination is high, and the trajectory is complex.

In recent years, many researchers have studied vessel destination prediction by data-driven methods. Xu et al. (2012) designed a neural network of three layers to track the vessel trajectory. Mao et al. (2018) proposed an Extreme Learning Machine (ELM) to predict the vessel path and find the possible destination. Magnussen (2021) developed a Recurrent Neural Network (RNN) with the graph abstraction to predict global destinations for oil tankers.

(2) Travel destination prediction in transportation

In the non-vessel field, many destination prediction studies are about the vehicle, flight, and pedestrian. For vehicles, Tiesyte and Jensen (2008) constructed a Nearest-Neighbor Trajectory (NNT) technique to search destinations with history trajectories. Chen et al. (2010) proposed a Continuous Route Pattern Mining algorithm to build decision trees of destinations and predict movement patterns. For flight, Lin et al. (2018) developed a hidden Markov model to predict relative motion between positions. Chen et al. (2020) proposed a deep Gaussian process model to predict the temporal

correlations among flight positions. For pedestrians, Yi et al. (2016) designed a neural network (Behavior-CNN) to predict pedestrian destinations under crowded scenarios. Zhang et al. (2019) developed a Gaussian mixture model to detect stopover points and a BiGRU-based model to predict the pedestrian trajectory.

Besides, Morzy (2007) proposed a Traj-PrefixSpan algorithm to mine the frequent movement patterns to predict the destination. Prabhala and La (2015) developed an algorithm called PeriodicaS to find the periodicity in users' mobility traces and marked the periodicity with explicit semantic annotations to classify the destinations.

1.2.3 Trajectory models

Spatial-temporal data are the base for modeling the trajectory. These data have the characteristics of multi-scale and multi-time. Langran and Chrisman (1988) reviewed previous research results to conclude a series of spatial-temporal data models, such as the simple timestamp model. This work marks the beginning of systematic modeling in Geographic Information System (GIS). In the subsequent studies, Peuquet (2001) developed a model using time series events. Worboys and Hornsby (2004) proposed a dynamic geospatial domain model based on the event and object. Pelekis et al. (2004) incorporated the spatial, temporal, and semantic information into a model to build the fields-based tree model.

The trajectory model consists of various spatial-temporal data. The first trajectory model was designed to estimate the movements of hurricanes by Reap (1972). Parent et al. (1999) extended the traditional model such as the MADS model. Spaccapietra et al. (2008) merged the stop-move information of special events into the trajectory model.

Ying et al (2010) proposed a semantic model to find the relationship between different trajectories. Kwakye (2019) systematically summarized the development process of trajectory models and designed a synthetical and general platform to query and analyze the trajectory information.

Moreover, to measure the trajectory similarity in the model, different distance concepts are proposed, such as the Fréchet distance. Naderivesal et al. (2019) developed a new similarity measurement using the interval distance to capture the uncertainty. These similarity measurements provide the opportunity to cluster the trajectories. TRACLUS algorithm and T-OPTICS algorithm are the typical clustering approaches.

1.3 Research Motivation

The destination port prediction for tramp ships is the sub-project of the spatial-temporal heterogeneity problem that has not been researched well. In previous studies, in terms of the volatility of tramp shipping, scholars have not researched the short-term behaviors of tramp ships (one or two months) (Hennig et al., 2012; Wen et al.2016). Most papers focus on the macro freight rate (Tvedt, 2011; Yin et al., 2017), lacking the micro perspective. Some researchers consider the discrete choice models a possible way. But the discrete choice models have many limitations: Some variables such as the shipowner preference and the shipowner decision are difficult to quantify and collect; the endogeneity problem is hard to handle with; the performance of prediction is poor. Although recently the AIS big data can bring the opportunity for researchers to adopt the data-driven approaches to improve the prediction performance, it is still a gap to

develop the systemic prediction framework using these approaches that cover different stages of the sailing.

Therefore, in this study, we make the first attempt to conduct an in-depth investigation on the destination port prediction for tramp ships. We propose an integrated framework with multi-models of machine learning to predict the destination port under different stages of the sailing for VLCCs. The associated algorithm and application are also researched.

1.4 Research Content

Our goal is the destination port prediction for tramp ships in a dynamic process. The dynamic process includes three stages of a voyage: the stage of before sailing, the stage of during sailing, and the stage of before arriving. It is the fundamental and significant work to fix the spatial-temporal heterogeneity problem. However, the destination prediction task faces several difficulties:

- High uncertainty of the voyages.
- A long travel time for VLCCs.
- Common trajectory segments in different routes.

In this study, we use data mining techniques to search the hidden information of vessel trajectories. Based on the AIS data, we detect the stay regions and routes of VLCCs. Combining machine learning and trajectory information, we construct a destination prediction framework of three models. These models can predict the destination port in different stages of sailing. The high order sequence of port calls model is used before

sailing; the trajectory similarity model is used during sailing; the neural network model is used before arriving. All the models can be extended to other tramp ship types for prediction, such as bulk carriers.

Our research contents are described as follows:

(1) To begin with, we use the speed-based heuristic filtering method to clean the raw AIS data and use the Douglas–Peucker algorithm to compress the data. These preprocessing procedures can ensure the quality of our database and reduce the time cost of programs.

(2) Then, combined with the world ports list, we develop an optimized CB-SMoT algorithm to recognize the port calls information rapidly and accurately. Every cluster of stay points is unique to represent a port of call. This algorithm constructs the basis to provide data materials for our models. The sequences of port calls and the voyages with trajectories of VLCCs are extracted.

(3) Finally, based on the port calls information and the trajectories of VLCCs, we design a three models-based framework for destination port prediction. The model 1 is a high order sequence of port calls model, using the previous ports to predict the destination before the voyage. The model 2 is a trajectory similarity model for prediction during sailing. The effects of the common segments of sailing trajectories are discussed. The model 3 is a neural network model to predict the last few days trajectory, which is reliable to identify the destination port with two days in advance before arriving. Besides, the application of these models is introduced with a

detailed example.

According to our study, the framework to predict the destination ports of VLCCs can be modeled as the workflow shown in Figure 1.1.

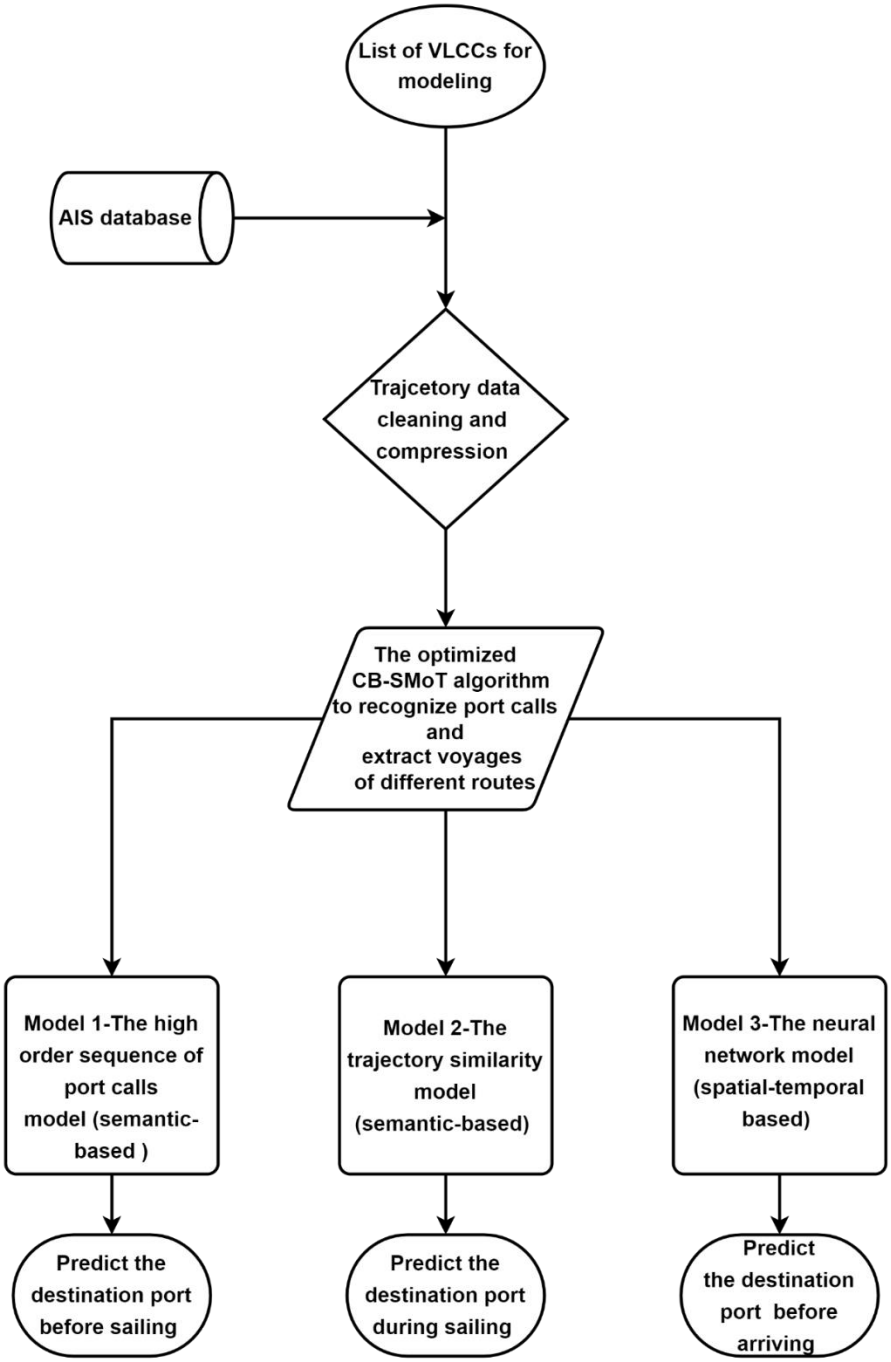


Figure 1. 1 The framework flowchart for destination port prediction

The chapter structures are organized as follows. Chapter 2 overviews the literature and states the ideas for research. Chapter 3 conducts the data preprocessing and elaborates on how to develop a new optimized algorithm to extract the sequence of port calls. Chapter 4 builds the framework of three models for prediction. The performances of models are analyzed, and the results are discussed. A guideline of application is explained. At last, Chapter 5 makes the conclusion and gives the suggestions for future work.

1.5 Summary of the Chapter

In this chapter, we introduce the research background and research motivation of our study in detail. We list some difficulties and state the research content to show what the logical structures of this study are.

Chapter 2: Literature Review

In this chapter, we review the literature associated with our research and explain the inspiration source of our ideas. Three parts of different literature comprise this chapter. These parts are introduced according to the logical order of research. In the first part, in terms of the overall goal, we review the papers about vessel trajectory prediction and propose the primary frameworks for our study. The primary frameworks include spatial-temporal trajectory framework and semantic trajectory framework. AIS data can provide the spatial-temporal trajectory directly for machine learning models, but the semantic trajectory needs to be processed. Therefore, we review the papers related to the semantic trajectory construction in the next two parts. In the second part, we review the papers of stay points recognition. Stay points recognition is important to reflect the port calls of a vessel that is the key semantic information we can get before sailing. Besides, the port calls from stay points recognition can help extract the voyages and associated trajectories for above frameworks. In the third part, we review the papers about distances for trajectory similarity measures. The suitable distance theory can tell the semantic information of similarities between different trajectories during sailing.

2.1 Vessel Trajectory Prediction

Currently, the methodology for the vessel trajectory prediction can be divided into two kinds of the frameworks: spatial-temporal trajectory mining and semantic trajectory mining.

2.1.1 spatial-temporal trajectory mining

(1) Statistical approach

Most theoretical models for the vessel trajectory prediction have been developed with statistical approaches. These models adopt history trajectory data to calculate the likelihood distribution of vessel locations for prediction. Combining the ship velocity and its timestamp with the statistical likelihood, the model can search the space range in the neighborhood area of the ship to predict the future trajectory. But the accuracy of statistical approach is low in some scenarios due to the mean distribution of probability (Sun and Zhou, 2017; Murray and Perera, 2018; Üney et al., 2019).

(2) Machine learning approach

The development of machine learning provides the opportunity for trajectory prediction. Kalman filter (KF) is one kind of traditional machine learning approach. KF constructs the equation of ship movement state by comparing the real trajectory with the predicted trajectory. Then KF can predict the vessel trajectory from the previous position (Peng et al., 2010; Perera and Soares, 2010; Xu et al., 2014). Another traditional machine learning approach is the Markov model. Markov model builds the state transition matrix for ship movements and uses this matrix to predict the vessel trajectory at the next moment (Qiao et al., 2014; Guo et al., 2018). However, KF and Markov approaches can only obtain the local optimal solution for short-term prediction. To ensure the accuracy, the timeframe of the predicted trajectory is less than 1 hour.

The neural network is another important branch of machine learning and has become popular in recent years, represented by the back propagation neural network (BP neural network), Gate Recurrent Unit (GRU) and Long Short-Term Memory (LSTM). These models are trained to achieve a high accuracy with historical trajectory data of a vessel. Then they can predict the vessel's trajectory for a long-term of several

days (Xu., 2011; Gao et al., 2018; Zhou et al., 2019; Tang et al., 2019). But when the shape of the trajectory is complex, the accuracy of the predicted vessel trajectory is limited. Moreover, the cost of adjusting parameters is huge.

(3) Grid-based approach

To monitor and predict the variation of a vessel trajectory in different days, grid-based approaches are proposed by many researchers. The route extraction and the motion pattern analysis represent one kind of grid-based approaches (Vespe et al., 2012; Pallotta et al., 2013; Liu and Chen, 2014). This approach requires the isolation of sea lanes by gridding to collect trajectories. It predicts the vessel trajectory by matching the closest known trajectory. Then use a "vectoral" approach to analyze the motion pattern for anomaly detection. In most studies, grid-based approaches focus on a specific geographical area to predict the target ship's trajectory before arriving the destination port in 48h.

Table 2.1 summarizes previous studies by the spatial-temporal trajectory mining. In this thesis, considering VLCCs often sail more than twenty days on their voyages, we try to use the BP neural network, LSTM and GRU in machine learning to predict the spatial-temporal trajectories of the last few days. Besides, we match the final locations of predicted trajectories with a VLCC ports list to estimate the destination ports.

Table 2. 1 Previous studies on vessel spatial-temporal trajectory mining

Source	Content	Methodology
Üney et al., 2019	Classify the vessel location and speed into different categories and calculated the likelihood distribution for future trajectory.	Statistical probability
Sun and Zhou, 2017	Divide the regions and marked the probability of vessel appearance to predict the positions.	Statistical probability
Xu et al., 2014	Develop an optimized Kalman filter model for vessel trajectory estimation.	Kalman filter
Peng et al., 2010	Propose a self-adaptive Kalman filter model with speed, location to forecast a short-term vessel trajectory.	Self-adaptive KF
Guo et al., 2018	Construct a K order Hidden Markov model using ship speed, ship's heading, location, weather condition as variables to improve precision of prediction.	Markov
Qiao et al. 2014	Design high-order standard Markov model of tree structures to predict vessel trajectory.	Markov
Zhou et al., 2019	Train a Back Propagation neural network to predict the vessel trajectory on the river.	BP neural network
Tang et al., 2019	Merge a sequence prediction method into LSTM to estimate the vessel trajectory.	LSTM
Gao et al., 2018	Introduce a bidirectional LSTM model using AIS data for the online prediction of ship behaviour.	LSTM
Vespe et al., 2012	Extract of waypoints connecting with sea lanes and routes for trajectory prediction.	Grid-based approach
Liu and Chen, 2014	Revise the missing AIS data using temporal link prediction based on tensor CP (CANDECOM/PARAFAC) decomposition.	Grid-based approach

2.1.2 semantic trajectory mining

The semantic trajectory connects the semantic information with a spatial-temporal trajectory to reflect the behaviors, intentions, and habits of a vessel. The semantic information can be derived from the original trajectories for different objectives. In the research of vessel destination prediction, main semantic information includes the stay points and the similarity among trajectories. Hence, this section for semantic trajectory mining is divided into two parts: stay points-based approach and similarity-based approach.

(1) Stay points-based approach

Most research focuses on vessel stay points for anomaly detection and develops lots of algorithms to build the semantic trajectory with stay region information. After filtering the anomaly stopovers on the route of a vessel, the stay points in port regions can be obtained to extract the semantic trajectory of port calls. Based on the history trajectories with port calls, the next port of a vessel can be predicted directly using the Markov model or the higher order shipping network model (Yang et al., 2014; Tao et al., 2017). This approach can be employed before the voyage.

(2) Similarity-based approach

The spatial-temporal trajectory can be converted into the similarity matrix, comparing with selected trajectories. The similarity matrix as comparison features forms the semantic trajectory for a vessel. Then these similarity trajectories can be used to train the classification machine learning model. When inputting the similarity trajectory of a vessel on different days, the destination port will be estimated. However, this approach is limited by the selected trajectories for comparison (Zhang et al., 2020).

Overall, Table 2.2 summarizes the related work of semantic trajectory mining. While most studies pay attention to the detection of semantic events and their links to the movement patterns, the literature of vessel destination prediction is rare. In our research, we contribute to the destination port prediction with both stay points-based approach and similarity-based approach of semantic trajectory. For the former approach, we scan the trajectory with a time window and record associated information, including port name, port id, arrival time etc. So, we can extract different orders of sequences of port calls. Combined with the random forest classifier, we can predict the next port in advance. For the latter approach, we use the TRACLUS algorithm (Lee et al., 2007) to cluster and output the representative trajectories for different routes. Hence, the similarity probability can be calculated between a sailing trajectory and these representative trajectories. Based on the similarity probability of different sailing days and other features, such as departure port, IMO number and deadweight tonnage (DWT), we can train the GBDT classifier to predict the destination port in real time.

Table 2. 2 Previous studies on vessel semantic trajectory mining

Source	Content	Objective
Vandecasteele et al., 2014	Incorporate the concept of semantic events with semantic trajectories.	Semantic trajectory framework
Huang et al., 2020	Detect the semantic descriptions for ship mobility patterns.	Semantic trajectory framework
Shahir et al., 2019	Mine the semantic events for illegal fishing.	Semantic events
Villa and Camossi et al., 2011	Infer the container itineraries with semantic risk routes.	Semantic events

Tao et al., 2017	Adopt HON algorithm to predict the future port sequence.	Destination prediction by semantic trajectory
Zhang et al., 2020	Construct the similarity distance matrix among trajectories to estimate the vessel destination.	Destination prediction by semantic similarity

2.2 Stay Points Recognition

In recent research, the stay points recognition plays an important role in trajectory data mining. The methods for stay points recognition have two categories: association rules-based method and clustering-based method.

(1) Association rules-based method

Stay points of most trajectories have a distinctive characteristic that the speed is nearly zero. This characteristic of stay points makes the association rules possible. Set a series of rules to associate with the speed and duration of a trajectory in advance. Then filter the points not meeting these rules to find the stay points of a trajectory. However, in the real world, many trajectories may have pseudo-stay points such as traffic jams. These pseudo points lead to the wrong recognition. To improve the recognition accuracy, more and more complex rules can be added but the efficiency and generality are limited. Hence, this method is often used for the preliminary recognition of stay points (Ashbrook and Starner, 2002; Schuessler and Axhausen, 2009; Huang, et al, 2016).

(2) Clustering-based method

To detect the continuous stay points of a trajectory, many researchers propose the clustering-based method. This method can recognize the stay points accumulating in the small space for a period. The aggregation region of many stay points is called a

cluster. Among different ways for clustering, density clustering for spatial-temporal trajectory is the typical one.

The algorithms for density clustering include TDBC,POSMIT, ST-DBSCAN and CB-SMoT etc. TDBC algorithm regards the points with constraints of the time and number thresholds in timeline as stay points of a trajectory (Fu et al., 2016). POSMIT considers the likelihood distribution of points in different locations to detect the possible stay points (Bermingham and Lee, 2018). However, TDBC and POSMIT only take either the time interval or the space distance into consideration. ST-DBSCAN combines the time and space threshold to cluster the stay points, while the hyperparameters are hard to decide (Birant and Kut, 2007). But ST-DBSCAN has the overlapping region of stay points in different clusters. It also needs to recalculate the density regions when just a few points are changed. Therefore, the scanning trajectory sequence algorithm appears for clustering stay points. CB-SMoT algorithm is the typical one. It improves the performance by recognizing a low-speed sub-trajectory. The points of sub-trajectory are detected as stay points (Palma et al., 2008). Moreover, some studies also develop the data field theory for stay points detection (Zhao et al., 2017).

Overall, Table 2.3 summarizes the related work for stay points recognition. In our study, we develop an optimized CB-SMoT algorithm to detect the stay regions in sequence by scanning the AIS data of a vessel. Combined with the port coordinates, we can get the sequence of port calls for this vessel. Our optimized CB-SMoT algorithm can identify most port calls of tramp ships. Based on the sequences of port calls for

VLCC, we can extract the spatial-temporal or semantic trajectories to build three different models. These models form the framework of destination prediction

Table 2. 3 Previous studies on stay points recognition

Source	Content	Stay points recognition
Ashbrook and Starner, 2002	Design an algorithm to detect the stay points of wearable computers.	Association rules
Schuessler and Axhausen, 2009	Process the GPS raw data for positions.	Association rules
Huang, et al, 2016	Discover the repeated behaviors by stay points.	Association rules
Luo et al., 2017	Adopt data filed theory to find stay points.	Clustering-based
Zimmermann et al., 2009	Use the CB-SMoT to detect error stops.	Clustering-based
Chen et al., 2014	Develop the T-DBSCAN algorithm for GPS trajectories.	Clustering-based
Zhao et al., 2017	Detect the hotspots from trajectories.	Clustering-based
Damiani et al., 2014	Propose a seqscan clustering algorithm to extract animal stay points.	Clustering-based

2.3 Distances for Trajectory Similarity Measure

To measure the similarity between two trajectories, the trajectory distance can be adopted as a metric. In recent work, the trajectory distance can be divided into three categories: point-based distances, shape-based distances, and segment-based distances.

(1) Point-based distances

This kind of point-based method calculates the distance by points of different trajectories, including Euclidean distance, Dynamic Time Warping (DTW), Longest Common Sub-Sequence (LCSS), Edit Distance on Real Sequence (EDR). Euclidean distance represents the true distance between a pair of corresponding points, only used for trajectories of the same length. DTW aims to get the smallest warping cost path between matched points of two trajectories. DTW takes the time differences into consideration but is sensitive to the noise (Keogh and Ratanamahatana, 2005). LCSS and EDR are distances with the adaptation of string similarity. LCSS counts the number of matched pairs and EDR counts the operation cost to fix unmatched pairs. However, both are limited by the matching threshold (Vlachos et al., 2002; Chen et al., 2005).

(2) Shape-based distances

The Hausdorff distance and Fréchet distance are two typical distances based on the shape of trajectories for calculating. Both distances work well when two trajectories have enough information to reflect the whole shape, but they are limited when missing parts of the trajectory records (Aronov et al., 2006; Min et al., 2007). Symmetrized Segment-Path Distance (SSPD) compares two trajectories as a whole and is not sensitive to the sudden variation of trajectories (Besse et al., 2016).

(3) Segment-based distances

The One-Way Distance and Locality In-between Polylines (LIP) distance represent the distance of segments. One-Way Distance considers a trajectory as the piecewise line segment and considers another trajectory as the series of points (Lin and Su, 2008). LIP distance calculates the polygon area between intersections points of

segments but is only available for 2D trajectories (Pelekis et al., 2007).

Table 2.4 summarizes the trajectory distances for similarity measurement in previous research. Based on the above research, we prefer SSPD of the shape-based distances to calculate the similarity matrix among trajectories. Because SSPD takes the length and variation of two trajectories into consideration to measure their physical distance. Moreover, the numbers of trajectory points in SSPD are not limited and SSPD have a good performance even if the data quality is low.

Table 2. 4 Previous works for similarity measurement with trajectory distances

Source	Content	Trajectory distance
Jin et al., 2017	Propose a shipping frequency model with DTW to learn trajectory information.	DTW
Li et al., 2019	Measure the trajectory similarity with LCSS.	LCSS
Liu and Yang., 2009	Design an optimized EDR model to analyze the trajectory similarity.	EDR
Sheng and Yin., 2018	Develop a trajectory clustering algorithm with Hausdorff distance.	Hausdorff distance
Mascret et al.; 2006	Compute the coastlines with Fréchet distance for measurement.	Fréchet distance
Zhang et al., 2020	Employ the Symmetrized Segment-Path Distance with random forest to forecast the vessel destination.	SSPD
Pelekis et al., 2012	Develop a similar trajectories visualization method with LIP distance	LIP distance
Ma et al., 2014	Adopted One-Way Distance to assess the vessel trajectory similarity.	One-Way Distance

2.4 Summary of the Chapter

In this chapter, based on spatial-temporal trajectory and semantic trajectory, we conclude two kinds of frameworks for vessel destination port prediction. We explain how to employ these two frameworks in our research. Then we review the stay points recognition methods and the similarity distance metrics in the frameworks. Based on these methods and metrics, we propose an optimized CB-SMoT algorithm in the following chapter for stay points recognition and decide SSPD as the similarity distance.

Chapter 3: Development of an Optimized CB-SMoT Algorithm for Port Calls Recognition

In this chapter, as the basis of research, high quality and reliable AIS data should be achieved at first by preprocessing that needs filtering and compressing. However, these data can just provide the fundamental information of a vessel. The trajectory patterns and associated voyages of the vessel still need further work to obtain. Therefore, we develop an optimized CB-SMoT algorithm to solve this problem and validate its performance. The optimized CB-SMoT algorithm can divide the trajectory of a vessel into the movement pattern and the port calls pattern rapidly and effectively by scanning the AIS data. It can also recognize the stay points. We can depend on the algorithm to extract voyages and more semantic information of port calls.

3.1 AIS Data Preprocessing

Before introducing the algorithm, some data preprocessing steps should be conducted to ensure the data quality in this study. AIS raw data contains many redundant information and noise points, which cannot be used directly. The data cleaning and compression is necessary. We conduct these tasks by two steps:

Step 1: Speed-based heuristic filtering to clean data.

Step 2: Douglas–Peucker (DP) algorithm to compress data.

In step1, the noise point is the outlier that has an obvious deviation in position. The simple and effective way to remove noise is the speed limitation. For the continuous records of AIS, the distance between two consecutive locations cannot exceed the

product of maximum speed and time interval. Hence, we adopt the maximum speed 15 knots of bulk carriers and 17 knots of tanker ships to filter noise. In general, the average service speed of VLCCs is 11~12 knots, which is not the same as the maximum speed. In step 2, to reduce the time cost, we employ DP algorithm to compress data for every vessel in our database. The compressed data capacity is about 70% of the raw data.

3.2 Introduction to CB-SMoT Algorithm

Clustering-based Stops and Moves of Trajectories (CB-SMoT) algorithm is presented for finding clusters in a single trajectory by Palma et al. (2008). It considers both the time and the distance of the sub-trajectory, aiming to recognize the low-speed region. Different from the well-known DBSCAN algorithm, this algorithm changes two important concepts for clustering: the Eps-linear-neighborhood of a point and the core point. These two concepts are described as the follow:

- Eps-linear-neighborhood.

Assuming a trajectory with points sequence $\{p_0, p_1, \ldots, p_k, p_{k+1}, \ldots, p_N\}$, the Eps-linear-neighborhood of a point p_k is denoted via $LBEPS(p_k)$:

$$\left(\sum_{i=m}^{k-1} \mathrm{dist}\left(p_i, p_{i+1}\right)\right) \leq \mathrm{Eps} \cup \left(\sum_{i=k+1}^{n} \mathrm{dist}\left(p_{i-1}, p_i\right)\right) \leq Eps, \tag{1}$$

where $t_0 \leq t_m < t_k < t_n \leq t_N$. As shown in Figure 3.1, in a trajectory, the red point O with the radius Eps of 50km has seven blue points as neighborhoods. But only point B, C, D, E can meet the above condition of $LBEPS(p_o)$ and become the linear neighborhood of point O. All these points form the sub-trajectory.

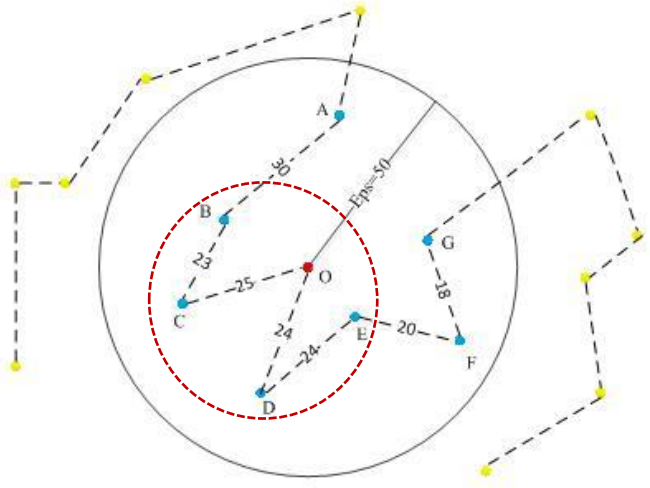


Figure 3. 1 An illustration for Eps-linear-neighborhood

- Core point.

If a point $p = (x_p, y_p, t_p)$ of a trajectory can meet the requirement: for points in *LBEPS(p)*, exist the $|t_n - t_m| \geq MinTime$, where m is the first point and n is the last point in the linear neighborhoods ordered by time. Then the point p can be regarded as a core point. The core point and its linear neighborhoods consist of a cluster as a set of contiguous time-space points.

When the CB-SMoT algorithm works, it scans a trajectory to find core points and construct different clusters. The clusters with directly density-reachable core points can be merged into one cluster. Moreover, the parameter can be self-adaptive adjustment by the quantile function. In fact, for a single vessel trajectory with some labeled clusters in Figure 3.2, the CB-SMoT can detect the unknown clusters X and Y as a kind of unsupervised learning and overcome the problem of clustering an incomplete trajectory with missing data.

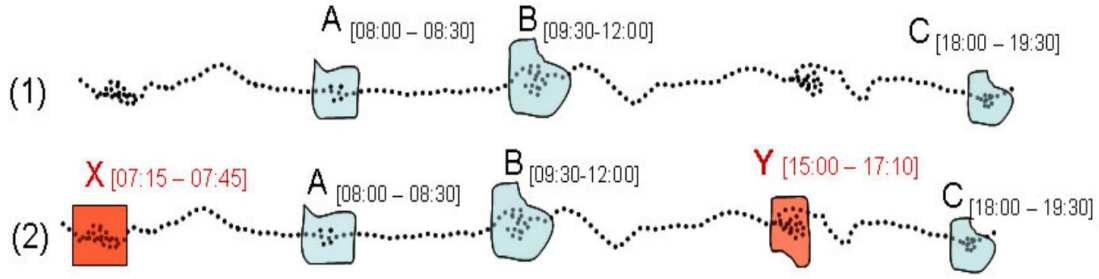


Figure 3. 2 The single raw trajectory and single semantic trajectory (Palma et al., 2008)

3.3 The Optimized CB-SMoT Algorithm

The tramp services usually have no fixed schedules for port calls. Aiming to extract the sequence of port calls for a tramp ship in a fast and accurate way, we develop the optimized CB-SMoT algorithm with a world ports list. The key contribution of our algorithm can be summarized as follows:

i. The port calls information and stay regions can be recognized.

ii. Every cluster is spatially and temporally disjointed. For different port calls, the corresponding cluster is unique.

iii. The time complexity O(n) of the optimized algorithm performs better than $O(n^2)$ of the original CB-SMoT algorithm.

In this section, we will present how our algorithm works and give the illustration of an oil tanker.

3.3.1 Basic definitions

The conceptual views and definitions of a trajectory are based on the original CB-SMoT algorithm. We change and put forward the following main definitions. They provide a foundation for our optimized algorithm.

Definition 1: Trajectory sample.

A trajectory sample of a vessel consists of a series of consecutive points, as $\{p_0, p_1, ..., p_N\}$. The point is in the form $p = (x_p, y_p, t_p)$, including the coordinates and timestamps. Hence, the trajectory sample can be denoted:

$$\textit{Trajectory sample: } \left[P_{\text{begin}} \ldots . P_{\text{end}} \right] \rightarrow \textit{ space and time} \tag{2}$$

Definition 2: Temporal sub-trajectory and Center point.

Any consecutive segment of a trajectory sample can be presented by $\{p_m, ..., p_n\} \in \{p_{start}, ..., p_m, ..., p_n..., p_{end}\}$. If the segment can meet both the time threshold $T_{interval}$ and the speed threshold *Maxspeed* as:

$$\textit{Time and speed thresholds: } \begin{cases} \left| t_n - t_m \right| \le T_{interval} \\ \dfrac{\sum_{i=m}^{n-1} velocity\left(p_i, p_{i+1} \right)}{n-m} \le Maxspeed \end{cases}, \tag{3}$$

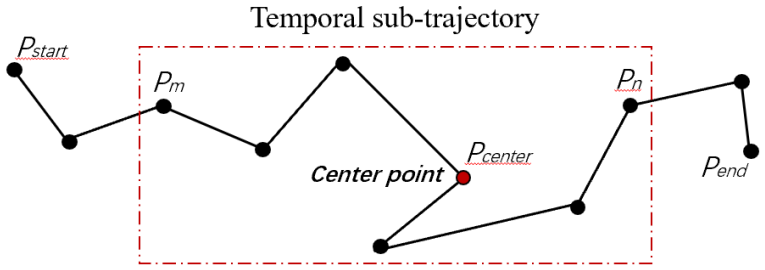
then it can be called a temporal sub-trajectory for stay points. The median point of a temporal sub-trajectory can be regarded as the center point P_{center}, as the red point shown in Figure 3.3(a).

Definition 3: Core point and cluster.

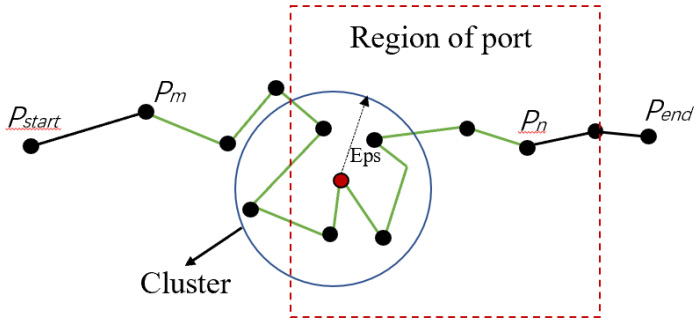
If the point P_K is the center point of a temporal sub-trajectory, and the number of points in the Eps-neighborhood of P_K exceeds the *MinP* as:

$$N\left(p_k \right) = \left\{ q \in P | \text{dist}\left(p_k, q \right) \le Eps \right\} > MinP \tag{4}$$

then P_K is called a core point. The core point and all its neighborhoods built a cluster for stay points, as the blue circle in Figure 3.3(b). If any point in the cluster belongs to the region of port, the cluster can be considered as the stay region of port calls.



(a) The temporal sub-trajectory and center point



(b) The core point and cluster of the port calls region

Figure 3. 3 Concepts of sub-trajectory, points, and cluster

Definition 4: Merging clusters.

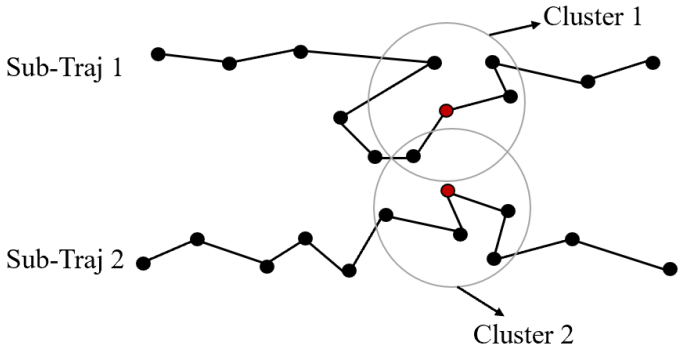
As Figure 3.4 (a) shows, for any two clusters of the sub-trajectory 1 and sub-trajectory 2, if one point exists in the overlapping part of the two clusters, then both clusters can be merged into one cluster. When the overlapping part contains the core point of a cluster, the two core points are directly density-reachable and other points of the two clusters are density-reachable or connectable. Moreover, Figure 3.4 (b) describes the situation that a port of call region has more than one cluster. Even though two clusters have no overlaps but in the same port region, they should be merged.

Definition 5: Extending clusters.

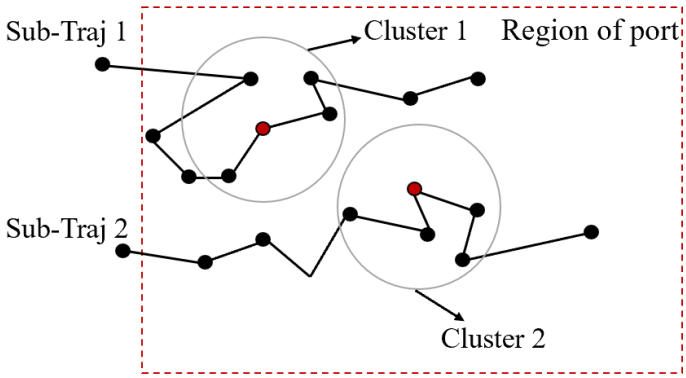
If two clusters of the sub-trajectory 1 and sub-trajectory 2 cannot be merged, and they are located at different regions of port A and port B. as shown in Figure 3.4 (c), the stay region of port calls should be extended to a new one.

Definition 6: Anomaly detection.

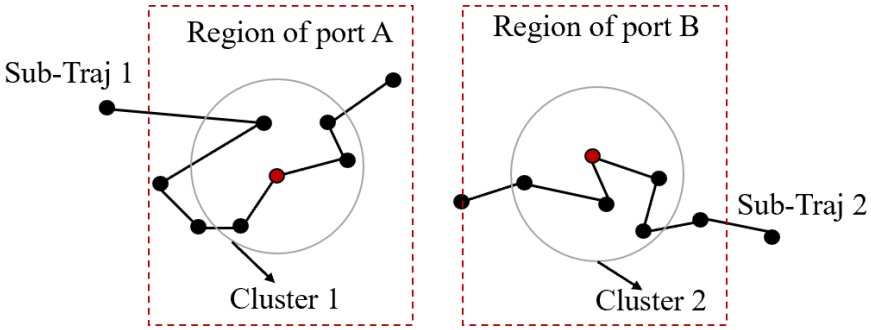
When a cluster B cannot neither be merged to previous cluster A in the port region nor be extended to form a new one, the cluster B is considered as the abnormal area of stay points.



(a) One merging situation



(b) Another merging situation



(c) The extending situation

Figure 3. 4 Concepts of merging and extending clusters

3.3.2 Algorithm workflow

The optimized CB-SMoT algorithm is presented by time and speed-aware, dense-based functions and by scanning the temporal sub-trajectories to merge or extend the cluster. The overall workflow and pseudo-code are described in Table 3.1 with many steps. In the first step, we extract the sub-trajectory from the complete trajectory with an ordered time window and find the center point. In the second step, we calculate the average velocity and Eps-neighborhood of the center point to identify whether the sub-trajectory is nearly stopping at a very slow speed and construct the cluster. Here, the core point can be obtained. In the third step, we match the cluster with the closest port. In the final step, if the cluster is not the initial cluster, we determine whether to merge the cluster with the previous one or extend it as a new cluster based on definition 4 and 5. When extending, the port of call information and associated stay region of the previous cluster can be recorded. After scanning the trajectory, the workflow can produce the sequence of port calls.

More details about the parameters and functions are beneficial to explain the algorithm workflow. The time window $T_{interval}$ and the radius *Eps* are adjusted with

respect to different sampling frequencies of the trajectory. Considering the possible AIS signal drift, the *Maxspeed* is a threshold near zero but not zero to delimit the space range of sub-trajectory. When matching a cluster with the world ports list, the distance between the core point and port coordinates is calculated by *Haversine distance*. If this distance is less than 10 n mile or between 20~30 n mile, the vessel is located at the berth or anchorage of the port. Otherwise, the stay region is abnormal.

Table 3. 1 The optimized CB-SMoT algorithm workflow

Algorithm 1 Optimized CB-SMoT.

Input: T=[P1, … ,Pn] *//the trajectory record from a vessel's AIS data*
Parameters:
    $T_{Interval}$ *//the time threshold for extracting the temporal sub-trajectory*
    Maxspeed *//the average speed threshold of the sub-trajectory*
    MinP *//the minimum number of points in the Eps-neighborhood*
    Eps *//the radius for the range of neighborhoods around a core point*
    Worldportlist *//the coordinates of world ports*

output: Port_call_inforamtion(Arrival_time, Departure_time, Stay_duration, Port_ID,Port_name)

Method:
cluster_core=Ø *//the set to collect core points*
cluster_point=Ø *//the set to collect points in cluster*
port_seq=Ø *//the set to collect the port calls information*
start_index=0 *//initial trajectory point index*

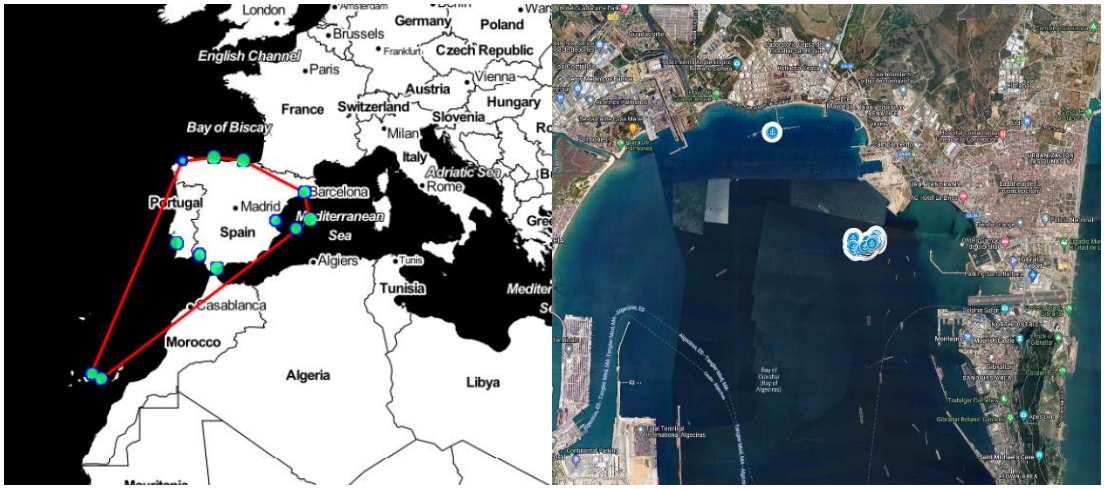
while start_index<len(T)**:**
    //extract the temporal sub-trajectory and find the index of center point and last point
    T_sub,center_index,end_index=**getSubtrajectory**(start_index,T,$T_{Interval}$)
    //calculate the spatiotemporal distance matrix between each point in sub-trajectory
    timeDisMat, disMat=**compute_squared_EDM**(T_sub)
    //calculate the average speed of the sub-trajectory
    avg_v=**speed**(timeDisMat,disMat)
    // search the number and points of neighborhoods
    N,Neighbor_points=**searchNeighbors**(T_sub,center_index,disMat,Eps)

    if avg_v<=Maxspeed and N>=MinP**:**
        //match the port information

```
        port_id,port_name=portMatch(Worldportlist,T_sub,center_index)

        if cluster_core,cluster_point,port_seq=Ø and port_id,port_name!=null value:
            //find the first cluster
            cluster_core.append(Point[center_index])
            cluster_point.append(Neighbor_points)
            port_seq.append([port_id,port_name])
        elif port_id,port_name!=null value:
            //identify whether two sub-trajectories can be directly density-reachable
            link=linkDensity(cluster_core[-1],T_sub,Eps)

            if link==True or (link==False and port_id== port_seq[-1][0]):
                cluster_point,cluster_core=merge(cluster_point[-1],Neighbor_points,
                                                 cluster_core[-1],Point[center_index])
            elif (link==False and port_id!= port_seq[-1][0]):
                Extend{cluster_core.append(Point[center_index])
                       cluster_point.append(Neighbor_points)
                       port_seq.append([port_id,port_name])}

            Arrival_time=Datetime(cluster_point[-2][0])
            Departure_time=Datetime(cluster_point[-2][-1])
            Stay_duration=Departure_time-Arrival_time

            writetocsv(Arrival_time, Departure_time, Stay_duration, port_seq[-2][0],
                       port_seq[-2][-1])
            endif
        endif
    endif
endwhile
```

3.3.3 Illustration of the algorithm

To illustrate the optimized CB-SMoT algorithm, we choose an oil tanker of Spanish flag (MMSI 224432000) to extract its sequence of port calls in 2017. As shown in Figure 3.5, this ship visited 13 ports around Spain. For Gibraltar port, the stay region was in the berth and very close to the crude oil wharf. Table 3.2 presents the result of the sequence of port calls with 60 voyages in 2017.



(a) All the visited ports (b) The stay points in Gibraltar port

Figure 3. 5 An illustration for the stay regions

Table 3. 2 An illustration of the sequence of port calls

Port calls order	Arrival time	Departure time	Stay duration (days)	Port ID	Port name
1	2017/1/1 9:33	2017/1/16 20:20	15.4491	14011	Puerto de la Hondura Oil Terminal
2	2017/1/20 15:23	2017/1/22 12:29	1.8794	13433	Sines
3	2017/1/25 2:48	2017/1/27 5:49	2.1256	18937	Valencia
4	2017/2/3 4:36	2017/2/18 23:13	15.7758	25923	Gijon
…	…	…	…	…	…
57	2017/12/12 12:49	2017/12/14 11:34	1.9480	19610	Punta Lucero Tanker Terminal
58	2017/12/16 11:39	2017/12/17 9:51	0.9248	25923	Gijon
59	2017/12/17 20:44	2017/12/20 11:28	2.6140	19610	Punta Lucero Tanker Terminal
60	2017/12/20 22:29	2017/12/22 19:17	1.8667	25923	Gijon

3.4 Performance of the Algorithm

3.4.1 Data and metrics

To validate our optimized CB-SMoT algorithm, we select 500 bulk carriers and 500 tanker ships with known port calls in 2017, respectively. The frequency of port calls for bulk carriers is 17170 and for tanker ships is 43730. The categories of these tramp ships are described in Appendix A.

The metric used to measure the algorithm performance is grounded on the longest common sequence (LCS). It can assess the text sequence similarity as the equation:

$$Simi_{text} = \frac{2L_{LCS}}{L_{predicted} + L_{known}}, \tag{5}$$

Where L_{LCS} means the length of LCS, $L_{predicted}$ and L_{known} are the lengths of predicted and known sequences of port calls. We calculate the average similarity for bulk carriers and tanker ships to represent the accuracy ratio of the algorithm. Moreover, we compare our algorithm with ST-DBSCAN and POSMIT.

3.4.2 Results

The results of different algorithms for port calls recognition are presented in Table 3.3. We can see our algorithm can achieve an accuracy larger than 85% for bulk carriers and an accuracy larger than 90% for tanker ships. In contrast, our algorithm's performance is better than those of ST-DBSCAN and POSMIT. Therefore, the optimized algorithm can be depended on to extract the sequences of port calls for VLCCs in the next chapter.

Table 3. 3 Measurement of algorithm performance

Algorithms	Bulk carriers		Tanker ships	
	Number of detected port calls	Accuracy ratio	Number of detected port calls	Accuracy ratio
ST-DBSCAN	9765	57.45%	27974	64.12%
POSMIT	13989	82.37%	37655	86.26%
Optimized CB-SMoT	**14927**	**86.41%**	**40653**	**90.63%**

3.5 Summary of the Chapter

In this chapter, we introduce the steps of AIS data preprocessing. Then we elaborate on how we develop the optimized CB-SMoT algorithm. The definitions and workflow are explained, and an illustration of an oil tanker is given. In addition, the algorithm performance is validated with a better accuracy than those of other similar algorithms.

Chapter 4: Destination Port Prediction Models

The optimized CB-SMoT in Chapter 3 has a strong connection with this chapter. It helps build the bases of data material (samples) for all three models. Voyages of different routes and semantic information from port calls are extracted by the algorithm. In this chapter, considering the AIS data sampling frequency for VLCCs, we set the main parameters of the algorithm in accordance with $T_{Interval}$*=15min, Maxspeed=0.1 knot, MinP=10, Eps=300m*. Our AIS data of 402 VLCCs of a fleet in 2020 are from the company of *CHINA MERCHANTS ENERGY SHIPPING*. The distribution of extracted samples is different for every model. For model 1, we use the full samples. For model 2, to enhance the effectiveness of prediction, we use partial samples that are voyages of VLCCs departing from the Middle East, which account for more than 90% of voyages (see Appendix B). Similarly, for model 3, we use partial samples that are voyages of frequent routes appearing more than 60 times. Besides, we analyze the model performance and related results in more detail. We also discuss the research findings and describe how to apply these models for practices.

4.1 Model 1-The High Order Sequence of Port Calls Model (Semantic-Based)

4.1.1 Data description

In model 1, we select the sequences of port calls of 402 VLCCs. These ships have the port calls every month. In total, 9587 port calls are extracted and recorded. As the sequence of port calls heatmap of Figure 4.1 shows, most port calls are in the Mideast, East Asia, North Sea, Gulf of Mexico and Gulf of Guinea. These regions include main

groups of export ports and import ports for crude oil. Other port calls are the transshipment ports such as Singapore.

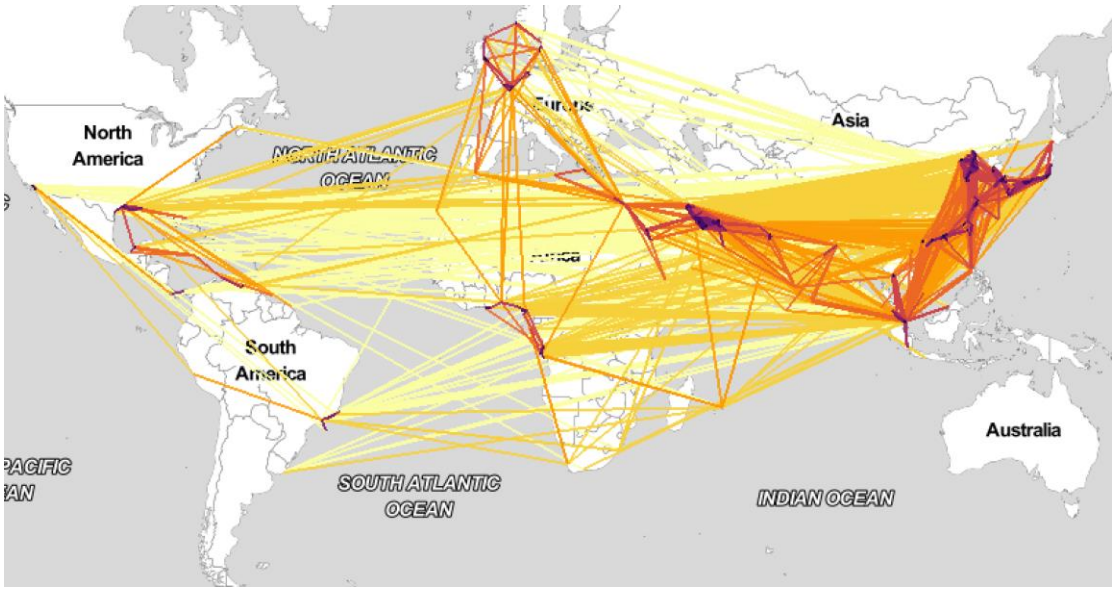


Figure 4. 1 The heatmap for VLCC sequences of port calls

4.1.2 Methodology and metrics

(1) Methodology

Before sailing, the vessel cannot produce the sailing trajectory. The possible and useful information is the historical records of port calls. These semantic records bring the chance to predict the destination port in advance. A random walk simulation with likelihoods in higher order networks is one typical method (Tao et al., 2017), but it is complex to find the weights of parameters. Hence, we attempt to train a simple but effective classification model with sequences of port calls to predict the destination.

The sequences of port calls with different orders are extracted by the optimized CB-SMoT algorithm at first. As shown in Table 4.1, the definition of order is the number of previous ports. The more previous ports are counted, the higher order is. Then combining the sequences of port calls under different orders with the random forest classifier from the Scikit-learn library, we can construct the classification model

for prediction. In the model, X is the feature and Y is the labels of categories. We can denote them as:

- X: Different order sequences of previous ports (minimum frequency 2)
- Y: Destination ports.

Here, we define the minimum frequency for a pair of departure and destination ports, which is at least more than 2. Because the single appearance of one pair cannot provide enough information and the movement pattern.

Table 4. 1 An illustration for high order sequence of port calls (IMO 9805099)

Order	Previous port 5	Previous port 4	Previous port 3	Previous port 2	Previous port 1	Destination port
1					Das Island	Fujairah
2				Zirku Island	Das Island	Fujairah
3			Sir Bani Yas Port	Zirku Island	Das Island	Fujairah
4		Mesaieed	Sir Bani Yas Port	Zirku Island	Das Island	Fujairah
5	Doha (Qatar)	Mesaieed	Sir Bani Yas Port	Zirku Island	Das Island	Fujairah

(2) Metrics

To measure the performance of the random forest classifier, we should know the concepts of TP (True positive), FP (False positive), TN (True negative), FN (False negative). The detail concepts are given as follows:

TP: The positive sample is classified into the positive group

FP: The negative sample is classified into the positive group.

TN: The negative sample is classified into the negative group.

FN: The positive sample is classified into the negative group.

Based on the above concepts, we can calculate the precision, recall, F1-Measure and accuracy as the indicators for measurement. For multi-class tasks, these indicators are the macro average values of different classes for prediction. The formulas of the indicators are shown as:

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$F1\text{-}score = \frac{2\ Precision\ *\ Recall}{Precision\ +\ Recall} \tag{8}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

4.1.3 Results

We study the effects of the order on the prediction by a random forest classifier. The datasets are divided into training and test sets randomly with the ratio 8:2. As the results shown in Table 4.2, the four indicators increase with the growth of order. When the order of the sequence of port calls is larger than 3, the F1-score and accuracy can exceed 0.80. Because the higher order contains more information about previous ports. For example, for an oil tanker, when the order is 1 including one previous port, the alternative destination ports are fifteen. But when the order is 2 including two previous ports, it has only ten alternative destination ports. As a result, the probability of being

correctly predicted becomes larger. Therefore, it is necessary to use a higher order sequence to achieve a good performance.

Table 4. 2 The performance of destination port prediction with different orders

Order	Precision	Recall	F1-score	Accuracy
1	0.309	0.305	0.307	0.307
2	0.489	0.475	0.481	0.480
3	0.596	0.601	0.598	0.605
4	**0.817**	**0.801**	**0.809**	**0.807**
5	**0.859**	**0.846**	**0.852**	**0.852**

4.2 Model 2-The Trajectory Similarity Model (Semantic-Based)

4.2.1 Data description

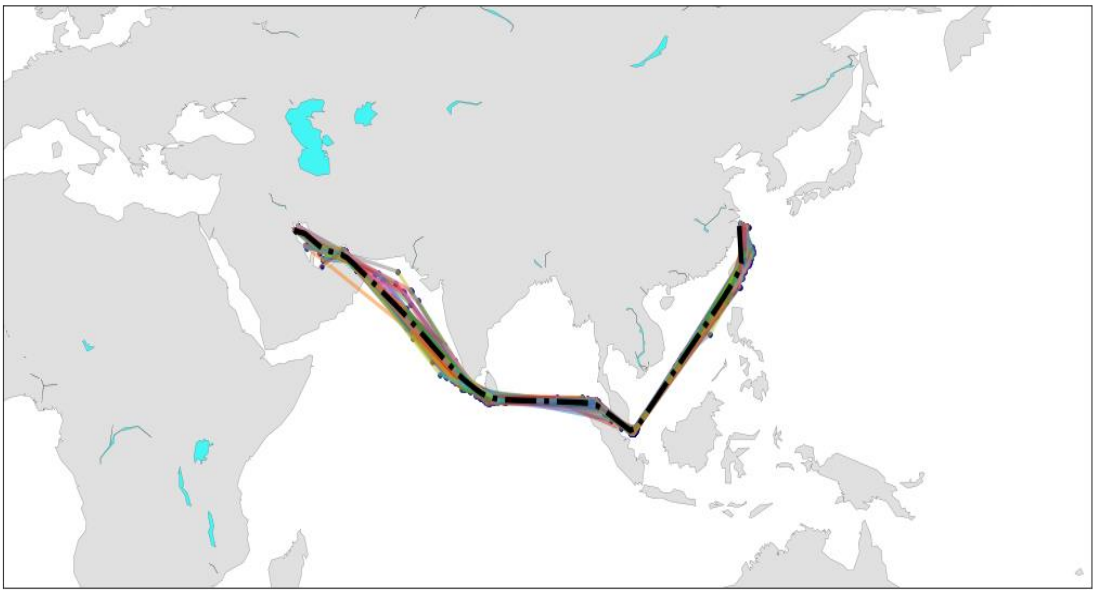
In model 2, we select 670 voyages from 329 VLCCs based on the port calls. The voyages are distributed among 108 sea routes. The departure ports of all routes are in the Middle East and Gulf (MEG). The destination ports are located at seven different regions, including West India, East India, South of China, Middle of China, North of China, Korea and Japan.

4.2.2 Methodology and metrics

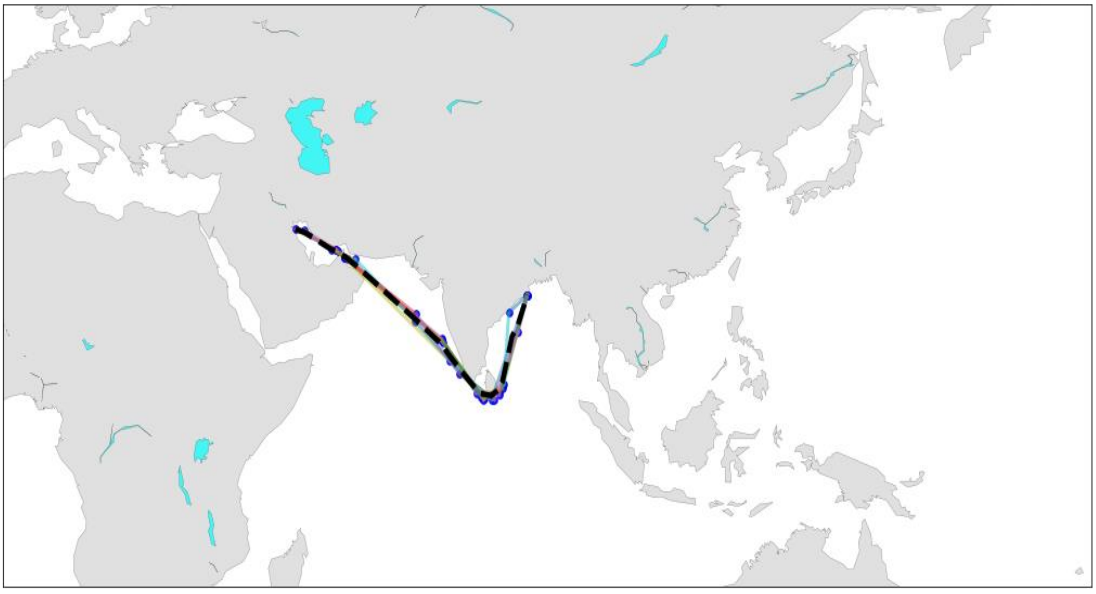
(1) Methodology

At first, we adopt the TRACLUS algorithm to extract the representative trajectories. The definition of representative trajectory is a trajectory with a series of points as the standard for the target route. The representative trajectory does not really exist and is produced by clustering the trajectories of different voyages along the same

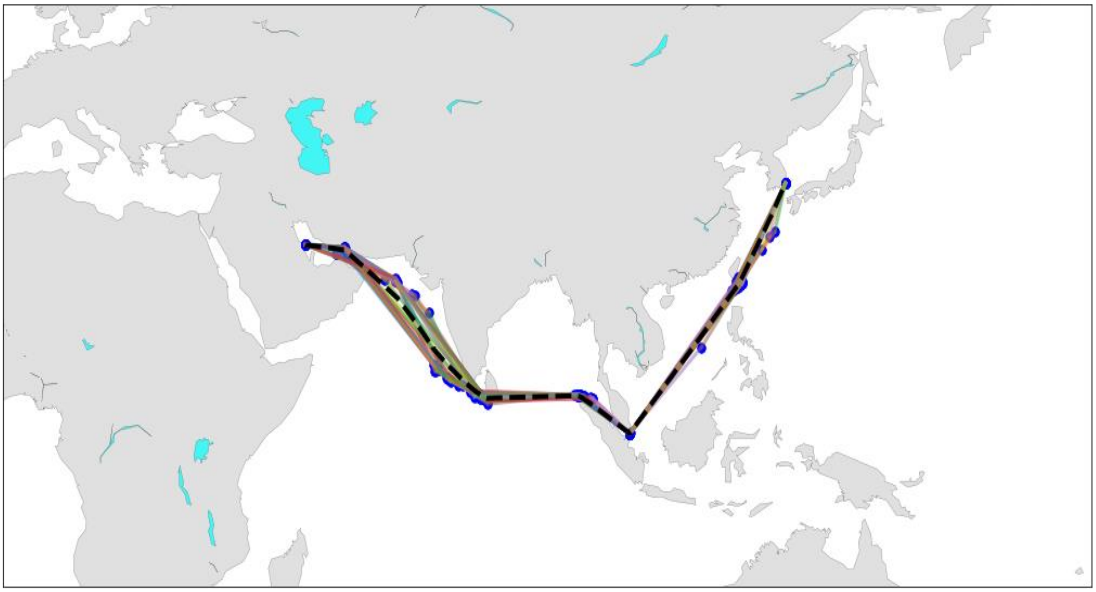
route. We get 108 representative trajectories between different ports and get 7 representative trajectories between different regions. As the illustration in Figure 4.2, the black dot-dash-line and black dotted line denote the representative trajectories.



(a) The route from Middle East and Gulf region to Middle of China region



(b) The voyage from port of Fujairah (United Arab Emirates) to port of Paradip (India)



(c) The voyage from port of Ju'aymah Crude and LPG Terminals (Saudi Arabia) to port of Onsan (Korea)

Figure 4. 2 Illustrations for representative trajectories

After extracting the representative trajectories, we use Symmetrized Segment-Path Distance (SSPD) to calculate the similarity for 670 voyages. The SSPD can be calculated as following equations:

$$D_{SSPD}\left(T^1,T^2\right)=\frac{D_{SPD}\left(T^1,T^2\right)+D_{SPD}\left(T^2,T^1\right)}{2}, \tag{10}$$

Where D_{SPD} is the average distance for points of one trajectory to another trajectory. Then we convert the D_{SSPD} to the likelihood by:

$$P=\frac{1}{e^{-D_{SSPD}}} \tag{11}$$

Here, we use $\delta=\{P_{rep\text{-}Tra\ 0},\ P_{rep\text{-}Tra\ 1},\ \ldots,\ P_{rep\text{-}Tra\ N}\}$ to indicate the similarity probability vector between a vessel trajectory and N representative trajectories. As Table 4.3 shows, we can construct the similarity probability matrix for vessel trajectories on different sailing days.

Table 4. 3 The similarity probability matrix

	Day 1	**Day 2**	**Day 3**	**...**	**Day j**
Vessel 1 similarity likelihood	δ_{11}	δ_{12}	δ_{13}	...	δ_{1j}
Vessel 2 similarity likelihood	δ_{21}	δ_{22}	δ_{23}	...	δ_{2j}
Vessel 3 similarity likelihood	δ_{31}	δ_{32}	δ_{31}	...	δ_{3j}
...	...	...	...	...	...
Vessel i similarity likelihood	δ_{i1}	δ_{i2}	δ_{i3}	...	δ_{ij}

Note: (δ_{ij} is a similarity probability vector comparing with N representative trajectories)

At last, combining the ship information with the similarity probability matrix of different sailing days, we can build the classification model using the Scikit-learn library for every sailing day. In the model, X denotes the features and Y denotes the labels of categories. The detailed form can be described as:

- X: IMO, DWT, similarity probability δ for different days
- Y: Destination port/region.

(2) Metrics

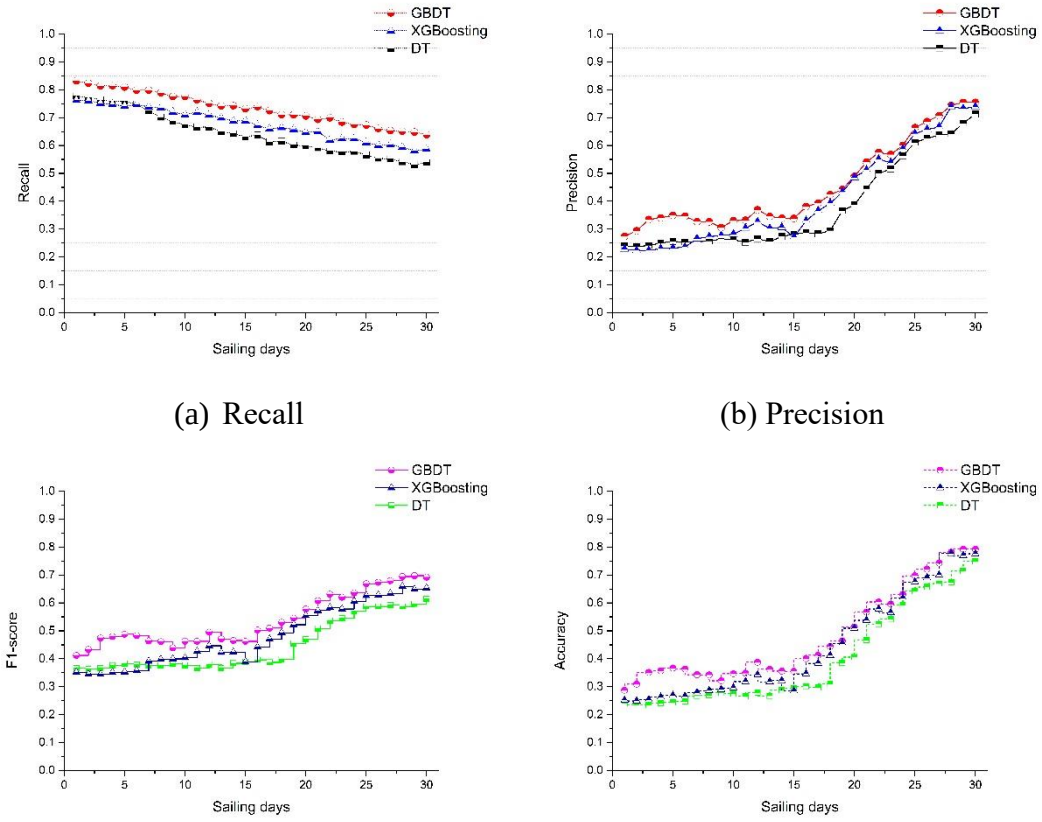
In this model, we use metrics as same as those in section 4.1.2. The metrics include precision, recall, F1-Measure and accuracy.

4.2.3 Results

(1) Model performance for destination port prediction

We use DT, GBDT and XGBoost classifiers to predict 34 destination ports for 670 voyages of every sailing day. We split the dataset into training and test sets randomly with the ratio 8:2. As the results shown in Figure 4.3, among three classifiers, the GBDT

performs best with the highest accuracy. The recall decreases and precision increases with the sailing days. Both F1-score and accuracy increase with the sailing days. The difference between the F1-score and accuracy is not significant, demonstrating the model has a good robustness and avoids the unbalanced samples.



(a) Recall (b) Precision

(c) F1-score (d) Accuracy

Figure 4. 3 The classification performance for destination ports

Note: (Each symbol represents the likelihood of the indicator for every sailing day)

Before the 15 days, the accuracy is low in the range of 0.3~0.35. To explain this phenomenon, we plot the trajectories of the first fifteen days. As Figure 4.4 shows, from the Middle East to Asia, trajectories have the common trajectory segments to pass Sri Lanka and Malacca Strait. The shapes of the trajectories have no sharp distinction. Therefore, the classifier cannot make a correct classification. When the VLCCs sail for

more days, the shapes of trajectories change a lot with different ships' headings to their destination ports. As a result, the similarity probabilities are very different from each other. The accuracy can exceed 0.6 after 20 days and reach 0.78 at the end.

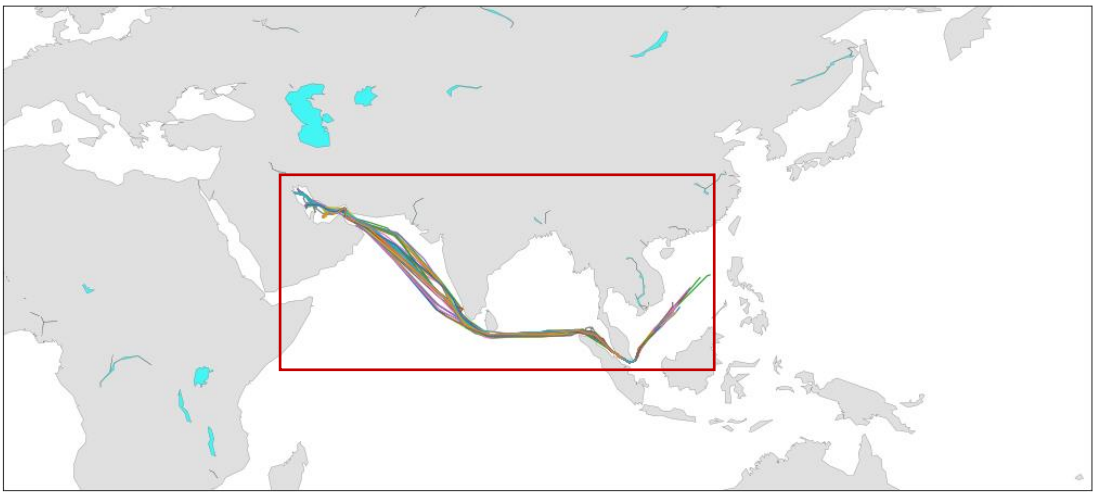


Figure 4. 4 The common trajectory segments in first fifteen sailing days

(2) The relationship between sailing days and destinations

To explore the relationship between sailing days and destinations, we study how many days are needed to identify different destination regions for VLCCs. As Figure 4.5 shows, to achieve a high and believable accuracy above 0.6, the destination region of West India can be identified on the sixth day and East India on the ninth day. Then the South of China regions can be identified on the seventeenth day, while the Middle and North of China regions are recognized during nineteenth to twentieth days. Korea and Japan are geographically close to each other, it needs more days to predict correctly.

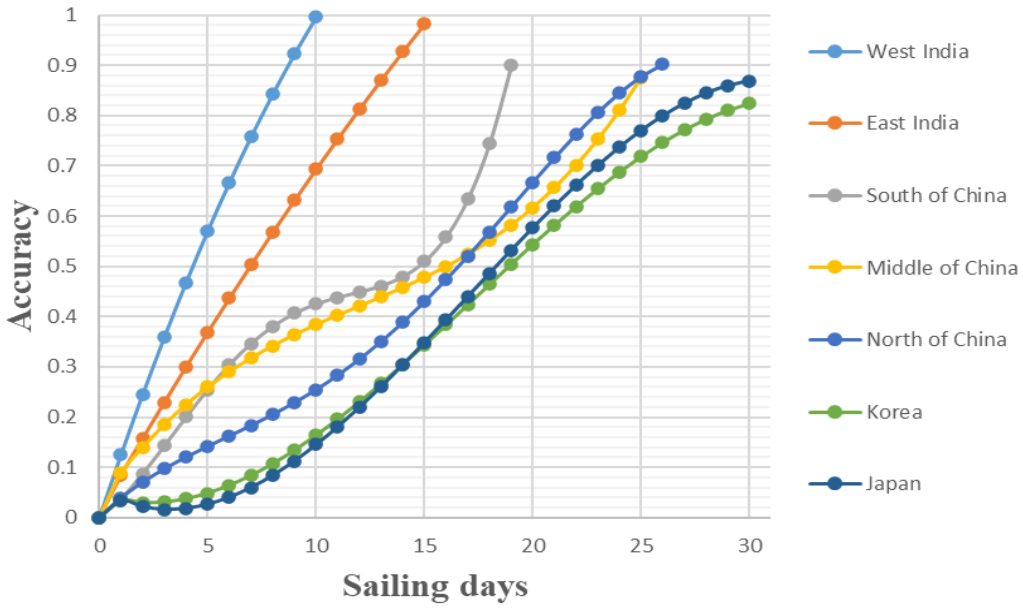


Figure 4. 5 The accuracy between sailing days and destination regions

Note: (Each dot represents the accuracy for every sailing day)

4.3 Model 3-The Neural Network Model (Spatial-Temporal-Based)

4.3.1 Data description

In model 3, we choose 200 voyages from three frequent routes of VLCCs, respectively. The detailed routes are:

Route 1: Ju'aymah Crude & LPG Terminals → US Gulf Lightering Zones. (72 voyages)

Route 2: Ras Tanura → Kiire. (65 voyages)

Route 3: Djeno → Qingdao. (63 voyages)

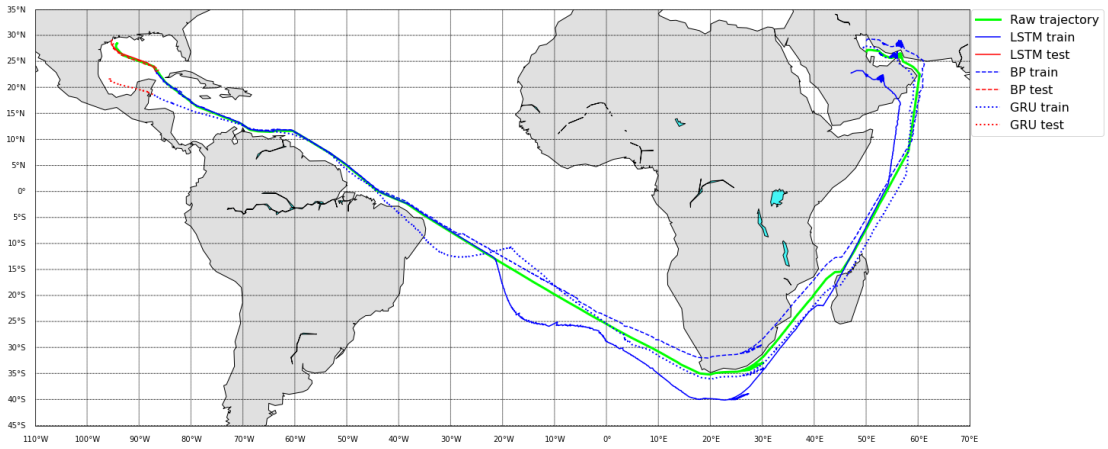
4.3.2 Methodology and metrics

(1) Methodology

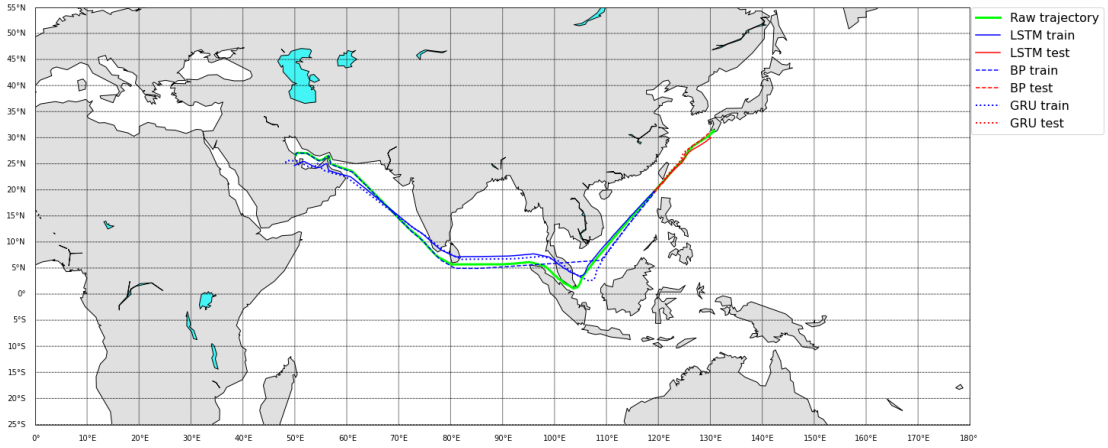
We build the BP, LSTM, GRU neural networks using the Keras library to predict the spatial-temporal trajectory of the last few days for a VLCC. The general parameters

of the neural networks are: one hidden layer of 5 neurons, one output layer of 5 neurons, the activation function for the hidden layer is *Relu*, the activation function for the output layer is linear, loss function is the mean squared error, the optimizer is *adam*, the batch size is 32. When training neural networks, input the time series data of latitude and longitude from the sailing trajectory that has been produced and fit these samples with a nonlinear mapping. Based on the fitting function, the last few days trajectory can be predicted for testing. Moreover, the destination port is regarded as located at the end of the last few days' trajectory.

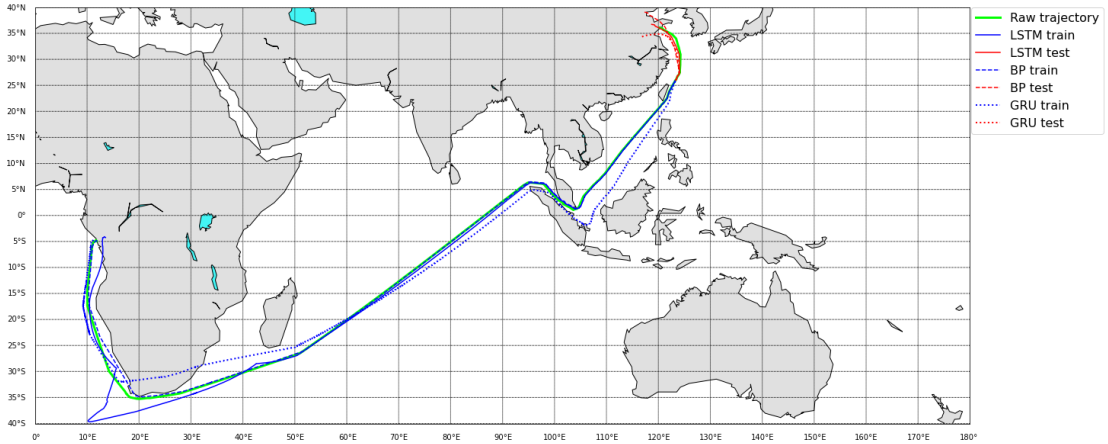
As shown in Figure 4.6, the illustrations of three neural networks present the training trajectory, which is produced by the VLCC that has sailed for more than 20 days, and the last-72h trajectory of the VLCC as the testing trajectory. Here, we focus on the predicted trajectory for testing in the red line. The end location of the predicted trajectory decides the destination port, which is adjacent to crude oil storage and berth facilities to accommodate the VLCC.



(a) Route 1-Ju'aymah Crude & LPG Terminals to US Gulf Lightering Zones



(b) Route 2-Ras Tanura to Kiire



(c) Route 3-Djeno to Qingdao

Figure 4. 6 Illustrations of neural networks for trajectory prediction

(2) Metrics

To measure the difference between the raw trajectory and the predicted trajectory, we adopt the MAE (mean absolute error) and RMSE (root mean squared error) as metrics for longitude/ latitude. The MAE and RMSE are calculated as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|, \tag{12}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}. \tag{13}$$

The correct rate of destination ports can be assessed by:

$$Correct\ rate = \frac{N_{real}}{N_{total}}, \tag{14}$$

where the N_{real} is the number of predicted trajectories that end at the real destination port of VLCC, the N_{total} is the total number of trajectories of the target route.

4.3.3 Results

(1) Assessment of neural networks

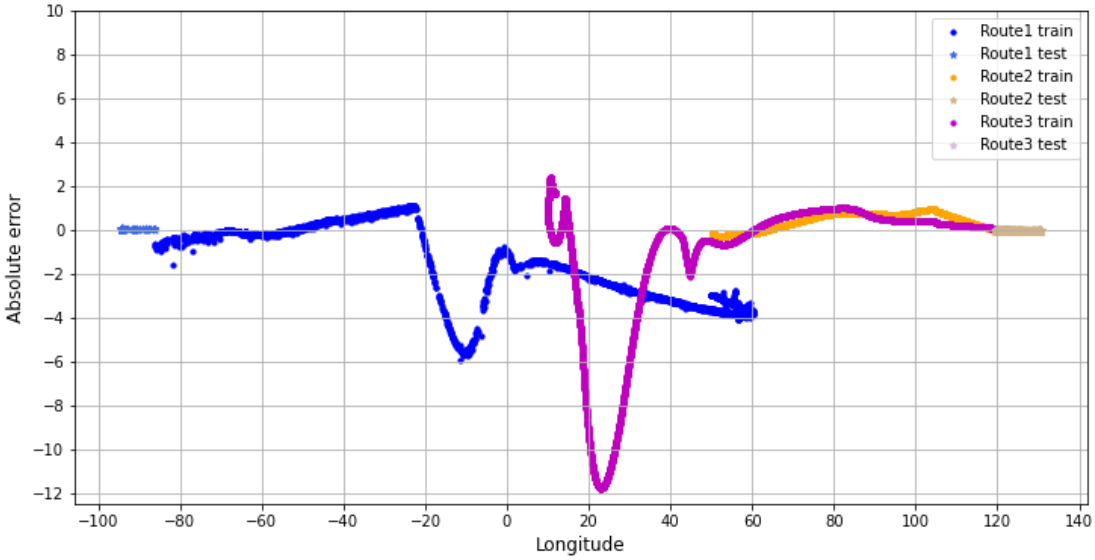
We use BP, LSTM, GRU to simulate the trajectories of three frequent routes. We adopt the VLCC trajectories from starting to the last-72h as the training dataset and the trajectories of the last-72h as the testing dataset. As Table 4.5 shows, the LSTM has the best performance with the minimum average MAE and RMSE for testing trajectories. The average MAE and RMSE of longitude and latitude do not exceed 0.1. Hence, LSTM is selected for the following work.

Table 4. 4 The performance of predicted last-72h trajectories

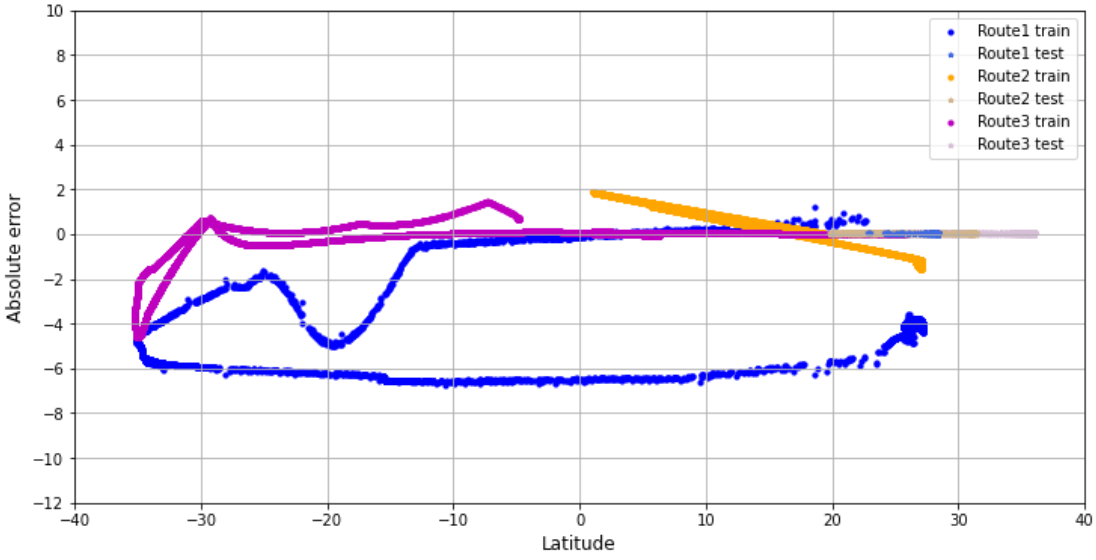
Neural networks	Routes	Longitude		Latitude	
		Ave MAE	Ave RMSE	Ave MAE	Ave RMSE
BP	route 1	0.095	0.096	0.063	0.064
	route 2	0.431	0.47	0.523	0.674
	route 3	0.185	0.148	0.292	0.236
GRU	route 1	1.375	1.384	6.209	6.250
	route 2	0.71	0.904	0.429	0.534
	route 3	1.836	2.212	0.733	0.871
LSTM	**route 1**	**0.068**	**0.067**	**0.059**	**0.097**
	route 2	**0.029**	**0.036**	**0.037**	**0.039**
	route 3	**0.045**	**0.056**	**0.063**	**0.073**

Comparing the training trajectory with the test trajectory using LSTM, we can find the difference of absolute error distributions between them. As shown in Figure 4.7, the absolute error of training trajectory fluctuates in a wide range. One reason is that the sailing time for training is in the long term more than 20 days. Another reason is that

the error is sensitive to the ship's heading, where the longitude and latitude change a lot. However, the absolute error of the testing trajectory is near zero. This result indicates the end location of the testing trajectory is near to the real destination region, and we can try to predict the destination port based on LSTM.



(a) The absolute error of longitude



(b) The absolute error of latitude

Figure 4. 7 The absolute error distribution of locations in LSMT

(2) Destination port prediction performance

We use the LSTM to simulate the 200 trajectories for three routes according to

different prediction time lengths, respectively. We identify the ratio of trajectories ending at the real destination port of VLCC to reflect the correct rate. As shown in Table 4.6, when the time length for prediction is shorter, the correct rate is higher. The correct rate can reach 0.83 by predicting the last-48h trajectory. It implies reliable results can be provided two days in advance before arriving. Therefore, during the last few days of the voyage, the neural network model is helpful to predict the destination port.

Table 4. 5 The relationship between the correct rate and prediction time length

Model	Routes	Correct rate of different prediction time length				
		last-72h trajectory	last-60h trajectory	last-48h trajectory	last-24h trajectory	last-12h trajectory
LSTM	route 1	0.651	0.731	**0.836**	**0.854**	**0.892**
	route 2	0.633	0.714	**0.817**	**0.842**	**0.881**
	route 3	0.648	0.725	**0.828**	**0.853**	**0.887**

4.4 Research Findings and Application of the Destination Port Prediction Models

4.4.1 Research findings

Based on the above contents, we have some meaningful research findings deriving from the prediction framework of models. Before sailing, no trajectory is produced, and the important factor is the historical information of previous ports. It has a positive correlation between the likelihood of prediction result and the number of previous ports. During sailing, the trajectory can be recorded and updated every sailing day. The important factor is the trajectory similarity compared with representative trajectories of

known routes. But the common segments of sailing trajectories have a negative effect on the prediction until the target ship's heading becomes different from those of other routes. Before arrival, the target ship is near the destination port. The important factor is the fitting performance of the sailing trajectory that has been produced during the voyage. This factor determines whether the predicted trajectory of the last few days ends at the destination port.

4.4.2 Application of the destination port prediction models

The description of how to apply our prediction models is necessary in the real world. Given the initial information of an interested VLCC, the model 1 is employed firstly to output a destination port for reference before sailing. When this VLCC starts its voyage, the model 2 begins to work and outputs the destination port for every sailing day. Do not update the original destination port produced by model 1 until the predicting likelihood of model 2 exceeds that of model 1. As the VLCC has finished most of the voyage and slows down below a specific threshold, transfer to the model 3 to predict the trajectory of the last few days and identify the destination port. In general, when the VLCC becomes closer to the destination, the result of model 3 will be more accurate than those of model 1 and 2.

Here, an example can help understand the application process of three models: The target VLCC (IMO 9828950) was ready to depart from the port of Mina al Fahalon on 2020/1/1. The model 1 predicted the destination port was the port of Yokkaichi with a probability of 0.69 before the voyage. When the target VLCC departed, the model 2 gave the predicted destination port every day. At first, the probabilities of these

prediction ports were less than 0.69. Therefore, the port of Yokkaichi as the destination could not be replaced. But after the sixteen days' sailing, the model 2 predicted the destination port was the port of Chiba with a probability of 0.75 lager than 0.69. The destination port was updated. Then the target VLCC tended to slow down on 2020/1/20 and the model 3 were adopted for prediction. The model 3 produced the same prediction result that the destination port is the port of Chiba. Finally, the target VLCC arrived at the port of Chiba on 2020/1/22, which had been predicted correctly by the model.

4.5 Summary of the Chapter

In this chapter, we built three models with different sample distributions to predict the destination port. These models can cover different stages of sailing and provide the prediction results. The model 1 with a high order can predict the destination port before sailing; the model 2 can predict the destination port in the second half of the voyage; the model 3 can predict the destination port 48h in advance before arriving. Besides, we discuss some key points of our models and explain how to apply these models in the real world in detail with an example.

Chapter 5: Conclusion and Future Work

The summary of every chapter has been given at the end of the corresponding chapter. The reviews of these summaries are that: Chapter 1 introduced the research background and motivations. The structure of the whole study was given. Chapter 2 reviewed the related literature to tell how the inspiration of this study came. The theoretical foundations and frameworks of destination port prediction were determined, as well as the stay points recognition and the shape-based similarity distance. Chapter 3 gave the general AIS data preprocessing steps. The optimized CB-SMoT algorithm was developed to detect the sequences of port calls and help segment the trajectory of different voyages for more semantic information, which was the cornerstone of modeling. Chapter 4 proposed and analyzed the prediction framework of three models that were used for different sailing stages. Some important characteristics and findings of these models were discussed and the application processes in the real world were introduced. Based on these highlights, in this chapter, we can make a conclusion of this study and give the possible future work.

5.1 Conclusion of this Study

Spatial-temporal heterogeneity problem reflects the imbalance of supply and demand in tramp shipping. The information opacity of the tramp shipping market increases the operation cost and decreases the management efficiency. To solve this problem, the fundamental and primary work is the destination port prediction. The results of the predicted destination ports provide the supply variation information in the

market. Based on this, the operations research model can be established in the follow-up research.

In this study, destination port prediction of tramp ships is researched as an independent project. The literature is rare about forecasting the designation port directly. At the same time, destination port records in AIS are unreliable and nearly 70% of records are wrong. It is a challenge to predict the destination port. To achieve the research objective, we combine the vessel trajectory analysis and machine learning approaches to develop three models for prediction in different stages of sailing.

The detailed contributions of our study are presented as follows:

(1) For our study, AIS raw data were cleaned and compressed to ensure the data quality and reduce time cost of programs. We cleaned the data by speed-based heuristic filtering algorithm and compressed the data by Douglas–Peucker algorithm. The compressed data size was about 70% of the raw data size.

(2) Extracting the voyage and its trajectory of different routes needed to know the port calls information to determine the departure port and associated destination port. Our AIS data did not give the port calls for every vessel. Therefore, developing an algorithm to recognize port calls became the precondition. We proposed an optimized CB-SMoT algorithm. Combined with the world ports list, our algorithm adopted new definitions of cluster, merging cluster, and extending cluster. It reduced the time complexity to scan the AIS data of a vessel. The recognized port calls and associated cluster of stay points were spatially and temporally disjointed from each

other. To validate our algorithm, we compared it with other algorithms to detect the known sequence of port calls. As a result, our algorithm recognized 86.41% port calls for bulk carriers and 90.63% port calls for tanker ships.

(3) Based on the optimized CB-SMoT algorithm and AIS data of VLCCs, the voyages with trajectories of different routes and sequences of port calls were extracted for feature engineering in machine learning. We depended on both the spatial-temporal trajectory and the semantic trajectory to develop the framework of three machine learning models. The framework covered different stages of a voyage: before sailing, during sailing and before arriving. All the models in our framework were data-driven for prediction. The distribution of data samples was different for every model to enhance the effectiveness.

(4) The model 1 was a high order sequence of port calls model. We proposed this model to predict the destination port before sailing. The most important factor in this model was the previous ports information. We defined the order to indicate the number of previous ports. The results of the random forest classifier showed the accuracy was related to the order. The higher order sequence of port calls of a VLCC was considered the higher accuracy was obtained. When the order was larger than 3, the accuracy was more than 0.80. Therefore, it was suggested to use at least four previous ports of a VLCC to predict its destination.

(5) The model 2 was a trajectory similarity model. We proposed this model to predict

the destination during sailing. When a VLCC sailed on the sea, the trajectory varied with the sailing day. To identify where the VLCC tended to arrive, some known routes were necessary for comparison. Hence, we used the TRACLUS algorithm to produce the representative trajectories that did not really exist. The similarity distance between the sailing trajectory of different sailing days and the representative trajectories was measured by Symmetrized Segment-Path Distance (SSPD). We converted the similarity distance to the likelihood matrix as the important feature. We also considered another two features of the IMO number and the DWT of a VLCC. The results of GBDT showed the accuracy increases with days and exceeds 0.6 after 20 sailing days. However, the accuracy was affected by the shape of a trajectory. At the beginning of sailing, most VLCCs' trajectories had common trajectory segments that implied the probability distribution of destination ports was the same for every VLCC. As a result, it was hard to predict correctly. When the trajectory shape (ship's heading) had a distinguished difference from others, it had a much better prediction performance. We also demonstrated this by the accuracy variation of prediction between sailing days and seven different destination regions.

(6) The model 3 was a neural network model. We proposed this model to predict the destination port before arriving. The neural networks were trained with the sailing trajectory that had been produced during the voyage. Hence, it was used for prediction of the last few days' trajectory before a VLCC's arrival. The results of

LSTM had the minimum error of latitude and longitude for three frequent routes. The ending location of the predicted trajectory by LSTM decided the destination port. Compared with the real destination port, the correct rate was more than 0.83 by predicting the last-48h trajectory. It was available to decide the destination port two days in advance before arriving.

(7) Based on the findings of three models, we also gave the guideline of our prediction framework for application with an example. The model 1 provided the initial prediction result of the interested VLCC before its voyage. When the VLCC started to sail, the model 2 gave the prediction result for every sailing day. The initial destination port produced by model 1 was not replaced until the predicting likelihood of model 2 exceeded that of model 1. When the VLCC had finished most of the voyage and slowed down below a specific threshold, it was transferred to the model 3 for prediction.

Overall, we design a comprehensive framework with multi-models for destination port prediction. This framework represents a complete data mining process to realize the goal. Our study can be improved according to different purposes and extended with more functionality.

5.2 Future Work

The future research of this study will focus on modeling to solve the routing optimization problem for tramp shipping. Our destination port prediction framework

will provide the dynamic information of ship movements in the market, which is a very significant factor. In the real world, it is a game with incomplete information. As a result, the strategies of different shipping companies are unknown and hard to develop the model when missing the destination port information of competitors. Therefore, our trajectory-based models that use the AIS data to mine the dynamic destination port variation can bring a new avenue to develop the operations research model (see Appendix C for our initial model).

Moreover, our study still has some limitations to be explored and completed in the future. Firstly, this study thesis lacks a systematic extension to other tramp ship types. In the future, it is meaningful to extend this work not only for VLCCs but for bulk carriers and other tanker ships. Secondly, it is suggested to collect the data of different years and companies to repeat our proposed models. Thirdly, it is necessary to consider the draught to infer whether the VLCC is in ballast status, which can help narrow the prediction range with the port characteristics of loading or unloading. We believe our study has opened a path to others.

Bibliography

Aronov, B., Har-Peled, S., Knauer, C., Wang, Y., & Wenk, C. (2006, September). Fréchet distance for curves, revisited. In European symposium on algorithms (pp. 52-63). Springer, Berlin, Heidelberg.

Ashbrook, D., & Starner, T. (2002, October). Learning significant locations and predicting user movement with GPS. In Proceedings. Sixth International Symposium on Wearable Computers, (pp. 101-108). IEEE.

Bermingham, L., & Lee, I. (2018). A probabilistic stop and move classifier for noisy GPS trajectories. Data Mining and Knowledge Discovery, 32(6), 1634-1662.

Besse, P. C., Guillouet, B., Loubes, J. M., & Royer, F. (2016). Review and perspective for distance-based clustering of vehicle trajectories. IEEE Transactions on Intelligent Transportation Systems, 17(11), 3306-3317.

Birant, D., & Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial–temporal data. Data & knowledge engineering, 60(1), 208-221.

Chen, L., Lv, M., & Chen, G. (2010). A system for destination and future route prediction based on trajectory mining. Pervasive and Mobile Computing, 6(6), 657-676.

Chen, L., Özsu, M. T., & Oria, V. (2005, June). Robust and fast similarity search for moving object trajectories. In Proceedings of the 2005 ACM SIGMOD international conference on Management of data (pp. 491-502).

Chen, W., Ji, M. H., & Wang, J. M. (2014). T-DBSCAN: A Spatiotemporal Density Clustering for GPS Trajectory Segmentation. International Journal of Online Engineering, 10(6).

Chen, Z., Guo, D., & Lin, Y. (2020). A deep Gaussian process-based flight trajectory prediction approach and its application on conflict detection. Algorithms, 13(11), 293.

Damiani, M. L., Issa, H., & Cagnacci, F. (2014, November). Extracting stay regions with uncertain boundaries from GPS trajectories: a case study in animal ecology. In Proceedings of the 22nd ACM SIGSPATIAL international conference on

advances in geographic information systems (pp. 253-262).

Fu, Z., Tian, Z., Xu, Y., & Qiao, C. (2016). A two-step clustering approach to extract locations from individual GPS trajectory data. ISPRS International Journal of Geo-Information, 5(10), 166.

Guo, S., Liu, C., Guo, Z., Feng, Y., Hong, F., & Huang, H. (2018, June). Trajectory prediction for ocean vessels base on K-order multivariate Markov chain. In International Conference on Wireless Algorithms, Systems, and Applications (pp. 140-150). Springer, Cham.

Hennig, F., Nygreen, B., Christiansen, M., Fagerholt, K., Furman, K. C., Song, J., ... & Warrick, P. H. (2012). Maritime crude oil transportation–a split pickup and split delivery problem. European Journal of Operational Research, 218(3), 764-774.

Huang, G., He, J., Zhou, W., Huang, G. L., Guo, L., Zhou, X., & Tang, F. (2016). Discovery of stop regions for understanding repeat travel behaviors of moving objects. Journal of Computer and System Sciences, 82(4), 582-593.

Huang, L., Wen, Y., Guo, W., Zhu, X., Zhou, C., Zhang, F., & Zhu, M. (2020). Mobility pattern analysis of ship trajectories based on semantic transformation and topic model. Ocean Engineering, 201, 107092.

Jia, H., Prakash, V., & Smith, T. (2019). Estimating vessel payloads in bulk shipping using AIS data. International Journal of Shipping and Transport Logistics, 11(1), 25-40.

Jin, X., Yang, Y., & Qiu, X. (2017, July). Framework of Frequently Trajectory Extraction from AIS Data. In Proceedings of the 2017 The 7th International Conference on Computer Engineering and Networks (pp. 22-23).

Keogh, E., & Ratanamahatana, C. A. (2005). Exact indexing of dynamic time warping. Knowledge and information systems, 7(3), 358-386.

Kwakye, M. M. (2019). Semantic data warehouse modelling for trajectories. arXiv preprint arXiv:1904.06484.

Langran, G., & Chrisman, N. R. (1988). A framework for temporal geographic information. Cartographica: The International Journal for Geographic Information

and Geovisualization, 25(3), 1-14.

Lee, J. G., Han, J., & Whang, K. Y. (2007, June). Trajectory clustering: a partition-and-group framework. In Proceedings of the 2007 ACM SIGMOD international conference on Management of data (pp. 593-604).

Li, W., Zhang, C., Ma, J., & Jia, C. (2019, July). Long-term vessel motion predication by modeling trajectory patterns with AIS data. In 2019 5th International Conference on Transportation Information and Safety (ICTIS) (pp. 1389-1394). IEEE.

Lin, B., & Su, J. (2008). One way distance: For shape based similarity search of moving object trajectories. GeoInformatica, 12(2), 117-142.

Lin, Y., Zhang, J. W., & Liu, H. (2018). An algorithm for trajectory prediction of flight plan based on relative motion between positions. Frontiers of Information Technology & Electronic Engineering, 19(7), 905-916.

Liu, C., & Chen, X. (2014). Vessel track recovery With incomplete AIS data using tensor CANDECOM/PARAFAC decomposition. The journal of navigation, 67(1), 83-99.

Liu, K., & Yang, J. (2009). Trajectory distance metric based on edit distance. Journal of Shanghai jiaotong university, 43(11), 1725-1729.

Luo, T., Zheng, X., Xu, G., Fu, K., & Ren, W. (2017). An improved DBSCAN algorithm to detect stops in individual trajectories. ISPRS International Journal of Geo-Information, 6(3), 63.

Ma, W., Wu, Z., Yang, J., & Li, W. (2014, August). Vessel motion pattern recognition based on one-way distance and spectral clustering algorithm. In International Conference on Algorithms and Architectures for Parallel Processing (pp. 461-469). Springer, Cham.

Magnussen, B. B., Bläser, N., Jensen, R. M., & Ylänen, K. (2021, September). Destination Prediction of Oil Tankers Using Graph Abstractions and Recurrent Neural Networks. In International Conference on Computational Logistics (pp. 51-65). Springer, Cham.

Mao, S., Tu, E., Zhang, G., Rachmawati, L., Rajabally, E., & Huang, G. B. (2018). An

automatic identification system (AIS) database for maritime trajectory prediction and data mining. In Proceedings of ELM-2016 (pp. 241-257). Springer, Cham.

Mascret, A., Devogele, T., Le Berre, I., & Hénaff, A. (2006). Coastline matching process based on the discrete Fréchet distance. In Progress in Spatial Data Handling (pp. 383-400). Springer, Berlin, Heidelberg.

Min, D., Zhilin, L., & Xiaoyong, C. (2007). Extended Hausdorff distance for spatial objects in GIS. International Journal of Geographical Information Science, 21(4), 459-475.

Morzy, M. (2007, July). Mining frequent trajectories of moving objects for location prediction. In International workshop on machine learning and data mining in pattern recognition (pp. 667-680). Springer, Berlin, Heidelberg.

Murray, B., & Perera, L. P. (2018, September). A data-driven approach to vessel trajectory prediction for safe autonomous ship operations. In 2018 Thirteenth International Conference on Digital Information Management (ICDIM) (pp. 240-247). IEEE.

Naderivesal, S., Kulik, L., & Bailey, J. (2019). An effective and versatile distance measure for spatiotemporal trajectories. Data Mining and Knowledge Discovery, 33(3), 577-606.

Pallotta, G., Vespe, M., & Bryan, K. (2013). Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction. Entropy, 15(6), 2218-2245.

Palma, A. T., Bogorny, V., Kuijpers, B., & Alvares, L. O. (2008, March). A clustering-based approach for discovering interesting places in trajectories. In Proceedings of the 2008 ACM symposium on Applied computing (pp. 863-868).

Parent, C., Spaccapietra, S., & Zimanyi, E. (1999, November). Spatio-temporal conceptual models: data structures+ space+ time. In Proceedings of the 7th ACM international symposium on Advances in geographic information systems (pp. 26-33).

Pelekis, N., Andrienko, G., Andrienko, N., Kopanakis, I., Marketos, G., & Theodoridis,

Y. (2012). Visually exploring movement data via similarity-based analysis. Journal of Intelligent Information Systems, 38(2), 343-391.

Pelekis, N., Kopanakis, I., Marketos, G., Ntoutsi, I., Andrienko, G., & Theodoridis, Y. (2007, June). Similarity search in trajectory databases. In 14th International Symposium on Temporal Representation and Reasoning (TIME'07) (pp. 129-140). IEEE.

Pelekis, N., Theodoulidis, B., Kopanakis, I., & Theodoridis, Y. (2004). Literature review of spatio-temporal database models. The Knowledge Engineering Review, 19(3), 235-274.

Peng, X. Y., Men, Z. G., & Liu, C. D. (2010). Volterra-kernel estimation and its application based on Kalman filtering algorithm. Systems Engineering and Electronics, 32(11).

Perera, L. P., & Soares, C. G. (2010, November). Ocean vessel trajectory estimation and prediction based on extended kalman filter. In The Second International Conference on Adaptive and Self-Adaptive Systems and Applications (pp. 14-20).

Peuquet, D. J. (2001). Making space for time: Issues in space-time data representation. GeoInformatica, 5(1), 11-32.

Prabhala, B., & La Porta, T. (2015, April). Spatial and temporal considerations in next place predictions. In 2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) (pp. 390-395). IEEE.

Qiao, S., Shen, D., Wang, X., Han, N., & Zhu, W. (2014). A self-adaptive parameter selection trajectory prediction approach via hidden Markov models. IEEE Transactions on Intelligent Transportation Systems, 16(1), 284-296.

Reap, R. M. (1972). An operational three-dimensional trajectory model. Journal of Applied Meteorology and Climatology, 11(8), 1193-1202.

Schuessler, N., & Axhausen, K. W. (2009). Processing raw data from global positioning systems without additional information. Transportation Research Record, 2105(1), 28-36.

Shahir, A. Y., Tayebi, M. A., Glässer, U., Charalampous, T., Zohrevand, Z., & Wehn, H.

(2019, December). Mining vessel trajectories for illegal fishing detection. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 1917-1927). IEEE.

Sheng, P., & Yin, J. (2018). Extracting shipping route patterns by trajectory clustering model based on automatic identification system data. Sustainability, 10(7), 2327.

Sirimanne, S. N., Hoffman, J., Juan, W., Asariotis, R., Assaf, M., Ayala, G., ... & Premti, A. (2019, October). Review of maritime transport 2019. In United Nations Conference on Trade and Development, Geneva, Switzerland.

Spaccapietra, S., Parent, C., Damiani, M. L., de Macedo, J. A., Porto, F., & Vangenot, C. (2008). A conceptual view on trajectories. Data & knowledge engineering, 65(1), 126-146.

Sun, L., & Zhou, W. (2017, March). Vessel motion statistical learning based on stored ais data and its application to trajectory prediction. In Proceedings of the 2017 5th International Conference on Machinery, Materials and Computing Technology (ICMMCT 2017), Beijing, China (pp. 25-26).

Tao, J., Xu, J., Wang, C., & Chawla, N. V. (2017, April). HoNVis: Visualizing and exploring higher-order networks. In 2017 IEEE Pacific Visualization Symposium (PacificVis) (pp. 1-10). IEEE.

Tiesyte, D., & Jensen, C. S. (2008, November). Similarity-based prediction of travel times for vehicles traveling on known routes. In Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems (pp. 1-10).

Tvedt, J. (2011). Short-run freight rate formation in the VLCC market: A theoretical framework. Maritime Economics & Logistics, 13(4), 442-455.

Üney, M., Millefiori, L. M., & Braca, P. (2019, May). Data driven vessel trajectory forecasting using stochastic generative approachs. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 8459-8463). IEEE.

Vandecasteele, A., Devillers, R., & Napoli, A. (2014). From movement data to objects behavior using semantic trajectory and semantic events. Marine Geodesy, 37(2),

126-144.

Vespe, M., Visentini, I., Bryan, K., & Braca, P. (2012). Unsupervised learning of maritime traffic patterns for anomaly detection.

Villa, P., & Camossi, E. (2011). Semantic-based anomalous pattern detection from maritime trajectories. In MAD 2011 Workshop Proceedings (p. 139).

Vlachos, M., Kollios, G., & Gunopulos, D. (2002, February). Discovering similar multidimensional trajectories. In Proceedings 18th international conference on data engineering (pp. 673-684). IEE

Wen, M., Ropke, S., Petersen, H. L., Larsen, R., & Madsen, O. B. (2016). Full-shipload tramp ship routing and scheduling with variable speeds. Computers & Operations Research, 70, 1-8.

Wen, Y. T., Lai, C. H., Lei, P. R., & Peng, W. C. (2014, July). Routeminer: Mining ship routes from a massive maritime trajectories. In 2014 IEEE 15th International Conference on Mobile Data Management (Vol. 1, pp. 353-356). IEEE.

Worboys, M., & Hornsby, K. (2004, October). From objects to events: GEM, the geospatial event model. In International conference on geographic information science (pp. 327-343). Springer, Berlin, Heidelberg.

Xu, T., Cai, F. J., Hu, Q. Y., & Chun, Y. (2014). Research on estimation of AIS vessel trajectory data based on Kalman filter algorithm. Mod. Electron. Tech, 5, 97-100.

Xu, T., Liu, X., & Yang, X. (2011, September). Ship Trajectory online prediction based on BP neural network algorithm. In 2011 International Conference of Information Technology, Computer Engineering and Management Sciences (Vol. 1, pp. 103-106). IEEE.

Xu, T., Liu, X., & Yang, X. (2012). BP neural network-based ship track real-time prediction. Journal of Dalian Maritime University, 38(1), 9-11.

Yang, J., Xu, J., Xu, M., Zheng, N., & Chen, Y. (2014, November). Predicting next location using a variable order Markov model. In Proceedings of the 5th ACM SIGSPATIAL International Workshop on GeoStreaming (pp. 37-42).

Yi, S., Li, H., & Wang, X. (2016, October). Pedestrian behavior understanding and

prediction with deep neural networks. In European Conference on Computer Vision (pp. 263-279). Springer, Cham.

Yin, J., Luo, M., & Fan, L. (2017). Dynamics and interactions between spot and forward freights in the dry bulk shipping market. Maritime Policy & Management, 44(2), 271-288.

Ying, J. J. C., Lu, E. H. C., Lee, W. C., Weng, T. C., & Tseng, V. S. (2010, November). Mining user similarity from semantic trajectories. In Proceedings of the 2nd acm sigspatial international workshop on location based social networks (pp. 19-26).

Yu, K., Liang, X. F., Li, M. Z., Chen, Z., Yao, Y. L., Li, X., ... & Teng, Y. (2021). USV path planning method with velocity variation and global optimisation based on AIS service platform. Ocean Engineering, 236, 109560.

Zaccone, R., & Martelli, M. (2018, October). A random sampling based algorithm for ship path planning with obstacles. In Proceedings of the International Ship Control Systems Symposium (iSCSS) (Vol. 2, p. 4).

Zhang, C., Bin, J., Wang, W., Peng, X., Wang, R., Halldearn, R., & Liu, Z. (2020). AIS data driven general vessel destination prediction: A random forest based approach. Transportation Research Part C: Emerging Technologies, 118, 102729.

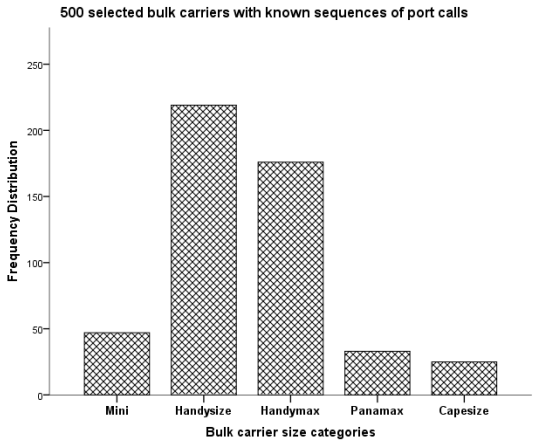
Zhang, W., Sun, L., Wang, X., Huang, Z., & Li, B. (2019). SEABIG: A deep learning-based method for location prediction in pedestrian semantic trajectories. IEEE Access, 7, 109054-109062.

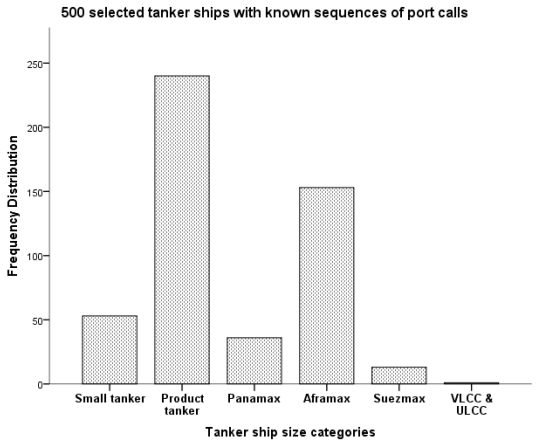
Zhao, P., Qin, K., Ye, X., Wang, Y., & Chen, Y. (2017). A trajectory clustering approach based on decision graph and data field for detecting hotspots. International Journal of Geographical Information Science, 31(6), 1101-1127.

Zimmermann, M., Kirste, T., & Spiliopoulou, M. (2009, November). Finding stops in error-prone trajectories of moving objects with time-based clustering. In International conference on intelligent interactive assistance and mobile multimedia computing (pp. 275-286). Springer, Berlin, Heidelberg.

Appendix

Appendix A - the selected tramp ships to measure the performance of port calls recognition algorithms in Chapter 3. The Figure shows the frequency distribution of size categories for both ship types- bulker carrier and tanker ship.





Appendix B - the voyage distribution of VLCCs in Chapter 4 based on the regional pairs of departure and destination ports. The Table shows the statistics of voyages among different geographic regions.

Destination region / Departure region	China North (CN-N)	China Middle (CN-M)	China South (CN-S)	Japan (JP)	Korea (KR)	West coast India (WCI)	East coast India (ECI)
Middle East and Gulf (MEG)	**150**	**91**	**69**	**73**	**184**	**289**	**16**
Caribbean Sea (CBS)	1	1	1	1	/	/	/
South American (SA)	14	1	1	1	/	/	/
United Kingdom Continent (UKC)	13	1	0	1	/	/	/
United States Gulf (USG)	18	4	5	5	3	5	2
West African (WAF)	43	1	10	10	/	/	4

Appendix C - The initial operations research model of routing optimization problem for reference in the future work in Chapter 5.

The selected variables of our operations research model include the cargo category and tonnage information of the target destination port, freight rate, bunker price, and fixed cost. The model objective is the maximum profit for a shipping company by the optimal speed. To simplify the model, we give some hypotheses:

H1: One interested destination port is assessed every time.

H2: The maximum profit is updated every day.

H3: The commodity contract is first come first served.

Based on above hypotheses, the operations research model can be written as:

$$Maximum\ \pi = r(Q-q) - \varepsilon_{bunker}\, m_{oil} - C\frac{S}{V}, \tag{15}$$

Subject to:

$$q = \sum_{i=1}^{k} \theta_k W_k P_k, \tag{16}$$

$$\theta_k = \begin{cases} 0, & if\ \dfrac{S_k}{V_k} < \dfrac{S}{V} \\ 1, & if\ \dfrac{S_k}{V_k} \geqslant \dfrac{S}{V} \end{cases} \tag{17}$$

$$m_{oil} = \alpha V^3, \tag{18}$$

$$r, S, V, C, Q, Q-q \geqslant 0. \tag{19}$$

Where π is the profit, r is the freight rate, Q is the total cargo tonnages in the target destination port, q is the tonnages occupied by competitors, $\varepsilon_{\text{bunker}}$ is the unit bunker price in the market, m_{oil} is the estimated oil consumption, C is the fixed cost of every day, S is the distance to the interested destination port, V is the average speed of a sailing day. For the constraint conditions (16) ~ (19), k is the number of vessels in ballast that are estimated to the interested destination port in the same sea aera. θ_k is the binomial function to identify whether the competitor can acquire the cargo. W_k is the

DWT of another company' vessel and P_k is the predicted likelihood to the same port. m_{oil} is calculated by the typical cubic law, where α is acting as the scale parameter.

Combined with the above framework of destination port prediction models, the key factors of k, θ_k , W_k , P_k in the constraint equation (16) and (17) can be achieved for different sailing stages of the competitive vessels. It helps to find the closed-form solutions.