NOVEL MARITIME TRAFFIC ANALYSIS TECHNIQUES TO ENHANCE MARITIME SITUATIONAL AWARENESS

by

Lubna Mohamed Eljabu

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
August 2024

Contents

List of Tables

List of Figures

Abstract

The maritime domain is characterized by complex vessel movements with intricate spatiotemporal patterns and interdependencies. However, Automatic Identification System (AIS) data, despite being a rich real-time source of vessel positioning, violates the assumption of independent and identically distributed (i.i.d.) data points due to inherent temporal, spatial, and network dependencies. Traditional data analysis techniques under such assumptions encounter limitations when applied to AIS data; ignoring dependencies in data points can result in inaccurate clustering or pattern detection, underestimation of uncertainty in predictions, and biased parameter estimates in models assuming independent observations. This thesis aims to develop advanced data-driven frameworks and methodologies that leverage time-series analysis, spatial data mining, and network science to develop a novel model for destination port prediction. The objective is to explore the potential of supporting port authorities in forecasting traffic inflow and outflow within their local environment by monitoring AIS messages.

This thesis first presents a novel approach to enrich trajectory representations by integrating AIS data with port information and segmenting trajectories based on port points, thereby homogenizing vessel movement patterns. A semi-supervised clustering algorithm is then proposed for these trajectory segments, employing contextual data to derive clustering constraints. This algorithm effectively identifies preferred vessel paths, and port-to-port traffic flows directly from AIS trajectories. Building upon these clusters, a data-driven method is developed where trajectory patterns dictate the network topology. This scalable graph adapts to different geographical regions and traffic densities, eliminating the need for static route networks. Utilizing the traffic network representation and trajectory similarity measures, a prediction method is developed to forecast vessel destinations based on recent movements. Evaluations on real-world AIS datasets demonstrate promising results, with the model expressing uncertainty through probability distributions for potential destinations and dynamically updating these probabilities as the vessel progresses. This research advances maritime analytics by developing data-driven methodologies that model intricate spatiotemporal patterns and dependencies in AIS data, account for the complex connectivity of maritime traffic, and enable enhanced prediction capabilities. By overcoming the limitations of traditional techniques, this work contributes to the state-of-the-art in maritime data analytics and decision support systems.

List of Abbreviations Used

AIS Automatic Identification System

BIRCH Balanced Iterative Reducing and Clustering using Hierarchies

CL Curve Length

COS Course Over Ground

DBSCAN Density-based spatial clustering of applications with noise

DFD Discrete Fréchet Distance

DP Douglas-Peucker

DTW Dynamic Time Warping

GIS Geographic Information System

MMSI Maritime Mobile Service Identity

MPA Marine Protected Area

MSA Maritime Situational Awareness

OPTICS Ordering Points To Identify the Clustering Structure

PCM Partial Curve Mapping

RRoT Reference Route of Trajectory

SOG Speed Over Ground

SPTCLUST Spatial Clustering of Vessel Trajectories

Glossary

bearing : a bearing is the clockwise angle between two points on the earth's surface, measured from true north, and calculated using spherical trigonometry to account for earth's curvature. Let point B and C have positions $(\text{lat}^1, \text{lon}^1)$ and $(\text{lat}^2, \text{lon}^2)$, respectively. Let point A be the North Pole. The angle Δ is the difference between the longitudes. The angle β, representing the bearing from B to C, $\beta_{B\to C}$, can be calculated using the following relations: $\beta = atan2(X, Y)$, where X and Y can be calculated as, $X = cos(lat^2) \times sin\Delta$, $\Delta = lon^1 - lon^2$, $Y = cos(lat^1) \times sin(lat^2) - sin(lat^1) \times cos(lat^2) \times cos\Delta$ [89]. If $X = 0$ and $Y = 0$, indicating that two geographic points are identical, the bearing is defined to be zero. xvii, 5, 19, 25, 57, 58, 65, 66, 90

cluster : a cluster, C_i, is a set of trajectory segments, where all associated origin and destination pairs (O_i, D_i), representing the starting and ending locations of its segments are identical, and this pair is unique to that cluster and not found in any other cluster. Let C_i represent a cluster, and $\mathcal{S}$ denote the set of all trajectory segments. Each cluster C_i consists of trajectory segments s such that: $C_i = \{s \in \mathcal{S} \mid (O(s), D(s)) = (O_i, D_i)\ \forall\ s \in \mathcal{J}_i\}$. $\mathcal{J}_i$ is the index set of all segments in cluster C_i. (O_i, D_i) is the unique origin-destination pair for the cluster C_i. Thus $C_i \not\subseteq C_j\ \forall\, i \neq j, \therefore (O_i, D_i) \neq (O_j, D_j)$. 5, 6, 20, 24, 55–57, 65, 68, 75, 76, 78, 79, 85, 86, 89, 93, 95, 97, 99, 100

completeness : the degree to which all trajectory segments following same direction between two ports are grouped into the same cluster. 72–74, 76

de facto maritime routes : refer to the commonly followed navigation routes taken by vessels in a particular maritime region when navigating from one port to the next port. These routes are not formally designated by any regulatory authority but have emerged over time based on factors such as navigational efficiency, safety, trade routes, and historical usage. 7, 20, 24, 27, 55

fragment : A fragment of a trajectory segment, fr, can be defined as follows: Let $s = \{(l_1, t_1), (l_2, t_2), \ldots, (l_n, t_n)\}$ denote a trajectory segment, where l_i represents the location at time t_i. A fragment fr of the trajectory segment s is a subsequence $\{(l_j, t_j), (l_{j+1}, t_{j+1}), \ldots, (l_k, t_k)\}$, where j is the index of the first location in the segment and k is the index of the last location before reaching the destination port. Thus, a fragment starts from the origin port but does not extend to the destination port, capturing a portion of the trajectory segment. 91, 92, 95, 97, 99

homogeneity : the degree to which an individual cluster includes only trajectory segments that follow one direction between two ports. 72–74, 76

linear interpolation : involves reconstructing a continuous trajectory or trajectory segment from AIS data by uniformly sampling spatiotemporal positions along its length and estimating values between two known points. Specifically, linear interpolation of a vessel trajectory estimates the vessel's position at a given time by assuming linear motion between two known positions. Given two data points (t_1, l_1) and (t_2, l_2): $l(t) = l_1 + \frac{(t-t_1)}{(t_2-t_1)} \cdot (l_2 - l_1)$, where $(t_1 \leq t \leq t_2)$. For positions $(l_1 = (x_1, y_1))$ and $(l_2 = (x_2, y_2))$: $(x(t) = x_1 + \frac{(t-t_1)}{(t_2-t_1)} \cdot (x_2 - x_1)$, $y(t) = y_1 + \frac{(t-t_1)}{(t_2-t_1)} \cdot (y_2 - y_1))$. 26, 44, 68

location (l): is a geolocation defined as, $l = \langle x_i, y_i \rangle$, where x_i represents the latitude, and y_i represents the longitude of the vessels' location. xvi, xviii, 4, 6, 16, 62, 85, 91, 95, 100

maritime traffic network : is represented as a directed graph, where nodes represent ports and edges represent the common routes taken by vessels between ports. This network topology, inferred from trajectory data, delineates the structure and connectivity patterns of maritime routes and traffic flows, derived directly from vessel movement trajectories captured by the Automatic Identification System (AIS) data. 5, 20, 45, 56, 67, 70, 73, 78, 79, 84, 85, 89, 100

must-link constraints : are denoted as $C = (s_i, s_j)$ where instances s_i and s_j from the dataset $\mathcal{S}$ must be assigned to the same cluster. The collection of must-link constraints is represented as $C_=$ [39]. 19, 25, 53, 57

origin and destination : the origin and destination points of a trajectory segment are defined as follows: center geolocation of the nearest port to the first point in a segment is designated as the *origin point*, O, while the center geolocation of the nearest port to the last point of a segment is designated as the *destination Point*, D. xv, xvii, 4, 25, 43, 58, 63, 72, 73, 78, 89

port : is represented as geometric point or area on the map, indicating its location or boundaries. It serves as important node in the maritime traffic network where vessels originate, terminate, or make port calls. 4, 19, 42, 48, 56, 57, 59, 63, 65, 78, 89

reference route : is a representative route comprising a sequence of locations linking two ports within the maritime traffic network (i.e, geospatial line). It is a mean trajectory segment that indicate the common route within a cluster of trajectory segments. 44, 46–48, 50, 51, 58, 70, 73, 77, 78, 84, 85, 89, 90, 92–94, 97, 99, 100

segment s_i: a trajectory segment, or route, is a sequence of locations within a trajectory, starting at one port (origin) and ending at the next port (destination). The trajectory is divided based on partitioning positions into consecutive segments, denoted as s_i, where $s_i = \langle l_j, \ldots, l_k \rangle$, with $j \geq 0$ and $k \leq n$. This process of dividing a trajectory into segments is known as trajectory segmentation. xv, 5, 19, 20, 25, 38, 43, 44, 46, 47, 50, 55, 56, 58, 59, 63, 64, 67, 69, 70, 72, 73, 75, 77, 78, 84–94, 99, 100

segments' endpoints' similarity : the measure of positional and directional similarity between trajectory segments, s_1 and s_2, is determined by the bearing values of their origin and destination points. Given origin points $(\text{lat}^1, \text{lon}^1)$ for O_1 and $(\text{lat}^2, \text{lon}^2)$ for O_2, for two segments, the bearing from O_1 to O_2 is given

by $\beta_{O_1 \to O_2}$. Similarly, the bearing from D_1 to D_2 is given by $\beta_{D_1 \to D_2}$. The pair of bearings ($\beta_{O_1 \to O_2}$, $\beta_{D_1 \to D_2}$) collectively defines the overall directional similarity between the two segments, which is referred to as the pair of bearings of the segments' endpoints. 5, 25, 75

semi-supervised : semi-supervised clustering is a variant of the classic clustering problem that incorporates background knowledge into the clustering process. This knowledge typically involves specifying whether instances should be grouped together (must-link) or kept separate (cannot-link). In the literature, this problem is commonly referred to as Constrained Clustering (CC) [39]. 5, 19, 25, 53, 57, 78

similarity : a distance generalization quantifies the degree of resemblance between two trajectories or two segments, denoted as $dist(\tau_1, \tau_2)$. Specifically, in the context of destination port prediction in this study, $dist(s_i, fr_j)$ signifies the distance between a trajectory segment, s_i, and a fragment of another trajectory segment, fr_j. A greater value indicates less similarity between the two patterns. 20, 29, 30, 39, 46, 48, 51, 53, 55, 75, 78, 85, 86, 88, 91

trajectory (τ): is a time-ordered sequence of AIS records of a moving vessel, $\tau = \{(MMSI, \{(l_1, t_1, f_1), \ldots, (l_{|\tau|}, t_{|\tau|})\}, f_{|\tau|})\}$, where $MMSI$ the maritime mobile service identity, a unique 9-digit number uniquely identify a ship. l_i is vessel location at time t_i, and f_i is a feature vector describing the vessel dynamics and characteristics at time $t_i, t_i < t_{i+1} \, \forall \, i \in \{1, ..., \mid \tau \mid\}$. The features vector includes: cos, sog, vessel_type, etc. 1, 6, 16, 19, 36, 39, 43, 44, 48, 58, 59, 62, 64, 68, 78

Acknowledgements

I would like to express my sincere gratitude to all those who have supported and guided me throughout this journey of pursuing a doctoral degree.

First and foremost, I am deeply indebted to my supervisor, Professor Evangelos E. Milios, whose unwavering guidance, invaluable insights, and constant encouragement have been instrumental in shaping this research. His expertise, patience, and belief in my abilities have been a constant source of motivation, pushing me to strive for excellence.

I would also like to extend my heartfelt thanks to my previous supervisor, Professor Stan Matwin, whose initial guidance and support were instrumental in laying the foundation for this research. His early mentorship played a crucial role in setting me on this path, and I am grateful for the knowledge and skills I gained under his supervision.

I am also grateful to the members of my dissertation committee, Dr. Vlado Keselj, Dr. Yannick Marchand, for their time, constructive criticism, and valuable suggestions that have significantly improved the quality of my work. I would also like to express my gratitude to Dr. Howard J. Hamilton for kindly agreeing to serve as my external thesis examiner and for dedicating his valuable time to reviewing my thesis.

I would like to express my sincere gratitude to Dr. Mohammad Etemad for his invaluable collaboration, insightful discussions, and unwavering support throughout this journey. His expertise and dedication have been instrumental in shaping this work. Furthermore, I would like to acknowledge Dr. Gabriel Spadon De Souza, for his thought-provoking discussions and constructive feedback, which have significantly enriched the quality of this research.

To my beloved parents and my husband, words cannot express my gratitude for your unconditional love, encouragement, and unwavering belief in me. Your constant support and understanding have been the driving force behind my perseverance and determination.

Chapter 1

Introduction

Marine shipping is essential for global freight transportation, with 90% of commodity shipments being transported by sea due to its economic benefits [17, 40, 52]. However, as the world's population continues to increase, demand for goods will increase, leading to more maritime traffic. The implications of maritime traffic on the economy, environment, safety, and security have strengthened the need to enhance Maritime Situational Awareness (MSA) [10]. *Maritime situational awareness* can be thought of as maintaining constant awareness of the surroundings, understanding the events happening around, and anticipating their potential impact on vessels [32].

The primary challenge facing Maritime Situational Awareness (MSA) is the efficient transformation of vast amounts of spatio-temporal data into actionable and dependable information for decision-making. Bridging this gap between raw data and end-users necessitates substantial research efforts across various domains. Data integration involves consolidating information from diverse sources, while knowledge discovery focuses on extracting meaningful movement patterns [10, 67]. Subsequently, knowledge exploitation aims to utilize the derived insights to enhance MSA by providing users with real-time maritime situational updates [10].

Maritime Situational Awareness (MSA) heavily relies on surveillance and tracking systems, notably the Automatic Identification System (AIS), mandated by the Safety of Life at Sea (SOLAS) convention [7, 10, 74]. AIS facilitates automatic data exchange between ships and shore stations every 2–10 seconds [74]. Consequently, a substantial volume of AIS messages compose ship trajectory is continuously generated, encompassing static details such as the Maritime Mobile Service Identity (MMSI), length, and width of ships, which are specified during AIS installation. Additionally, dynamic information, including ship positions by latitude and longitude, timestamp, Course Over Ground (COG), and Speed Over Ground (SOG), is automatically transmitted to track vessel movements. Finally, voyage data provide general information about

the voyage, such as the destination port, estimated time of arrival, and draught, which are manually entered before each journey. As the maritime industry transitions from manual processes to digital solutions, predictive analytics and deep learning models derived from historical AIS data offer ship operators insights into vessel navigation and facilitate data-driven decision-making.

AIS data, despite its tabular format, is complex and requires substantial processing before becoming useful [40]. Notably, crucial information such as voyage start and end flags is not readily available within the data. Research aimed at identifying ships' arrivals and departures indicates that about 62% of AIS destinations are inaccurate and inconsistently updated [95]. Additionally, studies on specific ports have shown reported destination accuracy as low as 4% [61]. The lack of precise vessel destination information poses challenges for port authorities in organizing safe and efficient vessel operations and guiding maritime traffic routes. Given that port efficiency significantly impacts global trade and supply chains, addressing congestion issues becomes imperative, as they are responsible for 93.6% of delays [9, 70].

As maritime traffic increases due to rising commodity demands, automated predictive models can assist port authorities in anticipating challenges such as vessel congestion and facilitating proactive management of port operations. Research has explored various applications for predicting vessel destinations [40, 58, 72, 74, 23, 37, 68, 97, 104]. However, a common challenge acknowledged across these studies is the difficulty in reliably predicting vessel behaviour, attributed to the continuous movement of sea vessels and their susceptibility to environmental factors such as weather, currents, and seasonal variations. Although numerous methods employ knowledge mining and pattern extraction techniques to construct maritime traffic networks, typically represented as abstracted directed graphs through trajectory discretization, challenges persist regarding complexity and alignment with real-world routes [58, 72, 74, 97, 98]. Discretizing trajectories for destination prediction presents several challenges: Simple vertex-edge models often fail to capture the spatial relationships between trajectories, ports, and land masses that significantly influence vessel behavior [22, 46]. Additionally, the model must account for the complex patterns and interdependencies inherent in AIS data points. Another key challenge is extracting

meaningful waypoints[1] (vertices) and route boundaries from massive trajectory data in a robust and adaptive manner, without relying on predefined or fixed route networks [3, 74]. Thus, obtaining high-precision maritime traffic network information is crucial for port destination prediction [72, 74]. This thesis works to resolve the problem of trajectory discretization and construct a high-precision maritime traffic network.

Specifically, this thesis will introduce a novel model for predicting the future routes of vessels as they travel towards their next destination, utilizing Automatic Identification System (AIS) data. A new directed graph representation will be constructed, in which the edges are continuous, smooth spatiotemporal sequences connecting ports. These sequences will then be used to compute similarities with the partial trajectory of a moving vessel, which allows the prediction of the vessel's next destination.

1.1 Problem Statement

In congested ports, port authorities face challenges in efficiently allocating resources for safe and secure cargo loading and unloading. Congestion often results from vessel masters making unscheduled port calls due to weather conditions, medical emergencies, or mechanical issues. These late notifications disrupt port schedules, leading to delays and inefficiencies in resource allocation, such as docking space, cranes, and labor. Access to a tool for monitoring vessel movements, utilizing AIS messages, and predicting their next destination port in advance can enhance the resource allocation process for port authorities. This, in turn, can alleviate congestion and minimize delays within ports. Inspired by related research in destination prediction [40, 58, 72, 74], we propose a two-step model to bridge the gap between historical AIS data and the ability to predict vessels' routes to their next destination. Given a vessel (tanker, cargo ship, or transit ferry), our model takes as input the AIS data of that vessel and outputs a probability distribution of future routes, including the destination. Our destination prediction technique assigns probabilities to multiple potential destinations when a vessel is distant from its endpoint. These predictions evolve as the vessel advances, allowing for continuous updates to the probabilities of

[1] Manoeuvre points at which the vessel changes its course, speed, or velocity that are clustered according to a density threshold.

arrival at each potential port. This probability estimation for all vessels arriving at a port serves as a valuable tool for pre-arrival planning. The probabilities enable the model to convey uncertainty, thereby aiding in human decision-making by indicating when the model's predictions are trustworthy, especially when it assigns a high probability to a single destination.

1.2 Contributions

This research contribution introduces novel AIS data mining techniques aimed at enhancing Maritime Situational Awareness (MSA) by addressing existing challenges such as detecting maritime routes that accurately reflect real-world traffic flow, creating new graph representations of maritime routes, identifying abnormal routes, and predicting destination ports. These advancements play a crucial role in enhancing the accuracy of predicting vessels' future routes to their next destination, thereby ensuring the safety and security of port operations and facilitating informed human decision-making. The findings of this thesis advance the field by providing effective solutions to challenges encountered in maritime data analysis and vessels' destinations' predictions. The research results benefit both the scientific community and practical operators, as discussed below.

1.2.1 Scientific Contributions

The scientific contributions of this research involve the development of novel data mining techniques tailored to address the inherent complexity of AIS data. These techniques have led to the development of a new model for predicting vessels' routes towards their destination ports. The contributions of this thesis are:

1. We propose integrating background knowledge of ports within AIS data to define origin and destination points for trajectory segmentation at the preprocessing stage. *Trajectories* are sequences of ports and the locations between them. In contrast, *trajectory segments* are sequences of locations starting and ending at two consecutive ports, referred to as the origin and destination points, respectively. This segmentation divides AIS data sequences at consecutive ports,

ensuring homogeneity in movement patterns, consistency in local trajectory features, and a new representation that facilitates semi-supervised clustering for generating clustering constraints. We show that these segments enhance interpretability for analysts, align closely with the purpose and context of vessel movements, and provide clear semantic meaning for each segment. By leveraging this combined knowledge, advanced AIS data analysis techniques can capture spatial relationships between vessel movements and ports, thereby improving analysis accuracy and interpretability.

2. We propose a novel semi-supervised clustering method for trajectory segments, which clusters based on the segments' endpoints' similarity. This similarity, defined by the pairs of bearings of the segments' endpoints, captures both the directional and positional relationships between the segments. Our approach addresses the limitations of existing clustering algorithms that rely on geometric assumptions, such as distance or density, which often overlook clusters with non-standard shapes or varying densities. These algorithms also face scalability issues due to the computational expense associated with full similarity matrix calculations. Experimentally, our method outperforms four baseline approaches by automatically identifying the number of clusters, detecting non-linearly separable clusters with irregular shapes and varied densities in linear time, and effectively capturing complex trajectory structures.

3. We propose a novel adaptive thresholding method to filter outlier segments within the generated clusters, as they significantly degrade prediction accuracy. This method offers user assistance in selecting appropriate threshold values to effectively filter out outliers. Our approach demonstrates flexibility and adaptability to the dynamic nature of vessels' motion across diverse sailing scenarios, resulting in cleaner and more accurate representations of historical routes, i.e., cleaner clusters.

4. We propose inferring a novel graph representation of the maritime traffic network from the generated clusters of trajectory segments. This data-driven approach eliminates the need for static route networks that may not accurately

reflect real-world conditions. Instead, the trajectory patterns themselves dictate the topology of the maritime traffic network. Two representations of the maritime traffic network are introduced. The first delineates directed connections between ports, featuring clusters of AIS trajectory segments representing maritime routes along each connection. This enriched network facilitates the prediction of vessel future locations by comparing recent AIS data with similar historical routes. The second representation is a summarized version, where a summarization algorithm aggregates trajectory segments within each cluster between port pairs to reduce computational complexity and simplify comprehension of maritime traffic structure, unraveling complex ship interrelationships.

5. We propose utilizing trajectory similarity measures with the constructed traffic network to predict the destination of a moving vessel. Similarity measures, discrete Fréchet distance (DFD), and dynamic time warping (DTW), are utilized to establish optimal non-linear mappings between two sequences, accommodating timing differences and variations in length. This enables direct matching of AIS data points from recent vessel movements with historical AIS maritime routes. When a vessel is far from its endpoint, probabilities are assigned to multiple potential destinations. These predictions evolve as the vessel progresses, allowing for continuous updates to the probabilities of arrival at each potential port. This probabilistic feature is valuable as it allows the model to express uncertainty.

6. We design extensive experiments to further demonstrate the effectiveness of the proposed methods for extracting high-precision representations of historical maritime routes, thereby increasing the prediction accuracy of destination ports. Our findings demonstrate that our methods efficiently identify maritime routes that accurately reflect real-world traffic flow. These routes enable our prediction model to capture the spatial relationships and navigational constraints characteristic of geographic areas such as the Gulf of Mexico. By leveraging these representations, prediction algorithms are able to uncover hidden patterns and trends in vessel movements, leading to more accurate extrapolation of future routes and their corresponding destination ports.

1.2.2 Practical Implications

The practical implications of this thesis are manifold:

This thesis introduces a model designed to support human decision-making by providing a probability distribution of future routes towards their destinations. This distribution enables the model to express uncertainty, allowing humans to determine when to trust it, particularly when a high probability is assigned to a single route leading to its destination. This enables an analytical risk management approach rather than binary, deterministic decisions. Additionally, the probabilities allow evaluating the risks and benefits of different operational decisions, like scheduling vessel arrivals.

Ships adhere to established maritime traffic routes (i.e., de facto maritime routes) for safe and efficient transportation, with these routes concealed within AIS data. Our model is capable of identifying these routes and monitoring vessels' recent movements by aligning them with these routes, offering significant assistance to port authorities in various ways. Firstly, it optimizes port operations and resource allocation by comprehending primary vessel routes, which can aid berthing schedules and resource deployment. Secondly, it can be used to identify potential chokepoints and high-traffic areas, enabling targeted resource allocation for smooth traffic flow. Thirdly, it can facilitate maritime spatial planning by providing insights into de facto routes, aiding in the designation of official shipping lanes and marine protected areas. Lastly, it can enhance maritime safety and security by monitoring deviations from established traffic patterns to identify potential threats, anomalous behavior, or distress situations that require investigation or intervention by authorities.

1.3 Publications

This thesis is a compilation of five publications that have been submitted, peer-reviewed, and published in the following sources:

1. Lubna Eljabu, Mohammad Etemad and Stan Matwin. (2021). "Destination Port Detection for Vessels: An Analytic Tool for Optimizing Port Authorities Resources". World Academy of Science, Engineering and Technology, Open Science Index 176, International Journal of Civil and Architectural Engineering, 15(8), 398 - 406.

2. Lubna Eljabu, Mohammad Etemad and Stan Matwin, "Anomaly Detection in Maritime Domain based on Spatio-Temporal Analysis of AIS Data Using Graph Neural Networks," 2021 5th International Conference on Vision, Image and Signal Processing (ICVISP), Kuala Lumpur, Malaysia, 2021, pp. 142-147.

3. Lubna Eljabu, Mohammad Etemad and Stan Matwin, (2022). "Spatial Clustering Model of Vessel Trajectory to Extract Sailing Routes Based on AIS Data". World Academy of Science, Engineering and Technology, Open Science Index 190, International Journal of Computer and Systems Engineering, 16(10), 482 - 492.

4. Lubna Eljabu, Mohammad Etemad and Stan Matwin, "Spatial Clustering Method of Historical AIS Data for Maritime Traffic Routes Extraction," 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 2022, pp. 893-902.

5. Lubna Eljabu, Mohammad Etemad and Stan Matwin, "Charting the Course of Ship Track Prediction: A Novel Approach for Maritime Traffic Analysis and Enhanced Situational Awareness," 2023 IEEE International Conference on Big Data (BigData), Sorrento, Italy, 2023, pp. 2588-2597

1.4 Thesis Organization

The thesis structure is organized as follows: Chapter 2 provides a review of related work, exploring existing research and identifying areas for further development. Chapter 3 presents the datasets utilized in our studies, alongside the methodology employed for destination prediction through the monitoring of recent vessel movements. Chapter 4 introduces our preliminary model for predicting destination ports based on recent vessel movements, along with the experiments and evaluations conducted for this model. Chapter 5 introduces Spatial Clustering of Vessel Trajectories (SPTCLUST) as our initial approach to addressing the clustering task of vessels' trajectory segments. This chapter also includes the experiments and evaluation of this algorithm. Chapter 6 introduces enhancements to the SPTCLUST clustering approach to extract high-precision representations of the maritime traffic network, leveraging these

representations for our destination port prediction model, with experiments and evaluations included. Finally, Chapter 7 summarizes our findings and concludes with considerations for future research.

Chapter 2

Related Work

This chapter provides a comprehensive review of research endeavors aimed at enhancing Maritime Situational Awareness (MSA), with a particular focus on leveraging Automatic Identification System (AIS) data for predicting ship destinations. The methodologies employed in mining AIS data to create graph abstraction and predict vessel destination are delineated into six primary categories, which will be thoroughly discussed within this chapter. Furthermore, this review identifies avenues for further exploration and enhancement. Section 2.1 delves into the literature pertaining to the detection of stay points and turning points from AIS data. Section 2.2 focuses on segmenting vessel trajectories. Subsequently, Section 2.3 introduces clustering methods for extracting key points from vessel trajectories to establish a route network, while section 2.4 focuses on outlier detection in trajectories. Section 2.5 presents methods for constructing maritime traffic networks to summarize vessel movements. In addition, Section 2.6 reviews destination prediction approaches that leverage derived knowledge. Lastly, Section 2.7 outlines the research gaps identified within the reviewed literature.

2.1 Integration of Geospatial Background Knowledge with AIS Data

AIS records contain historical navigation routes of vessels, representing real-world maritime traffic. However, analyzing these trajectories solely based on spatio-temporal features, without considering geospatial background knowledge, makes maritime traffic pattern recognition tasks challenging and labor-intensive. Recognizing ports as points of semantic meaning in a trajectory allows a better understanding of vessel trips and detects changes in traffic flow [5, 11, 20, 79, 80]. In trajectory-destination prediction studies, vessels' trajectories are discretized into traversed vertices and/or traversed edges. Vertices represent waypoints, which denote locations along a trajectory where a vessel either remains stationary for a period (known as "stay points")

or makes a turn (referred to as turning points) [55, 98, 109, 106, 103, 111]. The DBSCAN clustering algorithm is commonly used to identify these waypoints. All these studies rely solely on the features of the trajectories themselves for clustering waypoint locations, which makes them susceptible to irregular spatial distributions. Moreover, vessels have different numbers of waypoints for the same route, complicating the identification of waypoint locations based solely on trajectory features. These challenges highlight the fact that making destination predictions in the maritime domain is difficult. To address the challenge of discretizing trajectories and accurately identifying maritime routes concealed within AIS data, a key idea of our work is integrating geospatial knowledge of ports, islands, or marine protected areas (MPAs) with historical AIS data. By doing so, trajectory classification results align with the geospatial shape, providing a more accurate reflection of ship navigation behavior. Leveraging ports' information aids in determining vessel orientation and understanding routes between ports within a specific area, thereby enhancing spatial comprehension and vessel awareness during navigation[1].

2.2 Trajectory Segmentation Utilizing Port Background Knowledge

When examining the entire trajectory as the study's focal point, local abnormal segments or similarities among trajectories may still be disregarded [86, 107]. Moreover, vessels commonly adhere to established maritime routes when navigating from one port to another. Segmenting trajectory data based on specific criteria can aid in identifying meaningful segments along these routes, accurately reflecting real-world traffic flow patterns between ports. The segmentation of ship trajectories is typically approached using various methods, including considering the interval time between trajectory points, the ship's turning angle, and stopping points [10, 15, 18, 54, 59, 79, 93, 101, 109]. Relying solely on Speed Over Ground (SOG) and Course Over Ground (COG) for segmenting ship trajectories can lead to inaccuracies due to technical malfunctions, coverage limitations, and noisy data. Additionally, overlooking continuous and nonlinear movement can make trajectory prediction a challenging task. In order to segment trajectories in a more comprehensive way, our idea is to segment the trajectories based on the integrated background knowledge of ports, which focuses on

[1]Navigation is the destination-oriented movement through space.

discovering vessel movements between two geographical locations (ports' locations) within a geographical space. This segmentation method will enable us to identify established maritime routes concealed within AIS data.

2.3 Clustering Methods to Extract Key Points (Trajectory Discretization)

Due to varying transmission frequencies, AIS data is not well-suited to be used directly as input features for a sequential prediction model [58, 62]. To address this frequency problem, a graph-based method is employed to discretize trajectories into sequences of traversed vertices or, alternatively, sequences of traversed edges [19, 40, 66, 57, 58, 68, 74, 75, 103]. Point clustering methods are utilized to construct these vertices. Vertices typically represent two types of locations: clusters of stop points corresponding to static patterns, and clusters of maneuver points indicating turning points. These vertices are then connected with straight lines to represent edges. However, despite offering a manageable data format, vertex discretization has several drawbacks, including the potential loss of detailed information regarding vessels' complex movement patterns. This method only captures a fraction of the original movement patterns and fails to account for the non-linearity and variability of vessel movements [92]. To address these issues, our approach diverges from clustering key points in AIS trajectories. Instead, we propose clustering the trajectory segments between ports, focusing on the original AIS sequences from one port to the next. Specifically, our idea is to capture continuous, smooth geospatial routes representing complete movement patterns between port pairs. Table 2.1 provides a summary of the clustering algorithms proposed by the authors for stay points vertices and turning points vertices.

2.4 Outlier Detection and Filtering

Employing clustering methods to segment trajectories facilitates the recognition of abnormal movement patterns and enhances data quality. Detecting and filtering outliers in ship trajectory segments serves various purposes, crucial for identifying errors, anomalies, or potential threats. While numerous studies propose methods for

Table 2.1:
A summary of the different approaches proposed by various authors for constructing stay points and turning points vertices.

Approach	Clustering Algorithm
Point based Clustering	Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [19, 68, 103].
	Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [66].
	Balanced iterative reducing and clustering using hierarchies (BIRCH) [72].
	Ordering Points To Identify Cluster Structure (OPTICS) [74].

outlier detection within spatio-temporal series [4, 26, 41, 88], they have not been applied to ship trajectories. Albanese et al. [4] proposed a method for outlier detection in spatiotemporal data, aiming to identify data points significantly deviating from the dataset norm. This method, based on rough set theory, efficiently identifies top outliers but suffers from high time complexity and parameter sensitivity, despite its interpretability in outlier judgment. Duggimpudi et al. [26] introduced Spatio-Temporal Outlier Detection Algorithms based on Computing Behavioral Outlierness Factor. Their proposed algorithms, ST-BDBCAN and Approx-ST-BDBCAN, are aimed at identifying outliers in spatiotemporal data. However, a limitation of these algorithms is their assumption that the data is not highly skewed on the time attribute, which may restrict their applicability in cases of temporal attribute skewness. Moreover, Gupta et al. [41] discuss various studies on outlier detection in trajectory data, including the Trajectory Outlier Detection Algorithm (TRAOD), which partitions trajectories into base units, the smallest meaningful sub-trajectories, and determines the outlier score based on neighboring base units. While these techniques offer interpretability for outlier judgment, they may lack generality for other outlier types. Additionally, relying on multiple distance calculations can be computationally expensive and may not fully capture the non-linear nature of vessel motion trends. Another approach discussed is grid discretization, which simplifies data representation and aids in detecting trajectory outliers. However, it may oversimplify the data and require careful tuning of grid size and shape for effective outlier detection.

The limitations of the reviewed trajectory outlier detectors discussed above motivate us to explore alternative approaches for detecting outliers in ship trajectory segments. Due to the fact that these segments represent maritime routes between two ports, the new outlier detector should account for the varied movement patterns of vessels between different port pairs, and should also be less computationally intensive. Additionally, providing user assistance in selecting threshold values for identifying significant deviations from the norm would enhance the method's effectiveness.

2.5 Maritime Traffic Network Construction

Methods for extracting maritime traffic networks can be classified into three types: vector-based, statistics-based, and grid-based. Table 2.2 summarizes each type's description, including its pros and cons. Our work focuses on extracting comprehensive, continuous, directed maritime traffic networks comprising ports as vertices interconnected by continuous geospatial lines, which are defined by AIS trajectory segments. Such a network is significant for ship anomaly detection, route planning, and navigation safety. Therefore, the compatibility of vector-based methods with detailed edges, has driven the concept of extracting smooth, continuous spatial representations of navigation routes. As a result, these representations can be used to predict ships' behavior and assist port authorities in making better decisions, improving maritime traffic management.

To gain deeper insights into ship navigation data, extracting a maritime traffic network from the vessel's historical trajectory is crucial for effective destination prediction. A graph-based approach is proposed to discretize trajectories, where vertices signify stay points or turning points, and edges represent typical movements between these points. This graph abstraction addresses the issue of varying transmission frequencies in AIS data, facilitating its use as input features for a sequential prediction model. In the literature, various studies have proposed methods to generate maritime traffic networks from AIS data using different algorithms. Some studies suggest utilizing genetic algorithms for this purpose, enabling long-term forecasting and planning of ship routes [24, 23, 37, 64, 65]. However, these methods suffer from

Table 2.2: Categorization of maritime network extraction methods.

Method	Description	Pros	Cons	Article
Vector-based	Create a maritime traffic network by extracting network nodes (waypoints) and edges (navigational legs). This allows traffic state modeling along shipping routes.	Provide a compatible graph-based representation of waypoints and edges connecting them at a larger scale.	Operative in areas with high traffic and less effective in unregulated areas.	[68] [98]
Statistic-based	Analyze ship traffic flow characteristics, such as traffic volume, capacity, speed distribution, and lateral distribution along shipping routes.	Extracting and modeling the ship's navigation route can help understand ship behavior and reveal its regular movement patterns.	The construction of statistical models of original data for large-scale traffic datasets, which requires heavy preliminary work.	[74] [96]
Grid-based	Divide the maritime area into a set of spatial grids with cells defined by the characteristics of passing ships.	Allow reducing the problem scale while improving the efficiency of knowledge storage.	It is unsuitable for areas characterized by complex traffic as it requires prior selection of the optimal cell size.	[48] [90]

high computational complexity and require careful control of several hyperparameters. Alternatively, other methods utilize DBSCAN to identify waypoints[2] in ship trajectories [10, 38, 49, 72, 104, 111]. Notably, DBSCAN-based approaches can complete this process in minutes, significantly faster than genetic algorithms, which can take hours. However, the paths generated by the constructed route network and the actual route trajectory consist of differing numbers of waypoints, making direct comparison challenging. Consequently, many studies resort to manual determination of major waypoints, a process susceptible to subjectivity and error, especially in complex open water environments. As a result, the trajectory discretization methods used to construct route networks still lack connection with real-world scenarios and fail to fully leverage AIS attributes [72, 97]. Moreover, including waypoints complicates network analysis, requiring an understanding of each point's significance and its influence on vessel movements, because waypoints can be traversed by different routes, which adds complexity to interpretation. Additionally, exclusively utilizing longitude and latitude data for calculating route similarities necessitates comparisons of three similarity aspects: vertical distance, parallel distance, and angular distance,

[2]Locations in the ship's trajectory where it changes course or enters or exits a port.

which require considerable computational power and can occasionally lead to inaccuracies. Table 2.3 summarizes the approaches the authors propose for constructing maritime traffic networks.

Table 2.3:
A summary of the different approaches proposed by various authors for constructing maritime traffic networks.

Maritime Traffic Network	Method
Graph-based representation[3]	Dobrkovic [24, 23]: GA to discover waypoints paired with spatial partitioning (QuadTrees).
	Filipiak [37]: Parallel GA for waypoints discovery and a k-d B-tree[4] algorithm for detecting edges between waypoints.
	Arguedas [10]: DBSCAN to identify the waypoints, then Hausdorff distances to define the routes.
	Frorti [38]: DBSCAN and Mahalanobis distance to detect the waypoints that consider the location and velocity features.
	Kontopoulos [49]: DBSCAN to extract waypoints and polynomial interpolation to ensure track locations' continuity, then DBSCAN to cluster trajectories.
	Ren [72]: Multi-clustering algorithm uses the CLIQUE-BIRCH clustering approach for waypoint extraction.
	Zygouras [111]: Partitioning the locations using sliding envelopes along the vessels' course. Then, the locations of the spatially close vessels are grouped together using DBSCAN. Smoothing the previously detected trajectory in the graph, using the B-Spline approach [21].

The aforementioned limitations of trajectory discretization to generate graph abstractions representing maritime traffic networks motivated us to develop an approach that makes use of standardized maritime routes concealed within AIS data to generate a representation of the maritime traffic route network. It contributes toward capturing clusters of maritime routes that reflect real-world traffic flow, by using the complete sequence of AIS data points between two ports on the network, instead of discretely traversed vertices. This representation aids in capturing the spatial relationships and navigational constraints within the studied area. Using the traffic network to represent the entire sequence of AIS data points between ports, we can

[3]Directed graph, whose vertices represent navigational waypoints, while edges represent navigational legs.

[4]Data structure that splits multidimensional spaces like an adaptive k-d tree, but balances the resulting tree like a B-tree.

directly compare the routes of moving vessels (incoming AIS data points) with historical AIS data. In this case, there is no need to record and discretize the ship's trajectory, nor should it cross some specific areas to enable comparisons. Furthermore, the direct comparison of AIS sequences allows for simultaneous comparison of direction, distance, and pattern similarities, which reduces the burden of calculating multiple similarity aspects.

2.6 Ship Destination Prediction

In the literature, the Traffic Route Anomaly Detection (TREAD) method is introduced by Pallotta et al. [68]. TREAD utilizes route clustering techniques for vessel destination prediction within predefined bounding boxes. While effective in constrained regions like the Strait of Gibraltar, TREAD's performance diminishes in areas with expansive regions such as the Indian Ocean due to constraints and density limitations [68]. In contrast, a random forest-based similarity measurement method for global vessel destination prediction without bounding box restrictions is proposed by Zhang et al. [104]. Despite achieving global prediction capability, the model's accuracy is limited to approximately 70 percent, indicating room for improvement in future research. Furthermore, the Hausdorff-distance similarity method for predicting vessel destination based on a multi-featured clustering route network construction method by Ren et al. [72]. Although the concept of utilizing the rich attributes within AIS data to cluster route trajectories, identify waypoints, and construct a maritime route network based on connections between AIS data points has been confirmed, the adjustment of the number of waypoints still requires manual modification.

A neural network-based method utilizing a sequence-to-sequence model with Long Short-Term Memory (LSTM) for predicting vessel destinations is introduced by Nguyen et al. [62]. Their approach involves translating vessel trajectories into sequences of spatial grids within the Mediterranean Sea, utilizing port information given in preprocessed datasets [62]. Inspired by advancements in natural language processing, the study discretized vessel coordinates into spatial grids to forecast arrival ports [62]. However, in a grid-based approach, there's a loss of information due to reduced detail when representing trajectories with grid cells. As the coordinate space expands, more grid cells are required, resulting in decreased data granularity. In contrast, Rong

et al. [74] proposed a hybrid approach using linear regression and Gaussian process regression models to predict ship destinations and trajectories, respectively. While integrating maritime traffic networks into trajectory prediction allows for long-term forecasting, relying on destination prediction findings reduces trajectory prediction accuracy. Furthermore, a framework for predicting vessel trajectories and destinations is proposed by Wang et al. [94]. The framework comprises two parts: the first part outlines a procedure for extracting information from raw AIS data for deep learning, while the second part applies multi-task learning for trajectory and destination prediction [94]. Utilizing deep learning models provides more generalized prediction results across different ocean regions. However, the trajectory prediction outputs only one position.

Predicting a vessel's destination is challenging due to its ability to change routes, speeds, and headings in response to environmental conditions and operational needs, which introduce uncertainties into the vessel's movements. Furthermore, existing methods often depend on port-to-port trajectory formation to classify routes based on historical patterns. By the time a clear trajectory is available, it is frequently too late for effective resource allocation. Waiting for trajectory formation shifts planning from proactive to reactive approach. The need to discretize the predicted trajectory into a sequence of stopping and turning points for comparison with historical route networks necessitates a manual adjustment of the number of traversed vertices. This is because discrepancies between the paths provided by the constructed route network and the actual route trajectory can lead to the loss of important movement pattern information and reduced prediction accuracy. Additionally, the proposed models only generate a single prediction instead of a probability distribution, thus lacking the capability to express uncertainty. While related work in this section achieves promising results, none of it directly addresses the problem we encounter in this thesis—predicting the vessel's next destination by monitoring its recent AIS messages. Therefore, this underscores the necessity for a novel model specifically designed to predict a vessel's next destination, aiding port authorities in proactive planning and resource allocation by leveraging AIS data from recent vessel activities within their local environment.

2.7 Existing Research Gaps

This section provides insights from existing research on knowledge mining and pattern extraction techniques from AIS data, as well as how to use this knowledge to predict future vessel destinations. These insights have motivated the research in this thesis.

Gap 1: Integration of Geographic Background Knowledge with AIS Data. In the existing literature, studies primarily depend on the features of trajectories themselves for extracting comparison features used in clustering and similarity analysis. However, this reliance makes these analysis methods susceptible to uneven spatial densities [94]. To overcome this limitation, we propose a novel method that integrates background knowledge of ports into AIS data to specify origin and destination points. These origin and destination details will motivate the next processing step, segmentation, which will improve the consistency of local trajectory features and create a new representation to facilitate a semi-supervised clustering approach to generate clustering constraints. This is presented in Ch. 4 and Ch. 5 of the thesis and published in [27, 28, 30].

Gap 2: Novel Semisupervised Clustering of Trajectory Segments. In the literature review, ship movement or maritime routes are often depicted by discretizing vessel trajectory into a sequence of traversed vertices. These vertices are defined through point-based clustering methods, which are utilized to cluster specific locations where vessels stop (known as "stop points" or make a turn (referred to as "turning points"). However, discretizing a ship's trajectory impedes the ability to capture the continuous nature of ship movements, which results in a loss of detailed features of the non-linear nature of actual maritime routes. To bridge this gap, we propose clustering trajectory segments (sequences of AIS data points) between port pairs (origin-destination points), so that the AIS segments can be partitioned into K non-overlapping clusters. Based on origin and destination points, must-link constraints is derived to establish a similarity measure between trajectory segments, requiring that segments with identical pairs of bearings at their endpoints be clustered together. This idea facilitates clustering between ports based on the shared directionality of AIS trajectory segments. Thus, this clustering method identifies non-linearly separable clusters with irregular shapes and varied densities in linear time, does not rely on

random initialization, is not sensitive to outliers, and automatically determines the number of clusters. This is presented in Ch. 5 and Ch. 6 and published in [30, 29].

Gap 3: Filtering Outlier Segments. Clustering methods must identify outliers because they can substantially affect the accuracy of data mining outcomes. However, many of the reviewed methods suffer from high computational complexity, parameter sensitivity, and most importantly, they overlook the continuous nature of vessel trajectories, as they were not specifically designed or employed to detect outliers in AIS trajectories [4, 26, 41]. Other methods have been proposed to detect outliers in ship trajectories [88, 100]. Nonetheless, a major limitation of these methods is their reliance on constant threshold values, potentially constraining their ability to assign distinct outlier scores, given the diverse nature of vessel movements between different ports. The aforementioned research gap motivates us to develop an adaptive thresholding approach to enhance the robustness of outlier identification and filtering. Our proposed method involves measuring the pattern similarity of segments within each cluster. The resulting similarity values are then presented as a histogram, offering a visual guide for users to determine an appropriate threshold value for each cluster in order to effectively filter outliers. This is presented in Ch. 6 and published in [29].

Gap 4: Maritime Traffic Network Construction. According to the literature, a maritime traffic network is a predefined or fixed route networks. However, fixed routes assume independent and identically distributed (i.i.d.) data points overlook temporal, spatial, and network dependencies in AIS trajectory data, resulting in inaccurate clustering and pattern detection [104]. Additionally, fixed route networks oversimplify complex patterns and hinder destination prediction accuracy. Relying on fixed routes limits extracting meaningful patterns, de facto maritime routes, vessel paths, and traffic flows from AIS trajectory data [92, 102]. Furthermore, determining the optimal distance in graph abstraction techniques is critical for balancing accuracy in capturing spatial relationships and connectivity patterns against complexity, impacting the fidelity of the abstract graph relative to the original geographical area. To fill this gap, we propose inferring the topology of the maritime traffic network from the generated clusters of trajectory segments. This approach enables the extraction of a flexible, data-driven representation of maritime traffic networks and traffic flows, thereby overcoming the limitations associated with fixed route networks and distance

thresholds. The maritime traffic network can be constructed as a directed graph, where nodes represent ports and edges represent the derived maritime routes connecting them (i.e., clusters of trajectory segments). Inferring the network topology from trajectory patterns enables modeling the intricate dependencies and complex patterns present in maritime traffic while accounting for the non-independent nature of AIS data points. This is presented in Ch. 4, Ch. 5, and Ch. 6 and published in [28, 30, 29].

Gap 5: Vessels' Destination Prediction. Previous research has made significant progress in analyzing navigation history data and predicting ship destinations; however, real-world applicability remains limited. Specifically, there is a need for a model that predicts vessel destinations by monitoring AIS data without relying on fixed route networks of discretized trajectories. Discretizing ship trajectories for destination prediction fails to capture the complex patterns, interdependencies, and non-independent nature of AIS data points within fixed route networks' model [3, 72, 74]. Accurate destination prediction using fixed route networks requires access to the full or a significant portion of the predicted trajectory, ensuring it contains multiple traversed vertices to facilitate port frequency and turn point matching. However, this requirement is not feasible in real-time prediction scenarios where only limited or incomplete trajectory data is available. Furthermore, these models produce a single prediction rather than a probability distribution, limiting their ability to express uncertainty. To overcome these limitations, we propose a tracking method for predicting vessels' destinations. This method involves directly matching our inferred maritime traffic network representation with a partial trajectory of recent AIS data for a moving vessel. The method assigns evolving probabilities to multiple potential destinations as the vessel approaches its next destination, enabling continuous updates to the likelihood of arrival at each port. This methodology is detailed in Chapters 4 and 6, and published in [28, 31].

In summary, this chapter provides a comprehensive review of existing research on knowledge mining and patterns' extraction techniques from Automatic Identification System (AIS) data to predict vessels' destinations utilizing maritime traffic networks. We identified some research gaps from the perspective of predicting vessels' future tracks toward their next destination without trajectory discretization. Building on

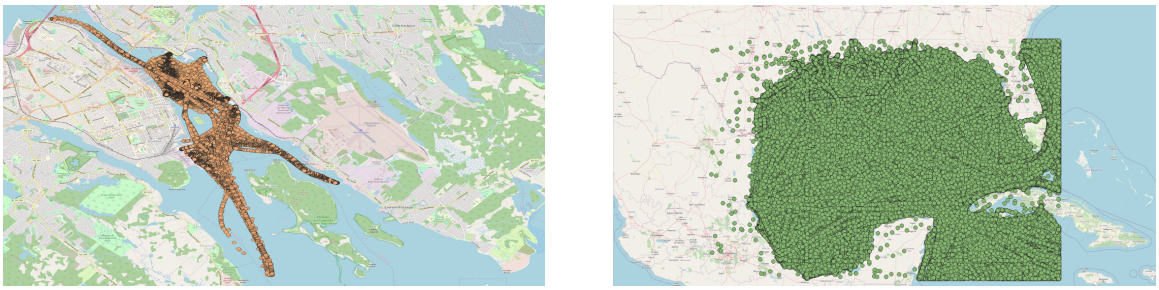
the identified limitations of current research, our research aims to explore novel solutions for maritime knowledge discovery and vessel destination prediction. Chapter 3 presents the datasets utilized in our studies and details the solution approach employed for destination prediction by monitoring recent vessel movements.

Chapter 3

Methodology Overview

3.1 Datasets

AIS (Automatic Identification System) datasets from two distinct maritime regions with varying traffic densities were selected for analysis. Our research team at the Institute for Big Data Analytics collected these datasets. Subsequently, we prepared subsets of these datasets to support the exploration and evaluation of our algorithms for maritime traffic analysis and destination port prediction. The AIS data fields utilized in this thesis are as follows: (i) Maritime Mobile Service Identity (MMSI): A unique 9-digit identifier for each vessel. (ii) Timestamp. (iii) Vessel type. (iv) Geographical position (latitude and longitude). Figure 3.1 depicts the coverage area of the AIS data, showing ship voyages in both Halifax Harbour and the Gulf of Mexico basin. As shown on the map, each point corresponds to a moving vessel at a specific time.



(a) AIS dataset form Halifax. (b) AIS dataset form Gulf of Mexico.

Figure 3.1: An overview of AIS datasets captured from two different maritime areas.

3.2 Solution Approach

The proposed model consists of two main parts. The first part aims to extract maritime routes that accurately reflect real-world traffic flow from AIS data, accomplished through a multi-step data pipeline. The second part focuses on developing a predictive

approach based on trajectory similarity capable of predicting a vessel's destination, given processed AIS sequences comprising *de facto maritime routes*. Figure 3.2 offers an overview of the model.

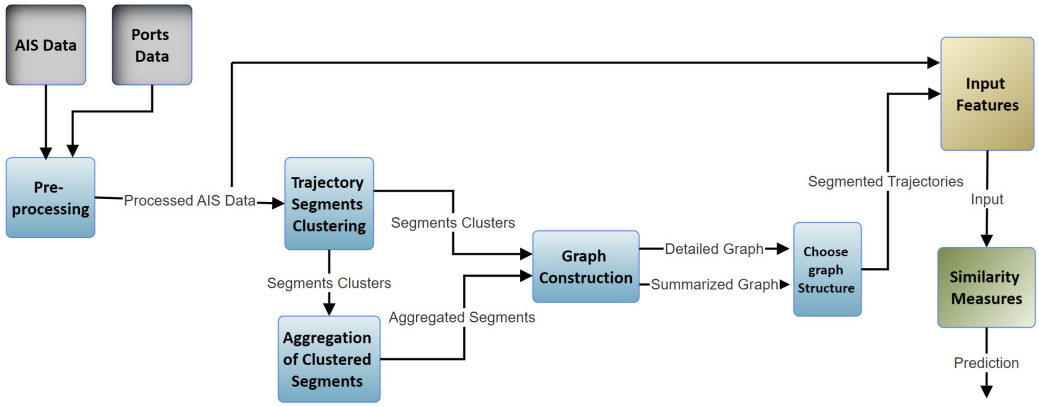


Figure 3.2: The schematic representation of the overall solution approach. The blue boxes depict the first part of the model.

3.2.1 Pre-processing

In the preprocessing step, we first conduct basic data cleaning, eliminating duplicate data points and those outside the study area or on land. Subsequently, AIS data points for each vessel (MMSI) are chronologically ordered. Next, we integrate AIS data with ports' data and segment trajectories based on origin-destination ports. This integration ensures that trajectory categorization and classification align with the geospatial shape, providing an accurate reflection of ship navigation behavior between ports.

3.2.2 Trajectory Segments' Clustering

The next step in the proposed model is to perform trajectory segments' clustering. A trajectory segment is an ordered sequence of AIS data points that originate at one port and end at the next port. The purpose of trajectory segments' clustering is to identify clusters of *maritime routes* that reflect the real-world traffic flow between ports. The clusters are then used for the construction of the graph representation outlined in Section 3.2.4. Given the AIS data, trajectory segments are identified by using the integrated port points during the first preprocessing step. As a result, port

points make it possible to deduce must-link constraints for establishing similarity between trajectory segments, ensuring that segments following the same direction between origin and destination points are clustered together. We developed a semi-supervised clustering approach, named SPTCLUST, to cluster trajectory segments based on segments' endpoints' similarity, which is defined as the pairs of bearings of the segments' endpoints. With this approach, clustering time complexity can be reduced significantly while maintaining the nuanced dynamic features of AIS data points. This method is presented in Ch. 5 and Ch. 6 and published in [30, 29]. The unique and stochastic characteristics of maritime traffic pose challenges for developing effective traffic clustering models using traditional techniques, as they often rely on assumptions about cluster shape or density that may not hold true in areas with diverse traffic densities and complex ship interrelationships. Therefore, the proposed SPTCLUST offers realistic solutions by identifying non-linearly separable clusters with irregular shapes and varied densities in linear time. It does not depend on random initialization, is not sensitive to outliers, and automatically determines the number of clusters.

3.2.2.1 Baseline Clustering

We utilize the following clustering baselines due to their popularity and effectiveness in ship trajectory data clustering:

Kmedoids is a partition-based clustering method. It aims to partition data into k clusters by minimizing the sum of dissimilarities between trajectory segments and a representative segment within each cluster (called a medoid) [47]. It requires ***one parameter: (k)***, which defines the number of clusters.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm [33]. It groups together closely packed trajectory segments based on ***two parameters: epsilon (ϵ)*** defines the radius within which neighboring segments are considered part of the same cluster, and ***minPts*** specifies the minimum number of segments required to form a dense region (core point).

OPTICS (Ordering Points To Identify the Clustering Structure) is an extension of DBSCAN that produces hierarchical clustering [8]. It computes the reachability distance for each segment, representing its proximity to the nearest core segment,

enabling the detection of clusters with varying densities and sizes. It requires ***three parameters: min_samples*** specifies the minimum number of segments in a neighborhood, ***min_cluster_size*** specifies the minimum reachability distance for clustering, and ***xi***, reachability distance cutoff; that establishes the relative decrease in density.

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) is a hierarchical clustering algorithm designed for large datasets [105]. It incrementally and dynamically builds a tree-based data structure to represent the clustering hierarchy. There are ***three parameters*** in this algorithm, which needs to be tuned. ***branching_factor*** specifies the maximum number of sub-clusters in each node, ***n_clusters*** is the number of clusters to be returned after the entire BIRCH algorithm is complete, and ***threshold*** is the maximum number of segments a sub-cluster in the leaf node can hold.

The directed Hausdorff distance [85] is used to compute the input distance matrix for baselines, but due to its time-consuming nature, the Douglas-Peucker (DP) algorithm [25] is employed to compress AIS data for efficiency.

3.2.3 Aggregation of Clustered Trajectory Segments

The subsequent step is to perform the aggregation of clustered trajectory segments. The purpose of aggregation is to generate continuous, smooth, representative segments from port to port, commonly referred to as reference routes. These representative segments are used to construct a graph representation that summarizes the trajectory data, outlined in Section 3.2.4. The representative segments are constructed by using linear interpolation and the arithmetic mean of the trajectory segments within each cluster. The rationale for this aggregation is twofold: firstly, to utilize linear interpolation for bridging the gaps between AIS data points, and secondly, to employ arithmetic mean to simplify complex information, providing insights into the central tendency of the segments by computing their average spatiotemporal features. This combination ensures smoother, more continuous representative segments. We developed an algorithm to generate Reference Routes of Trajectory (RROT algorithm), presented in Ch. 4, Ch. 5, and Ch. 6 and published in [28, 29]. Additionally, this

approach aids in determining cluster quality; for instance, clusters containing segments traveling in opposite directions yield incomplete reference routes. Therefore, considering the port connectivity of the resultant aggregated segments further helps cluster quality assessment. Additionally, utilizing a traffic network with summarized routes (network connections) enhances prediction computation complexity.

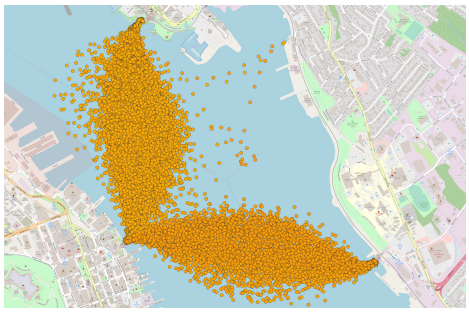
3.2.4 Graph Construction

At this stage, the generated clusters are labeled based on the vessels' type they belong to, while maintaining all trajectory features. These clusters form a detailed graph representation with detailed edges, where these edges represent the de facto maritime routes between two ports, which can be used as input for the prediction method, outlined in section 3.2.6. The detailed graph is presented in Figure 3.3b, where the circles indicate ports' area. Similarly, the aggregated trajectory segments also retain vessels' type labels. Illustrated in Figure 3.3c, these aggregated segments provide a summarized trajectory, representing shipping lanes between port pairs. Different colors indicate different directions. Additionally, these aggregated segments can be utilized as input for the prediction method outlined in Section 3.2.6.

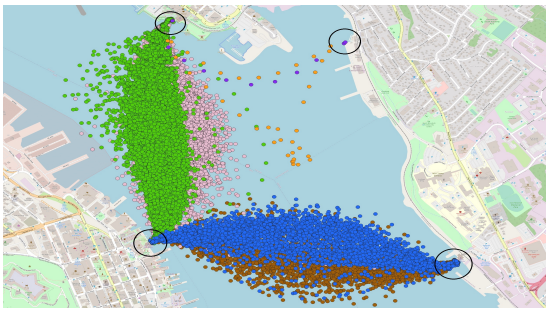
The proposed approach makes use of AIS data to identify de facto maritime routes maritime routes concealed within AIS data, aiming to create a high-precision representation of the traffic route network. It contributes toward capturing the traffic clusters with complete spatiotemporal sequences by using the complete sequence of AIS data points on the network instead of the traditional discrete traversed vertices. This approach ensures adaptability to traffic scenarios in diverse geographical waters while exploiting the wealth of information contained within AIS data.

3.2.5 Selection of Graph Structure for Input Features

The constructed graph representations discussed in Section 3.2.4 can be utilized for direct matching with recent trajectory data of a moving vessel of the same type to predict its destination. By utilizing the spatial features to compare the input sequences, the direction, distance, and pattern similarities can be captured simultaneously. The input sequences from graph representations for the prediction approach can be aggregated segments or clusters of segments. The aggregated segments can provide a



(a) Segmented Trajectory.



(b) Trajectory Segments Clusters.

(c) Aggregated Trajectory Segments.

Figure 3.3: An example of a ferry trajectory from Halifax Port preprocessed and segmented is shown in (a). In (b), a detailed directed graph consists of trajectory segments' clusters. In (c), a summarized graph consists of aggregated segments.

general idea of the vessel's route and destination, while clusters of detailed segments can offer a more precise prediction by considering specific routes and patterns. This approach allows for a flexible and accurate prediction method that can be adapted to different scenarios.

3.2.6 Similarity-Based Prediction

We propose a similarity-based prediction method to predict the destination of a moving vessel, as shown in Figure 3.4. Trajectory similarity is commonly evaluated by measuring the distance between their respective points. Distance is a way to measure the similarity of trajectory segments' patterns and proximity. The distance between two trajectories, denoted as $d(P, Q)$, reflects their dissimilarity, with higher values indicating lower similarity between them. These similarity measures are designed to accommodate varying trajectory lengths, considering that vessel trajectories differ in both distance traveled and recorded data points. Furthermore, they are capable of handling the non-linear nature of vessel movement.

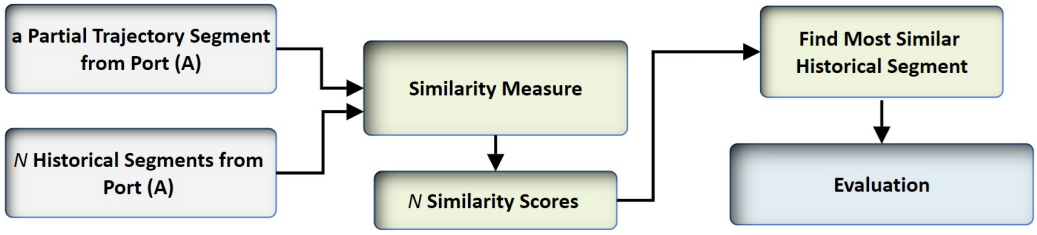


Figure 3.4: A schematic representation of the prediction stage, which constitutes the second part of the overall solution approach.

In Figure 3.4, the prediction process involves measuring similarities between the ongoing trajectory segment and historical trajectory segments originating from the same port. The selected similarity measure yields similarity scores, indicating the resemblance between the ongoing segment and each historical trajectory segment. Subsequently, for evaluation purposes, the destination of the historical trajectory segment with the highest similarity score to the ongoing segment is predicted as the vessel's destination. This prediction approach is presented in Ch. 4 and Ch. 6 and published in [28, 31]. This method facilitates direct comparisons between ongoing routes and historical routes, without the need for the trajectory of the moving vessel to be available in advance. As the vessel departs its port, the method assigns probabilities to multiple segments originating from that port. These probabilities are updated as the vessel progresses, enabling continuous adjustments based on the emerging path pattern, and thereby providing continuous updates to the likelihood of arrival at each port. Five similarity measures are explored: (1) Discrete Fréchet Distance (DFD)[84], (2) Dynamic Time Warping (DTW)[43], (3) Partial Curve Mapping (PCM)[44], (4) Area between two curves (Area)[13], and (5) Curve length (CL) [13]. These measures are selected for their parameter-free nature. DFD and DTW are widely used in trajectory similarity and have demonstrated effectiveness.

3.2.6.1 Discrete Fréchet Distance

The discrete Fréchet distance (DFD) is a widely used measure of similarity between trajectories because it preserves the location and sequence of points along the trajectories [14]. When calculating the discrete Fréchet distance, it takes the smallest maximum distance between the aligned points. A dynamic programming solution is

proposed for finding the discrete Fréchet distance between two trajectories [84]. The discrete Fréchet distance, also known as the coupling measure. The DFD, seeks the path in the distance matrix with the lowest minimum cost, with each cell (i, j) representing the cost of a pair of points from P and Q as measured by $d(p_i, q_i)$. Then, the DFD takes the maximum distance between points along a path. The discrete Fréchet distance $DFD(P, Q)$, is defined as follows:

$$DFD(P,Q) = \begin{cases} 0 & \text{if } |P| = |Q| = 0 \\ \infty & \text{if } |P| = 0 \text{ or } |Q| = 0 \\ \max\{d(Head(P), Head(Q)), & \\ \quad \min\{DFD(P, Tail(Q)), & \\ \quad\quad DFD(Tail(P), Q), & \\ \quad\quad DFD(Tail(P), Tail(Q))\}\} & \text{otherwise} \end{cases} \tag{3.1}$$

Where: $|P|$ represents the number of elements in sequence P. $d(Head(P), Head(Q))$ is the Euclidean distance between the first elements of sequences of P and Q. $Head(P)$ and $Head(Q)$ denote the first elements of sequences of P and Q respectively. $Tail(P)$ and $Tail(Q)$ represent sequences P and Q excluding their first elements. The function min returns the minimum value among its arguments. The function max returns the maximum value among its arguments.

The discrete Fréchet distance (DFD) operates with a fixed quadratic runtime of $O(nm)$. It returns a value of zero when P equals Q and increases positively as the trajectories become more dissimilar, and DFD is parameter-free [87].

3.2.6.2 Dynamic Time Warping

Dynamic Time Warping (DTW) is one of the most commonly used techniques to measure the similarity between two trajectories. DTW was originally designed to compare different speech patterns in automatic speech recognition [43]. The main objective of this method is to find the optimal alignment between two trajectories by finding a path between their points that minimizes the cumulative distance between them [13, 42].

The DTW algorithm searches through all point combinations between two trajectories using dynamic programming. Thus, DTW is an inclusive measure because it can be used with trajectories of different lengths [42]. Given two trajectories, P and Q of length n and m, respectively, DTW aligns these sequences by creating an n-by-m distance matrix in which the (i, j) element equals $|p_i - q_j|$. An alignment between two sequences is represented by a wrapping path $w = (w_1, w_2, ..., w_k)$ in the matrix, which has to be monotonic, contiguous, start from the bottom-left corner, and end at the top-right corner of the matrix. The optimal alignment of sequences is to arrange all sequence points by minimizing the distance. Hence, the DTW algorithm finds the alignment path, which runs through the low-cost cells in the distance matrix. The DTW distance is defined as:

$$DTW(P,Q) = \begin{cases} 0 & \text{if} |P| = |Q| = 0 \\ \infty & \text{if} |P| = 0 \text{or} |Q| = 0 \\ d(Head(P), Head(Q)) + & \\ \quad \min \begin{cases} DTW(P, Tail(Q)), \\ DTW(Tail(P), Q), \\ DTW(Tail(P), Tail(Q)) \end{cases} & \text{otherwise} \end{cases} \tag{3.2}$$

Where: $|P|$ represents the number of elements in sequence P. $d(Head(P), Head(Q))$ is the distance between the first elements of sequences P and Q. $Head(P)$ and $Head(Q)$ denote the first elements of sequences P and Q respectively. $Tail(P)$ and $Tail(Q)$ represent sequences P and Q excluding their first elements. The function min returns the minimum value among its arguments. The computational complexity of DTW is $O(nm)$. DTW is parameter-free, meaning it does not require any additional parameters to be specified. It is an unbounded measure, where identical trajectories result in a value of 0, while larger DTW values indicate greater dissimilarity between the sequences.

3.2.6.3 Partial Curve Mapping

Partial Curve Mapping (PCM) is a method used to measure the similarity between a given curve and segments of a larger curve. This technique was developed by Witowski et al. [44], to address the challenge of comparing curves of different lengths.

The PCM method to assess the similarity of vessel trajectories is employed by Jekel et al. [13]. The PCM algorithm operates by combining the arc lengths and areas between two curves of varying lengths. Initially, the arc length of the shorter curve is projected onto a section of the longer curve. Trapezoids are then constructed between the curves, and the areas of these trapezoids are summed. Subsequently, an offset is defined to "slide" the shorter curve along the longer one. This process is repeated iteratively, with the shorter arc length being imposed on different sections of the longer curve, until the last data point of each curve is considered. The PCM value is determined as the minimum area obtained from all attempted arc length offsets. PCM is a quadratic runtime complexity of $O(nm)$, parameter-free operation.

3.2.6.4 Area between two curves

A similarity measure for comparing trajectories was proposed, emphasizing the significance of equalizing the number of data points between curves to construct quadrilaterals for area approximation. [13]. To achieve uniformity, additional points are introduced to curves with fewer data points rather than removing points to preserve all available information. This augmentation involves bisecting existing points, prioritizing the largest Euclidean distance between consecutive points until both curves have an equal number of points. The assumption of straight lines between points is fundamental in polygon construction, as it ensures that adding points via linear interpolation does not alter the area between curves. Additional points serve solely to facilitate area approximation. As the number of data points increases on both curves, the accuracy of area estimation improves correspondingly.

A visual representation of the Area method is provided in Figure 3.5, illustrating quadrilateral construction between two curves (P and Q). While Q comprises four data points, P includes five. To reconcile this disparity, an artificial data point is introduced to Q data by bisecting consecutive points with the greatest Euclidean distance, thereby aligning the number of points in both sequences.

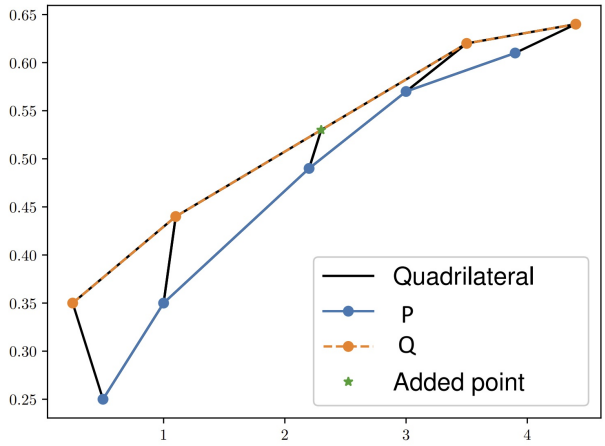


Figure 3.5: The area between two curves is approximated by summing quadrilaterals. Source: Elaborate by the author.

Quadrilaterals are then formed between corresponding pairs of points from each curve, with Gauss's area formula applied to compute the area of each quadrilateral. The summed areas yield the effective area between the curves.

$$Area = \frac{1}{2}|x_1y_2 + x_2y_3 + x_3y_4 + x_4y_1 - x_2y_1 - x_3y_2 - x_4y_3 - x_1y_4| \tag{3.3}$$

where (x_i, y_i) represents the vertices of the quadrilateral, the area between two trajectories is a positive value ($Area \geq 0$). All quadrilateral areas are summed to give an effective area between two trajectories. The area between two curves measure is robust to noisy trajectory data; it is a parameter-free measure.

3.2.6.5 Curve Length

The curve length method, developed by Jekel et al.[13], was inspired by the work of Andrade-Campos et al.[1]. New criteria for the determination of material model parameters were proposed, along with a novel curve length attribute to be included in the objective function to quantify the quality of fit between two curves [1]. The principle of the curve length method is to compare a point on one trajectory to its corresponding arc length location on the other trajectory. The authors stated that the data point values can be expressed as a function of the trajectory length distance from the first data point. A corresponding data point on the trajectory Q is calculated at

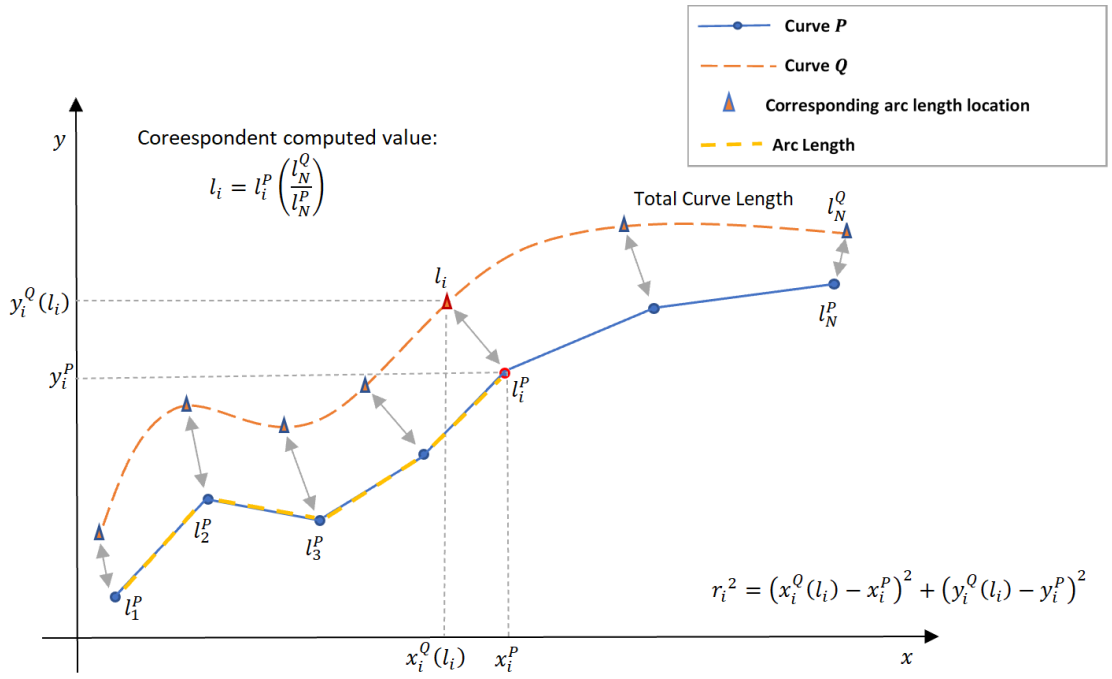


Figure 3.6: Illustration of the Curve Length method between two curves. Source: Elaborate by the authors.

the equivalent arc length location of the trajectory P. Then, squared residual values are calculated as a function of both the dependent and independent variables:

$$r_i^2 = (x_i^Q(l_i) - x_i^P)^2 + (y_i^Q(l_i) - y_i^P)^2, \quad where\, l_i = l_i^P \left(\frac{l_N^Q}{l_N^P}\right) \tag{3.4}$$

The sum of these squared residuals is used to quantify the difference between the two trajectories, as shown in Figure 3.6. The curve length method measure has a quadratic O(nm) time and is a parameter-free measure.

In summary, this chapter has comprehensively outlined the data sources, preprocessing techniques, and methodological frameworks utilized in this study. This provides a comprehensive foundation for the analysis and modeling of maritime traffic patterns and vessel destination prediction utilizing Automatic Identification System (AIS) data. Chapter 4 defines the first phase of our research. In this phase, we study the prediction of vessels' next destinations, employing AIS data of transit ferry and cargo vessels captured from Halifax Harbour. This includes data preparation processes, the construction of continuous smooth traffic networks to summarize vessel

trajectory data while preserving the fundamental shape of the trajectory, thereby capturing essential spatial relationships and connectivity patterns, and the exploration of five similarity measures for destination prediction tasks. The primary focus of this chapter is to explore the initial design of a prediction model that assigns evolving probabilities to multiple potential destinations, allowing continuous updates to the likelihood of arrival at each port.

Chapter 4

Destination Port Prediction Using a Novel Graph Representation of Maritime Traffic

The current approaches to ship destination prediction rely on graph abstraction techniques, which involve discretizing trajectories into sequences of traversed vertices and edges. The objective of graph abstraction is to capture essential spatial relationships and connectivity patterns while simplifying the representation. However, discretizing a vessel's trajectory into a sequence of traversed waypoints[1] poses challenges in accurately modeling vessels' movements, especially in high-ship density areas where overlapping trajectories complicate waypoints' identification. Additionally, graph abstraction techniques offer predefined or static route networks, where selecting an optimal distance threshold is crucial to maintaining a balance between accurately capturing spatial relationships and connectivity patterns while minimizing complexity. The distance threshold significantly influences how faithfully the abstract graph represents these relationships compared to the original geographical area. This chapter[2] presents a methodology for extracting shipping patterns from semantic trajectories by enriching AIS data with contextual information to facilitate trajectory segmentation and annotation for advanced pattern recognition. An algorithm is introduced to aggregate analogous trajectory segments based on origin-destination points, aiming to uncover higher-level maritime patterns and routes while reducing data complexity, without relying on distance thresholds. The resulting summarized trajectory representation is expected to improve the predictive accuracy of vessel destination forecasting.

[1] Manoeuvre points at which the vessel changes its course, speed, or velocity that are clustered according to a density threshold.

[2] This chapter is based on the publications [28, 27]

4.1 Introduction

Maritime transportation plays a crucial role in economic growth as the world's population expands. Accurate vessel destination forecasting can substantially enhance decision-making efficacy within the industry and contribute to ensuring a secure and efficient maritime traffic environment. Despite the availability of real-time vessel data provided by the Automatic Identification System (AIS), inaccuracies in manually entered fields like destination, along with noisy and complex data, present challenges [50]. Port authorities face difficulties in resource allocation when utilizing AIS data, as a significant portion of AIS messages either lack destination information or contain inaccuracies in reported destinations [40]. The lack of accurate information about vessels' destinations would subject port authorities to challenges like arranging port activities for safe and efficient vessel operations and guiding traffic routes to ensure the safety and efficiency of the maritime traffic environment. Hence, the research on predicting vessels' destinations holds significant value for port authorities seeking to automate decision-making processes and ensure the efficient allocation of resources for maintaining a safe and secure maritime traffic environment.

Detecting the destination port can be seen from the trajectory path that is normally traversed using AIS historical data, which is then compared to current trajectories to predict the destination. Thus, the similarities between traveling and historical trajectories can be measured and utilized to classify and predict the vessel's destination [28]. Meaningful shipping patterns can be extracted from semantic trajectories, see Appendix A.2, by integrating geographical domain information into AIS data. This enhances discrimination and allows for more complex analysis of vessel routes and changes in movement behavior [6, 60, 5, 12, 79]. Trajectory segmentation methods provide the basics for detecting changes in vessel movement behaviour [35]. By segmenting trajectories based on spatial context, specifically, port areas, the spatial dependencies can be better localized within each segment, as vessels exhibit more homogeneous movement patterns in specific regions. This approach facilitates the discovery of maritime routes and traffic flows. Similarity analysis is crucial for solving movement pattern recognition challenges like classification, clustering, and anomaly detection. Vries et al. [20] introduced a similarity measure using edit distance, applied in vessel type prediction. Alizadeh et al. [5] proposed a point-based model for vessel

location and traffic forecasts, utilizing similarity analysis of historical AIS data. Zhen et al. [108] presented an anomaly detection method for vessel behavior, devising a similarity measure based on spatial and directional features in trajectory data. This measure was then employed in clustering and classification tasks to identify abnormal vessel sailing behavior.

In this study, we propose integrating AIS data with geographical information to identify port stops. Subsequently, we segment trajectories based on these ports to facilitate trip understanding and track vessel activities in large water areas. Following trajectory segmentation, semantic labels are added to each segment, such as the path number to distinguish the segments with the same start and end ports, and the segment identifier, a unique number for each segment. The labeled segments are then incorporated into reference route construction to create a summarized reference trajectory by aggregating segments with the same path number. Then, we explore five similarity measures to determine which is most effective for comparing vessel routes. These measures are subsequently employed as classification techniques to predict destination ports and assess similarity between short segments and reference routes. Figure 4.1 presents our proposed approach for destination port prediction. This framework has four main steps: 1- data preparation, 2- reference route construction, 3- similarity measurements, and 4- destination port prediction.

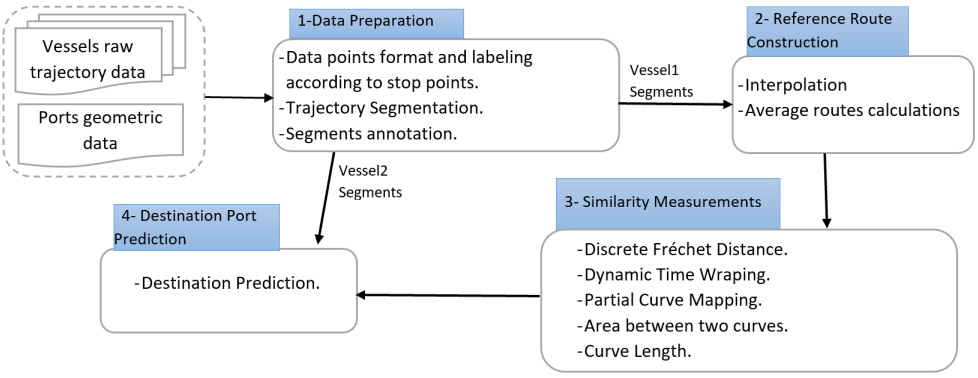


Figure 4.1: Framework of vessel destination port prediction.

The contributions of this chapter are as follows: (i) We propose a geographical knowledge annotation for the generated segments to distinguish them by their spatial features and context. (ii) A novel method to generate Reference Routes of Trajectory (RRoT) is proposed that aggregates a set of segments with the same path number

into a summarized trajectory. (iii) We compare five different similarity measures to understand the best similarity measure for comparing vessel trajectories. (iv) We present a method to predict the destination of a vessel by processing a short and recent segment. Our destination prediction technique assigns probabilities to multiple potential destinations when a vessel is distant from its endpoint. These predictions evolve as the vessel progresses, facilitating continuous updates to the arrival probabilities at each potential port. This probability estimation is valuable as it enables the model to express uncertainty, thereby allowing humans to determine when to trust the model, particularly when it assigns a high probability to a single route or destination.

4.2 Definitions

Definition 4.2.1 (Trajectory Point)**.** A trajectory point l_i, is a geolocation of moving vessel o at time i, and is defined as, $l_i^o = \langle x_i^o, y_i^o \rangle$, where x_i^o represents the longitude of the location, which varies from 0° to ±180°, and y_i^o represents the latitude of the location that varies from 0° to ±90°.

Definition 4.2.2 (Trajectory)**.** A trajectory τ, is a time-ordered sequence of trajectory points of a moving vessel o, $\tau^o = \langle l_0^o, l_1^o, .., l_n^o \rangle$. Because the trajectory data is from AIS, the trajectory points could provide additional information, including vessel identity, course/speed over ground, and ship type, which are considered trajectory features.

Definition 4.2.3 (Trajectory Segment)**.** A trajectory segment s_i is a set of consecutive trajectory points belonging to a trajectory τ^o divided based on *partitioning positions*, where $s^o = \langle l_j^o, \cdots, l_k^o \rangle$, $j \geq 0$, $k \leq n$ and s^o is a subsequence of τ^o. The process of generating segments from a trajectory is called trajectory segmentation.

Definition 4.2.4 (Segment Label)**.** A segment label is an identifier given to a segment to distinguish shipping lanes with the same origin and destination ports. These labels facilitate the discovery of maritime routes and the identification of traffic flow between ports.

Definition 4.2.5 (Port)**.** A port is a circular area of radius r centered on the geographical coordinates of a sea port [16].

Definition 4.2.6 (Origin and Destination Points)**.** The center location of the nearest port to the first point in a segment is designated as the *origin point*, O, of a segment, while the center location of the nearest port to the last point of a segment is designated as the *destination Point*, D of a segment.

Definition 4.2.7 (A reference route)**.** A reference route between origin point A and destination point B can be defined as a resampled trajectory segment denoted as R_{AB}, representing the average behavior of all trajectory segments starting from origin point A and ending at destination point B. The calculation of the reference route is explained in Algorithm 1.

Definition 4.2.8 (Trajectory Similarity)**.** *Trajectory Similarity* is a distance generalization to quantify the degree of resemblance between two trajectories. Specifically, in this work, $dist(\tau_1, \tau_2)$ represents the distance between a full trajectory segment τ_1 and a portion of trajectory segment τ_2. The greater the value, the less similarity between the two patterns.

Definition 4.2.9 (Linear interpolation)**.** Linear interpolation involves reconstructing a continuous trajectory or trajectory segment from AIS data by uniformly sampling spatiotemporal positions along its length and estimating values between two known points. Specifically, linear interpolation of a vessel trajectory estimates the vessel's position at a given time by assuming linear motion between two known positions. Given two data points (t_1, l_1) and (t_2, l_2): $l(t) = l_1 + \frac{(t-t_1)}{(t_2-t_1)} \cdot (l_2 - l_1)$, where $(t_1 \leq t \leq t_2)$. For positions $(l_1 = (x_1, y_1))$ and $(l_2 = (x_2, y_2))$: $(x(t) = x_1 + \frac{(t-t_1)}{(t_2-t_1)} \cdot (x_2 - x_1)$, $y(t) = y_1 + \frac{(t-t_1)}{(t_2-t_1)} \cdot (y_2 - y_1))$.

4.3 AIS Data and Data Preprocessing

Our analysis involves data on the trajectory of vessels captured in the Halifax harbor area in Nova Scotia, Canada. Section 4.3.1 describes and visualizes the data we use in this study. Section 4.3.2 details the AIS data preprocessing to maximize the utility of the data in our approach.

First, we do an experiment on the distribution of destination ports in AIS message 5 in order to see how much the destination data embedded in this field is useful. So,

it is practical for port authorities to be able to use this data efficiently. AIS message 5 includes the vessel's destination port information, with a free-text field allowing up to 20 characters. However, these manually filled fields are often missing or incorrect. A distribution of the destination field of message 5 in the AIS data is shown in Figure 4.2. The majority of the distribution shows that the destination port is not entered or the destination port field is unknown, where the name of a small town, bay, anchorage, or shipyard is entered. All these variations cause ambiguity in destination reports and lead to confusion and data interchange inefficiency.

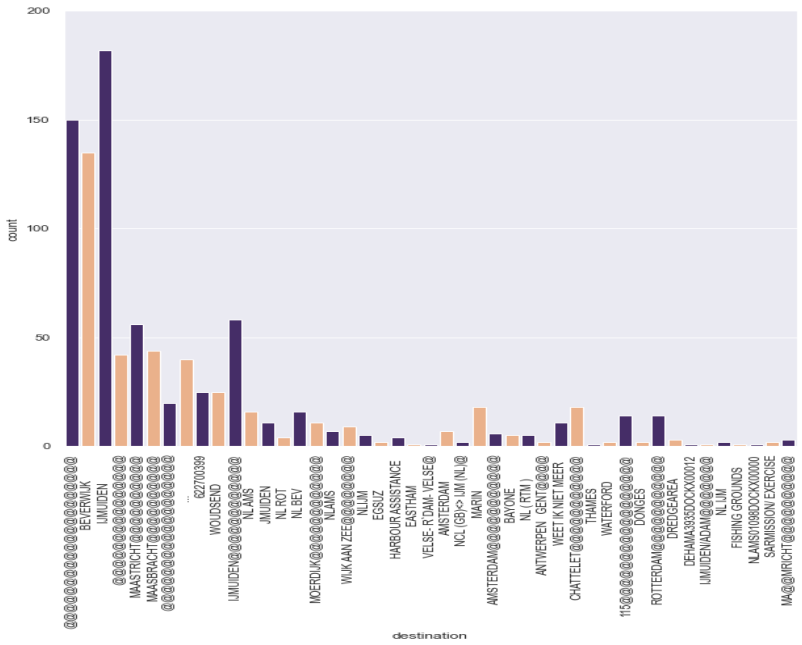


Figure 4.2: The distribution of destination ports in AIS message_5 shows most of the data submitted in this field is not accurate and not a representation of the real world.

4.3.1 AIS Data Overview

The AIS data used in this study is generated from AIS messages captured in the Halifax harbor area, which were collected by our team at the Institute for Big Data Analytics [36]. For our experiments, we explore:

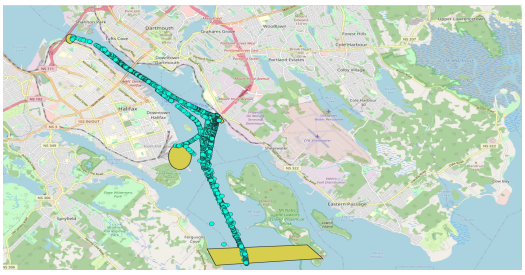
- Two datasets of the trajectories of two transit ferries navigating the Halifax port area. The first vessel, V_1, trajectory data was collected from March 18, 2019 to July 10, 2019. The dataset has 27,028 data records. The second vessel, V_2, has 103,161 data records collected from March 5, 2019 to July 12, 2019.

(a) Trajectory data of V_1.

(b) Trajectory data of V_2.

Figure 4.3: Overview of AIS data of two transit ferries, V_1, V_2, from Halifax port.

- Two datasets of the trajectories of cargo vessels navigating in the Halifax port area. The dataset vessel, D_1, trajectory data were collected from March 7, 2019 to July 9, 2019. The dataset has 38,852 data records. The dataset, D_2, contains trajectory data collected from March 16 to July 11, 2019, including 8,070 data records.



(a) Trajectory data of D_1.

(b) Trajectory data of D_2.

Figure 4.4: Overview of AIS data of two cargo vessels, D_1, D_2, from Halifax port.

4.3.2 AIS Data Preprocessing

The first step of data preparation is to form the vessel *trajectory* by sorting a set of *trajectory points* based on their date and time features. Then, we annotate the trajectory points using the semantic layer of ports' information (Figure 4.5). Annotating the trajectory points provides a more detailed description of the vessel's sailing behaviour.

The feature space of the AIS data is annotated according to the *port* area in the semantic layer; each *trajectory point* is checked to determine whether or not it is positioned within one of the ports' areas. If the trajectory point is located within a port area, it is annotated as a *stop*. If it is located outside of a port area, it is

port_name	geometry
port1	POLYGON ((-63.569431 44.649993, -63.56943966749199 44.64981656914741, -63.56946558649528 44.64964183742038, -63.5695085073956
port2	POLYGON ((-63.568048 44.663875, -63.56805666749199 44.66369856914741, -63.56808258649527 44.66352383742037, -63.5681255073956
port3	POLYGON ((-63.555828 44.662453, -63.55583666749199 44.66227656914741, -63.55586258649527 44.66210183742037, -63.5559055073956
port4	POLYGON ((-63.547843 44.648763, -63.54785166749199 44.64858656914741, -63.54787758649528 44.64841183742038, -63.5479205073956

Figure 4.5: A depiction of the ferry four terminals *(ports)* data.

annotated as a *move*. We have a total of eight ports: four terminals for transit ferries, three piers for cargo, and one polygon marking the entrance and exit points of the harbor.

Then, the annotated data is used to partition the trajectory into *segments*. This process is called trajectory segmentation. Trajectory segmentation facilitates an understanding of the trip's purpose and enables tracking vessels across a large area to identify their activities. The segmentation process detects origin and destination points in the trajectory and uses these positions to divide it into distinct segments. These segments accurately capture the movement between port pairs, which helps mine richer knowledge. Then, the centers of the nearest ports are added to the segment, representing the segment's start and endpoints. Each of these segments represents movement from the origin port to the destination port; see Figure 4.6.

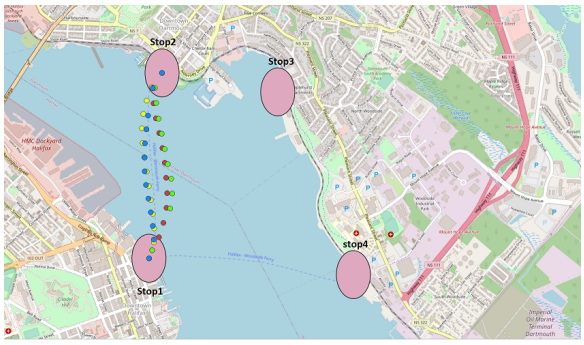


Figure 4.6: Segments between $stop_1$ and $stop_2$. The red and green segments represent the movement from $stop_1$ to $stop_2$. The blue and yellow segments represent the movement from $stop_2$ to $stop_1$.

To facilitate the interpretation of the sailing behavior of the segments, we propose to assign another feature to segments, *route*, and annotate segments with the same origin and destination points with the *Path* number to facilitate classifications. Figure 4.6 shows four segments representing trips through the area of the same shipping lane (e.g., $stop_1$-$stop_2$), but the segments are separated by *routes* with different start and endpoints (e.g., the red and green segments represent the movement from $stop_1$

to $stop_2$, and they are labeled as $Path_1$. The blue and yellow segments show the movement from $stop_2$ to $stop_1$ and are annotated as $Path_2$. To sum up, these segments may appear visually similar, but the direction is different. Table 4.1 represents the number of segments generated for each vessel trajectory.

Table 4.1: Vessels Trajectory Data Statistics.

Vessel Type	# trajectory Points	# trajectory Segments
Transit Ferry, V_1	27028	1186
Transit Ferry, V_2	103161	4263
Cargo, D_1	38852	24
Cargo D_2	8070	21

4.4 Reference Route Construction

After the trajectories are partitioned into segment and these segments are labeled, we construct *Reference Routes of Trajectory (RRoT)*. A reference route is a mean segment representing segments belonging to the same path label between a pair of ports (step 2). The reference route construction consists of two steps:

Step one is to interpolate the segments' points, i.e., latitude and longitude, which ensures the equality of segments' lengths when calculating their average. linear interpolation is used because it is the simplest and consumes the least computational power. The steps for generating a reference route are described in Algorithm 1. The input of this algorithm is all segments that belong to the same route. For each segment (line 1), the time is transformed to an increasing number (line 2), and then spacing is applied to create a list of required data points (line 3). Then, we pass the list of trajectory points created from the spacing method to the linear interpolation method, see Appendix B.1. Then, trajectory points are interpolated independently for the longitude values (line 4) and the latitude values (line 5). We have opted for an oversampling technique by selecting 500 as the number of interpolated points. This approach aims to improve anti-aliasing performance and enhance the overall resolution. In step two, each mean segment is calculated from all segments between two ports that belong to the same *route* (lines 10, 11, and 12). Finally, it returns the mean segment as a reference route. We implemented the linear interpolation and space functions in the NumPy Python library.

Algorithm 1: Reference Route Algorithm

Input: Trajectory segments with the same route labels `// All segments that has the same origin and destination ports;`
output: $Reference\_Route$ `// An average segment represents the mean of the segments in each cluster.`
for *each segment in segments with the same route labels* **do**
    `// converting time to seconds format;`
    $tm \leftarrow (segment[time])$;
    `// create an evenly spaced sequence in the specified period of tm.` $interp\_time \leftarrow linspace(tm[0], tm[-1], num\_points)$;
    `// functions return one-dimensional piece-wise linear interpolated lon/lat with given discrete data points (tm, lon/lat_points), evaluated at interp_time.`
    $interp\_longitude \leftarrow interpolate(interp\_time, tm, Longitude\_points)$;
    $interp\_latitude \leftarrow interpolate(interp\_time, tm, Latitude\_points)$;
    `// sum the time and coordinates values of the segments.`
    sum_interp_time;
    $sum\_interp\_longitude$;
    $sum\_interp\_latitude$;
end
`// compute the average time and average coordinates.`
$avg\_time \leftarrow mean(sum\_interp\_time)$;
$avg\_longitude \leftarrow mean(sum\_interp\_longitude)$;
$avg\_latitude \leftarrow mean(sum\_interp\_latitude)$;
`// Concatenate the averages and store the results in Reference_Route.`
$Reference\_Route \leftarrow (avg\_time, avg\_longitude, avg\_latitude, label, route)$;
return $Reference\_Route$;

The reference routes of the ferryboat's trajectory data, V_1, and the reference routes of cargo vessels' trajectory data, D_1, are shown in Figures 4.7, 4.8 respectively. These figures represent summarized maritime traffic networks (graphs), where nodes represent locations (such as ports), and edges represent the connections between them (the reference routes traveled by vessels). This graph representation assists in identifying common routes and patterns followed by vessels over time while also illustrating the connectivity between different ports based on vessel movements, represented by smooth, continuous reference routes. The segments of a ferryboat, V_2, and a cargo vessel, D_2, will be used as test sets to detect the segments' destinations. Therefore, reference routes are not constructed for these vessels (V_2, D_2).

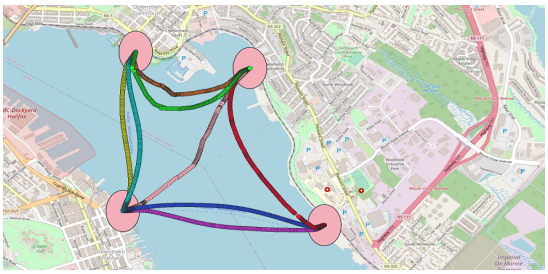


Figure 4.7: The reference routes of ferry trajectory data, V_1

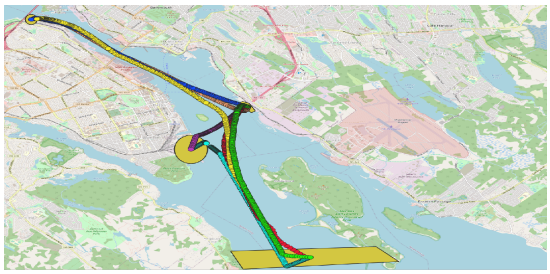
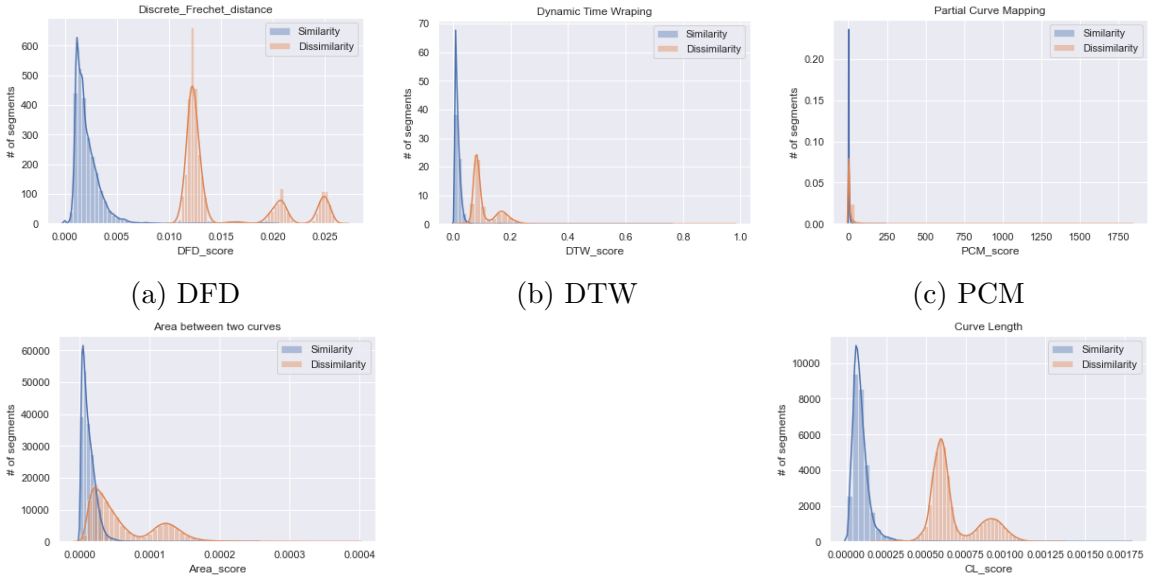


Figure 4.8: The reference routes of cargo trajectory data, D_1

4.5 Best Similarity Measures

To effectively compare the similarity between the reference routes and the ongoing route, five similarity measurements are explored. (1) Discrete Fréchet Distance (DFD) [84], (2) Dynamic Time Warping (DTW) [43], (3) Partial Curve Mapping (PCM) [44], (4) Area between two curves (Area) [13], and (5) Curve length (CL) [13]. These five measures are selected because they are parameter-free, which makes the implementation of the detector simple and efficient since there is no interruption for the algorithm to estimate and input the proper parameters. We assess the performance of these five measurements by observing the distributions of similarity and dissimilarity scores. It is a simple way to quantify the difference between each method's similarity and dissimilarity distributions. If a method has both distributions overlapped, it will not perform well in quantifying the differences between the compared routes and will be eliminated. The similarity measurements' performance is investigated using the distribution of quantified differences between all segments of the transit ferry dataset, V_1. For similar segments that have the same origin and destination ports, we calculate their similarity differences and make a distribution of the acquired scores. For the dissimilar segments, we choose a segment that represents one *route*, compare it with other segments, and make a distribution of the acquired dissimilarity scores.

The similarity and dissimilarity distributions are visualized side-by-side to infer each method's performance, as shown in Figure 4.9. The plots demonstrate that in the plot 4.9c PCM and the plot 4.9d Area methods, their similarity and dissimilarity distributions overlap. This means the proposed models using these methods tend to perform poorly in destination port classification and prediction. Therefore, these two



(a) DFD (b) DTW (c) PCM

(d) Area (e) CL

Figure 4.9: A depiction of similarity and dissimilarity distributions of the five similarity measures.

methods are eliminated. DFD, DTW, and CL performed the best across our simulation in distinguishing the different distributions of the similarity and dissimilarity scores. Therefore, these three similarity measures will be used to classify the ongoing voyage based on the extracted reference routes.

4.6 Performance Assessment Criteria

We consider this a multi-class classification problem because we need to predict which destination each route will travel to. To evaluate prediction performance, we consider accuracy, the F1-score, and the confusion matrix. We decided to use these metrics because accuracy alone can be misleading, as we have an unequal number of segments belonging to each path.

The confusion matrix is a tabular way of visualizing the prediction model's performance. Each entry in a confusion matrix denotes the number of predictions made by the model correctly or incorrectly. The X-axis contains the predicted reference routes, and the Y-axis includes the actual reference routes. The diagonal values are TP (true positive) values. A true positive is when the model correctly predicts the correct reference route. Precision measures, out of all predicted positives, how many

are actually positive. Precision focuses on predicted values (columns). Recall measures how many positive instances are predicted correctly. Recall focuses on actual values (rows). The F1-score is the harmonic mean of precision and recall.

$$F1\_score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4.1}$$

4.7 Destination Prediction

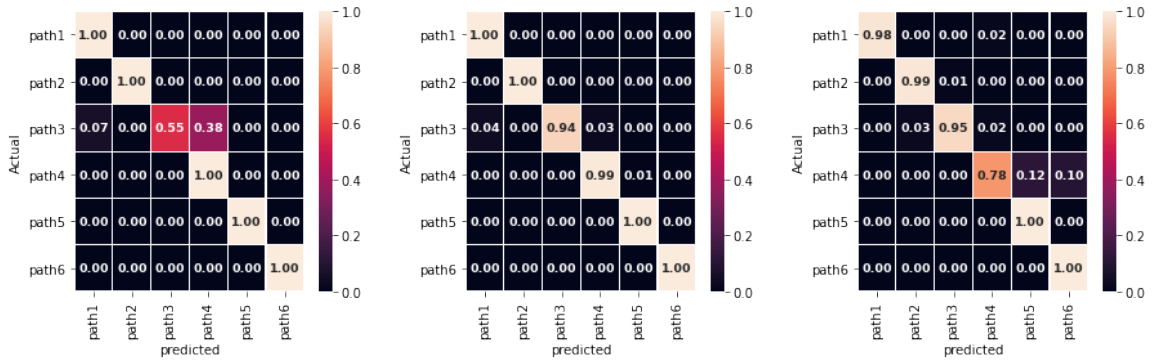
As vessel trips rarely follow a random trajectory, the prediction of the destination port could be considered a classification problem. The goal of the following experiments is to effectively predict the destination port by using the last points of a segment (sub-segment) before the destination port and calculating its similarity to the constructed reference routes. The reference route destination, which shares the highest similarity, is predicted to be the current sub-segment destination. Therefore, the proposed approach will be validated with two datasets of two different vessel categories: ferryboats and cargo vessels.

4.7.1 Destination Prediction of Ferryboats' Trajectories

The 4263 segments of V_2 are used as test data. First, a list of labels (i.e. $path_1$, $path_2$, $path_3$, $path_4$, $path_5$, $path_6$) is created, which represents the labels of the 4263 segments. Each segment's label is compared to the constructed reference routes' labels, and the corresponding route_id is appended to the list. Then, we choose the last six trajectory points of each segment (sub-segment) before the destination port. Based on the capture rate, six trajectory points provide the minimum information we need to calculate similarity. Then, the similarity between these six trajectory points and the reference routes is calculated.

Each reference route consists of 300 trajectory points. Using the three selected similarity measurements: Discrete Fréchet Distance (DFD), Dynamic Time Warping (DTW) and Curve Length (CL). The sub-segment destination is predicted based on its highest similarity to a reference route, along with its destination port. Therefore, to compare the performance of the selected similarity measures, first, the performance of each method is visualized using the multi-class confusion matrix. Then, the accuracy

and the F1 measure are reported for each measure. Figure 4.10 presents a visual interpretation of the confusion matrices, where the prediction output for the three similarity measurement models has six routes: $path_1$, $path_2$, $path_3$, $path_4$, $path_5$, and $path_6$. The diagonal elements represent the correct predictions per route. The lighter the color, the greater the number. Off-diagonal elements are mislabeled.



(a) Prediction performance using Discrete Fréchet Distance.

(b) Prediction performance using Dynamic Time Warping.

(c) Prediction performance using Curve Length.

Figure 4.10: Confusion matrices of destination port prediction performance using similarity methods: (a) Discrete Fréchet Distance, (b) Dynamic Time Warping, (C) Curve Length.

The confusion matrix in Figure 4.10a represents the percentage prediction of each destination port made by the model using the Discrete Fréchet Distance (DFD) similarity measure. Diagonal elements for $path_1$, $path_2$, $path_4$, $path_5$, and $path_6$ are perfectly predicted. However, it performs comparatively poorly for $path_3$; for all true destination ports for $path_3$, it only predicts 55% of them correctly. The confusion matrix in Figure 4.10b represents the percentage prediction of each destination port made by the model using the Dynamic Time Warping (DTW) similarity measure. Diagonal elements for $path_1$, $path_2$, $path_5$, and $path_6$ are perfectly predicted. Next comes $path_4$ with 99% correct predictions. Then comes $path_3$ with 94% correct predictions, which means that for this particular route, this DTW model outperforms (DFD). The confusion matrix in Figure 4.10c represents the percentage prediction of each destination port made by the model using the Curve Length (CL) similarity measure. Diagonal elements for $path_5$ and $path_6$ are perfectly predicted. Next comes $path_2$ with 99% correct predictions. Then $path_1$ with 98% correct predictions. After

that comes $path_3$ with 95% correct predictions, which means that for this particular route, this model of CL outperforms the model of (DFD) and is slightly better than the model of DTW. Then, $path_4$ could predict only 78%.

From the confusion matrices, we can infer that the discrete Fréchet distance (DFD) is a max-measure, defined as the maximum distance measured at each position. The dependence on the maximum value of distance leads to non-robust behaviour, where some variation in the sub-segments related to $path_3$ distorts the distance function by a large amount. Thus, the percentage prediction for $path_3$ is significantly low. Dynamic Time Warping (DTW) is a sum measure, defined as the sum of the distance measured at each position. Hence, this measure smooths the distortion in the DFD model. Thus, the DTW model's percentage prediction for $path_3$ is significantly improved. Curve Length (CL) is a measure of the i^{th} point of the sub-segment to the corresponding equivalent length of the curve of the corresponding route. The i^{th} point of the sub-segment does not correspond to the same abscissa, as in the DFD and DTW models, where some variation in the sub-segments can distort the distance function by some amount, as in $path4$ but notably less than the DFD model.

The destination prediction accuracy and f1 measure for DFD, DTW, and CL are shown in Table 4.2. The model of DTW surpasses the other two methods in accuracy and f1. As a result, DTW is a very robust technique to compare the peaks and troughs by taking into account the varying lags and phases in the trajectories.

Table 4.2: Accuracy and f1 measure of the three selected models: Discrete Fréchet Distance, Dynamic Time Warping and Curve Length

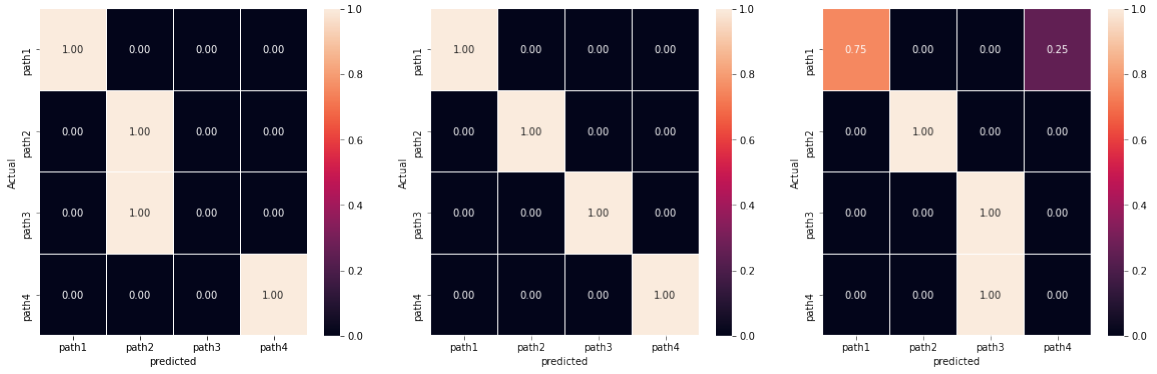
Discrete Fréchet Distance		Dynamic Time Warping		Curve Length	
Acc.	f1	Acc.	f1	Acc.	f1
95.82%	95.31%	98.97%	99.08%	89.75%	93.58%

4.7.2 Destination Prediction of Cargo Vessels' Trajectories

The 21 segments of cargo vessel D_2 are used as test data. First, a list of labels (i.e., $path_1$, $path_2$, $path_3$, $path_4$) is created, which represents **the labels** of the 21 segments. Each segment's route label is compared to the reference routes' labels, and the corresponding route_id is appended to the list. Based on the capturing rate, the choice of the last trajectory points of a segment before the destination port must

provide the minimum information to effectively calculate the similarity. Therefore, the choice of the number of last trajectory points of a segment depends on the length of its shipping lane. So, for long segments, we need to choose the last 200 trajectory points of each segment before the destination port. For segments following other paths, we need to choose the last 50 trajectory points of each segment before the destination port.

Each reference route consists of 1000 trajectory points. Using the three selected similarity measurements: Discrete Fréchet Distance (DFD), Dynamic Time Warping (DTW) and Curve Length (CL). The reference route destination, which shares the highest similarity with the sub-segment, is predicted as the current sub-segment destination. Therefore, first the performance of each method is visualized using the multi-class confusion matrix. After that, the F1 measure and accuracy are presented. Figure 4.11 provides a visual interpretation of the confusion matrices, where the prediction output for the similarity measurement models has four routes: $path_1$, $path_2$, $path_3$, $path_4$. The diagonal elements represent the correct predictions per route. The lighter the colour, the greater the number. Off-diagonal elements are mislabeled.



(a) Prediction performance using Discrete Fréchet Distance.

(b) Prediction performance using Dynamic Time Warping.

(c) Prediction performance using Curve Length.

Figure 4.11: Confusion matrices of destination port prediction performance using similarity methods: (a) Discrete Fréchet Distance, (b) Dynamic Time Warping, (C) Curve Length.

The confusion matrix in Figure 4.11a represents the percentage prediction of each destination port made by the model using the Discrete Fréchet Distance (DFD) similarity measure. Diagonal elements for $path_1$, $path_2$, and $path_4$ are perfectly predicted. However, it couldn't predict all true destination ports for $path_3$; all segments of $path_3$ are mislabeled as $path_2$. The confusion matrix in Figure 4.11b represents the percentage prediction of each destination port made by the model using the Dynamic Time Warping (DTW) similarity measure. Diagonal elements for $path_1$, $path_2$, $path_3$, and $path_4$ are perfectly predicted. This DTW model outperforms the DFD. The confusion matrix in Figure 4.11c represents the percentage prediction of each destination port made by the model using Curve Length (CL). Diagonal elements for $path_2$ and $path_3$ are perfectly predicted with 100% correct predictions. Next comes $path_1$ with 75% correct predictions. Then $path_4$ is totally mispredicted.

The destination port accuracy and f1 measures for DFD, DTW, and CL are shown in Table 4.3. The DTW model surpasses the other two methods in accuracy and f1. As a result, DTW is a robust technique to measure similarity between trajectories.

Table 4.3: Accuracy and f1 measure of the three selected models: Discrete Fréchet Distance, Dynamic Time Warping and Curve Length

Discrete Fréchet Distance		Dynamic Time Warping		Curve Length	
Acc.	f1	Acc.	f1	Acc.	f1
94.74%	92.24%	100%	100%	84.21%	86.96%

4.8 Limitations

The proposed prediction model exhibits several limitations. Firstly, it was tested only on a small dataset of vessel trajectories from Halifax port, which has a relatively low maritime traffic density and complexity. Secondly, as the volume of trajectory data expands, the computational demands for annotation increase, presenting scalability challenges. Additionally, this process necessitates specialized expertise, which makes it highly labor-intensive, time-consuming, and cost-ineffective. Subject matter expert verification is important to ensuring high-quality ground truth, reducing bias and subjectivity, and validating automated annotations to ensure accurate and reliable data for the prediction model. Thirdly, the method is constrained to producing a

single reference for segments sharing an identical path number, which may result in navigating through no-sail zones if such zones separate the segments.

In the next chapters, we intend to address these challenges through the following strategies: (1) Utilizing datasets from high-density maritime traffic areas characterized by longer-duration vessel trajectories and complex movement patterns. (2) Investigating a semi-supervised clustering algorithm that incorporates background knowledge of ports to generate clustering constraints, particularly must-link constraints, to guide the clustering process. This approach aims to mitigate the bottleneck associated with data annotation. (3) Enhancing the construction of reference routes by extracting multiple movement patterns corresponding to fine-grained AIS data clustering within the same network lane (connection).

4.9 Conclusions

This chapter explores the initial design of our model for forecasting the destination ports of moving vessels. The model incorporates a novel traffic network representation derived from summarizing vessel trajectories, and utilizes trajectory similarity measures to determine vessel destinations based on recent movements. The summarized representation retains the essential shape of trajectory[3], capturing spatial relationships and connectivity patterns, and providing adaptability in reflecting observed maritime routes without reliance on predefined route networks or fixed distance thresholds. Consequently, the maritime traffic network topology inferred from trajectory patterns, reveals the connectivity patterns of maritime routes and traffic flows, facilitating flexible modeling of intricate traffic patterns and dependencies. By utilizing the extracted traffic network, the prediction model enhances destination forecasting and decision-making in maritime operations. The model predicts destination ports by comparing recent movement data with reference routes, with the predicted destination port corresponding to the reference route with the highest similarity. As vessels away from their destination, the model predicts multiple potential destination ports along with associated probabilities. These predictions dynamically change throughout the voyage, allowing for updates to the probability of arrival at each port.

[3]The essential shape of a trajectory refers to the fundamental geometric path or pattern described by the trajectory points. It captures the core spatial characteristics of how the trajectory unfolds in space.

The purpose of the model is to support human decision-making processes; thus, a probabilistic approach is of value as it allows the model to express uncertainty. This, in turn, allows humans to determine when they can trust a model, particularly when it assigns a high probability to a single destination.

Chapter 5

Maritime Traffic Network Extraction using Novel Two-Step Clustering for Multiple Patterns on the Same Edge

The Automatic Identification System (AIS) supplies vessels' tracking data, which plays a crucial role in maritime navigation and safety. Although no physical roads exist at sea, vessels commonly adhere to *de facto maritime routes* for fuel efficiency, and security reasons. Intelligent systems in maritime transportation utilize clustering methods to identify common traffic patterns of vessels, enabling proactive decision-making and risk mitigation. However, trajectory clustering poses significant challenges: current algorithms struggle due to their reliance on geometric assumptions like shape or density, which can overlook clusters that don't fit predefined shapes or have varying densities. Additionally, the need for full similarity matrices is causing significant computational burdens, limiting scalability and effectiveness. To address these issues, this chapter[1] introduces a novel clustering method that can identify non-linearly separable clusters with varied densities and shapes of vessels' trajectory segments in linear time.

5.1 Introduction

Clustering is a valuable tool for uncovering patterns and commonalities in maritime trajectories. However, traditional clustering algorithms are constrained by the subjectivity inherent in configuring input parameters to determine the number or densities of clusters [53, 69, 78, 82, 83, 91, 99]. This subjectivity arises because there is no objective standard, given the absence of definitive "true" clusters [53, 34]. Another challenge for clustering algorithms is the computational expense associated with full similarity matrix calculations. While segmenting vessel trajectories based on course and speed over ground has limitations in capturing the intricacy of vessel behavior,

[1] This chapter is based on the publications [30, 29]

particularly in dynamic maritime environments where these attributes are influenced by external factors such as weather and ocean currents, leading to less accurate segmentation, this work takes a different approach. Here, vessel trajectories are segmented based on the ports' areas, achieved by combining AIS data with port data. The goal is to efficiently generate *"complete clusters"*, each includes all trajectory segments (i.e., routes) following the same direction between port pairs. Each generated cluster corresponds to a single shipping lane within the maritime traffic network. Subsequently, it is possible to cluster segments within each cluster once more in order to extract multiple potential movement patterns for the same network lane. Figure 5.1 depicts the proposed technique.

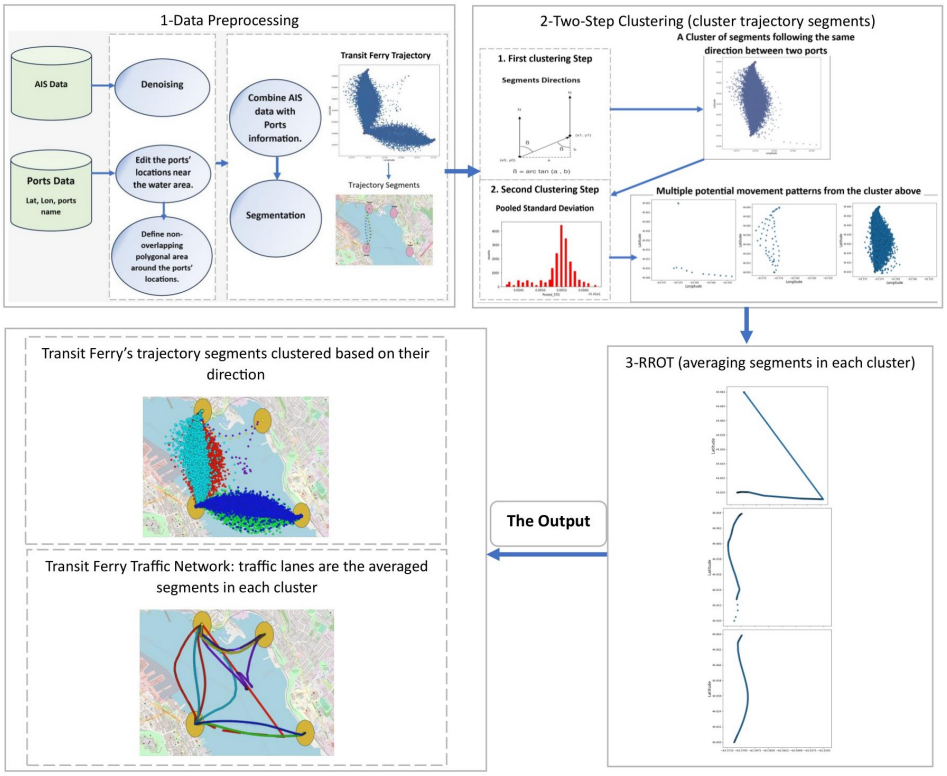


Figure 5.1: A Step-by-Step guide to extracting continuous maritime routes using trajectory segments clustering framework.

The contributions of this chapter are: (i) we present a framework detailing the process of extracting continuous and smooth maritime traffic networks with potential movement patterns for the same shipping lane from AIS datasets. (ii) We

present a novel two-step clustering technique, Spatial Clustering of Vessels' Trajectories Segments (SPTCLUST), for extracting ship routes between port pairs. This semi-supervised clustering approach aims to deduce must-link constraints from origin-destination port points to facilitate the detection of maritime routes that accurately reflect real-world traffic flow in linear time. It also aims to generate interpretable clusters and automatically determine the number and densities of clusters. (iii) We evaluate the proposed approach on four real-world AIS datasets from two different areas with varying traffic densities. The experiments demonstrate that our clustering technique produces higher-quality clusters in less time and achieves competitive results across all datasets.

5.2 Definitions and Preliminaries

Definition 5.2.1 (Bearing)**.** In maritime navigation, a "**bearing**" is the clockwise angle between two points on the earth's surface, measured from true north, and calculated using spherical trigonometry to account for earth's curvature. Let point B and C have positions $(\text{lat}^1, \text{lon}^1)$ and $(\text{lat}^2, \text{lon}^2)$, respectively. Let point A be the North Pole as shown in Figure 5.2. The angle Δ is the difference between the longitudes. The angle β, representing the bearing from B to C, $\beta_{B\to C}$, can be calculated using the following relations: $\beta = atan2(X, Y)$, where X and Y can be calculated as, $X = cos(lat^2) \times sin\Delta$, $\Delta = lon^1 - lon^2$, $Y = cos(lat^1) \times sin(lat^2) - sin(lat^1) \times cos(lat^2) \times cos\Delta$ [89]. If $X = 0$ and $Y = 0$, indicating that two geographic points are identical, the bearing is defined to be zero.

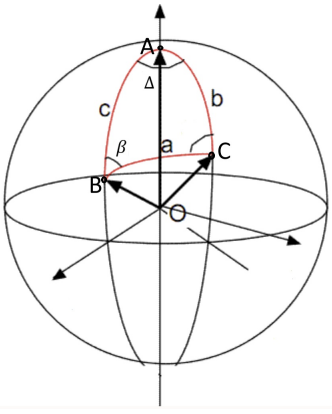


Figure 5.2: A spherical triangle on a sphere. Image imported from [2].

Definition 5.2.2 (Segments' Endpoints' Similarity)**.** The measure of positional and directional similarity between trajectory segments, s_1 and s_2, is determined by the bearing values of their origin and destination points. Given origin points $(\text{lat}^1, \text{lon}^1)$ for O_1 and $(\text{lat}^2, \text{lon}^2)$ for O_2, for two segments, the bearing from O_1 to O_2 is given by $\beta_{O_1 \to O_2}$; see definition 5.2.1. Similarly, the bearing from D_1 to D_2 is given by $\beta_{D_1 \to D_2}$. The pair of bearings $(\beta_{O_1 \to O_2},\ \beta_{D_1 \to D_2})$ collectively defines the overall directional similarity between the two segments, which is referred to as the pair of bearings of the segments' endpoints.

Definition 5.2.3 (Cluster)**.** *A cluster*, C_i, is a set of trajectory segments, where all associated origin and destination pairs (O_i, D_i), representing the starting and ending locations of its segments (definition 4.2.3) are identical, and this pair is unique to that cluster and not found in any other cluster. Let C_i represent a cluster, and $\mathcal{S}$ denote the set of all trajectory segments. Each cluster C_i consists of trajectory segments s such that: $C_i = \{s \in \mathcal{S} \mid (O(s), D(s)) = (O_i, D_i)\ \forall\ s \in \mathcal{J}_i\}$. $\mathcal{J}_i$ is the index set of all segments in cluster C_i. (O_i, D_i) is the unique origin-destination pair for the cluster C_i. Thus $C_i \not\subseteq C_j\ \forall\, i \neq j, \therefore (O_i, D_i) \neq (O_j, D_j)$.

5.2.1 Problem Statement

Given a set of trajectory data $\tau = \{1, ..., |\tau|\}$ and a list of ports $P = \{P_1, ..., P_v\}$, where v is the number of ports, our objective is to extract a set of potential reference routes $R_{i,j} = \{r_1, ..., r_m\}$ between P_i and P_j. Each reference route $r_t \in R$ represents a unique navigable path for a vessel, ensuring that $r_z \neq r_t$ for all $1 \leq t \leq m$, $1 \leq z \leq m$, and $t \neq z$.

5.3 Datasets

Two datasets were selected for the performance evaluation in this study: (i) AIS data from Halifax Harbour and (ii) AIS data from the Gulf of Mexico ocean basin. **AIS data from Halifax Harbour:** was collected at a latitude of 44° 34' 51.8952"N and 44° 40' 49.512"N and at a longitude of 63° 45' 18.7884"E and 63° 26' 42.2448"W for the period from March to July 2019. We selected two types of vessels from this dataset. Transit ferry data has 103162 trajectory points (definition 4.2.1), and cargo

vessel data has 38853 trajectory points. **AIS data from the Gulf of Mexico** was collected at a latitude of 18°N and 30°N and at a longitude of 79°W and 97°W for the period from 01-03-2021 to 01-10-2021. We chose historical AIS data for cargo and tanker vessels, assuming that vessels of the same type share similar route patterns. The AIS dataset of cargo vessels includes 2046 cargo trajectories with a total of 218213 trajectory points. The tanker vessels' dataset consists of 1846 trajectories comprising 229471 trajectory points. Figures 5.3 and 5.4 give an overview of AIS data in the selected areas on the map using QGIS 3.26.

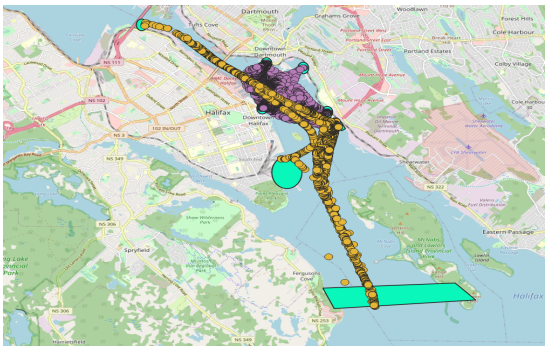


Figure 5.3: An overview of AIS dataset from Halifax Harbour.

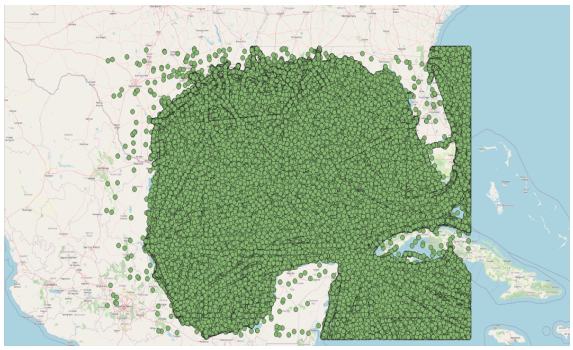


Figure 5.4: An overview of AIS dataset from Gulf of Mexico.

5.4 Data Preprocessing

AIS data cannot function alone; it needs to be projected into *geographic environment information* to be analyzed more effectively. ports' information, important for geographic insights, generally exists or can be extracted from a comprehensive digital map platform, such as the WPI (World Port Index)[2]. The main focus of this study is on latitude and longitude coordinates of ports as well as their names. In our previous work on vessel destination prediction, we demonstrated that AIS messages do not provide trustworthy information about ports of destination [28]. To extract trajectory segments and develop a more reasonable and rational explanation for patterns and behaviours detected in a vessel's trajectory, ports' information is typically linked with AIS data. To ensure the quality of the dataset, we cleaned the AIS data by removing duplicate records and observations outside the target area or on land, and

[2]https://msi.nga.mil/Publications/WPI

we validated the spatial distribution of the data using GIS applications. Figure 5.5 shows the AIS data in selected areas on a map using QGIS 3.26, while Figure 5.6 provides an overview of the cleaned AIS data in the selected areas.

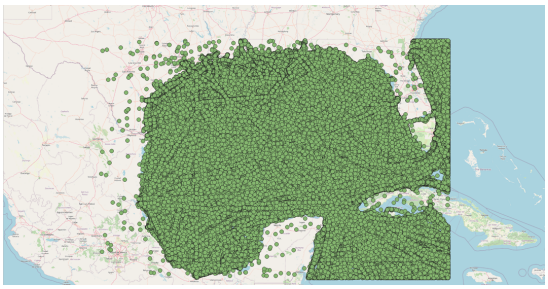


Figure 5.5: Overview of AIS dataset from Gulf of Mexico.

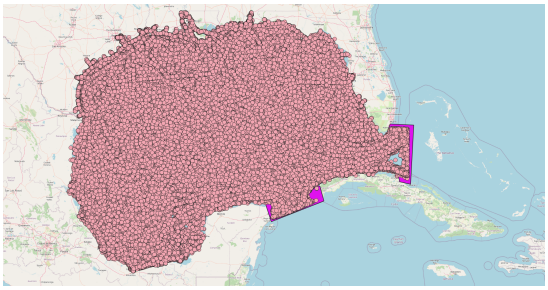


Figure 5.6: Cleansed AIS dataset from Gulf of Mexico.

5.4.1 Port-based data annotation

Identifying origin-destination (OD) points has been a widely used approach in handling spatiotemporal data across various trajectory types, including vehicle, human mobility, and AIS data [28, 27, 29, 56, 81, 93]. Incorporating port details is crucial for identifying and extracting trips from an origin port (O) to a destination port (D), with the positions of the ports being referenced by WPI (World Port Index) data. The AIS data labeling process (as depicted in Algorithm 2) is executed by integrating AIS and port data, leveraging MovingPandas [3], a Python library for managing movement data using Pandas and GeoPandas. The process encompasses the following steps:

1. Edit Ports Locations: The geographic coordinates (latitude and longitude) of the ports, obtained from World Port Index data (WPI), need to be relocated near the water area, as many of the locations are inside the city.

2. Generation of port area: To determine if a vessel's trajectory point falls within a port zone, circles are generated around all ports within the target region. This is implemented using the Python Geopandas package. A proximity threshold needs to be determined to identify when a vessel is considered to be at or near a port. This threshold can vary based on the accuracy of the AIS data and subjective criteria.

[3] https://geopandas.org

3. Data annotation: Annotate the AIS data with port points. When a vessel's position is within the defined proximity of a port, mark that position as a port point in the dataset. On the other hand, points outside the defined port proximity are labeled as "move".

The goal of this data annotation method is to facilitate the next step, which is trajectory segmentation. Formally, let τ be the set of AIS data records, where $\tau \in \mathbb{R}^d$ is the d-dimensional feature space. Let L be the set of corresponding labels or classes associated with the AIS data records in τ. $L = \{move, Ports\}$, where $Ports = \{(name_i, lat_i, lon_i)\}$, $1 \leq i \leq m$. For each data record $\iota \in \tau$, there exists a corresponding label $l \in L$, which represents the class or category to which ι belongs. This can be denoted using a function: $f : \tau \longrightarrow L$, where f is a mapping function that assigns labels to the input data records. The function f maps each data record ι to its corresponding label l.

$$f_{\iota+[l]} = \begin{cases} Port\_name & \text{if } \iota_j(lat, lon) \in Ports \\ Move & \text{if } \iota_j(lat, lon) \notin Ports \end{cases} \tag{5.1}$$

Alternatively, if we have a dataset consisting of n data samples, we can represent the labeling using a labeled dataset: $D = \{(\iota_1, l_1), (\iota_2, l_2), ..., (\iota_n, l_n)\}$, where (ι_j, l_j) represents a labeled data record, with $\iota_j \in \tau$ being the input data and $l_j \in L$ being its corresponding label.

[2]Azimuthal equidistant (AEQD) projection ensures accurate straight-line distances from the center point but does not preserve true directions at all graticule line intersections.

[3]World Geodetic System (WGS84) is a coordinate reference frame for establishing latitude, longitude points.

Algorithm 2: AIS data Labelling

Input: preprocessed trajectories $T = \{\tau_1, ..., \tau_n\}$, and Ports data
$Ports = \{(name_1, lat_1, lon_1), ..., (name_m, lat_m, lon_m)\}$
Result: Annotated trajectories with ports labels

```
/* Generate polygons around ports' locations                    */
polygons ← {}
for each i in Ports do
    latitude ← lat_i;
    longitude ← lon_i;
    radius ← input desired circle radius in meters;
    /* Get polygon with lat,lon coordinates in AEQD space^2     */
    point_azimuthal ← Transform((lon, lat), aeqd));
    /* creating a shapely buffer on the projected point         */
    buffer ← buffer(point_azimuthal) for radius;
    /* projecting the shapely buffer back to WGS84 space^3      */
    circle ← transform.to.WGS84(buffer);
    polygons_i ← Polygon(circle.exterior.coords);
end
add polygons as a new variable to Ports;
add polygons_centers as a new variable to Ports;
/* Ported Points detection and labelling                        */
Labels ← {};
for each p in T do
    for each i in Ports do
        if Port_i[polygon] contains p then
            Labels ← (name_i);
            else
                Labels ← (move) ;
            end
    end
end
add Labels as a new variable "feature or attribute" to T;
```

5.4.2 Segmentation

This study uses the annotated data to divide a trajectory τ into ship routes (i.e., segments), beginning at one port and ending at the next port [28, 93]. A *vessel's trajectory* can be defined as a series of ports and locations between them. A *trajectory segment* is defined by a sequence of locations that start at one port and end at the next port in the trajectory. We refer to the start and end ports of the segments as the

origin and destination (OD) points. The segmentation process segments the trajectory according to the positions of origin-destination points to ensure homogeneity of segments. We have a total of 42 ports: 40 ports designated for cargo and tankers, and two polygons marking the entrances from the east and south of the Gulf of Mexico area, indicating the points of entry and exit from this region.

Next, we suggest the **"segmentation trick"**, which maps the positions of endpoints of several segments inside a single port area to a shared position in common (i.e., the centriods of ports' zones). As seen in Figure 5.8, the endpoints of the segments are scattered among various locations within each port. This distribution across different positions within each port will produce inconsistent values with any categorization method, which makes clustering and classification calculations difficult. We use a workaround to solve this problem, which designates the start and end points of the associated segments as the center coordinates of the ports' zones. This trick guarantees that the endpoints of routes within the same port are identified by the categorization technique as being at the same position. Algorithm 3 describes the segmentation process.

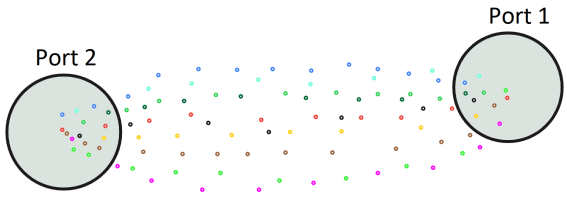


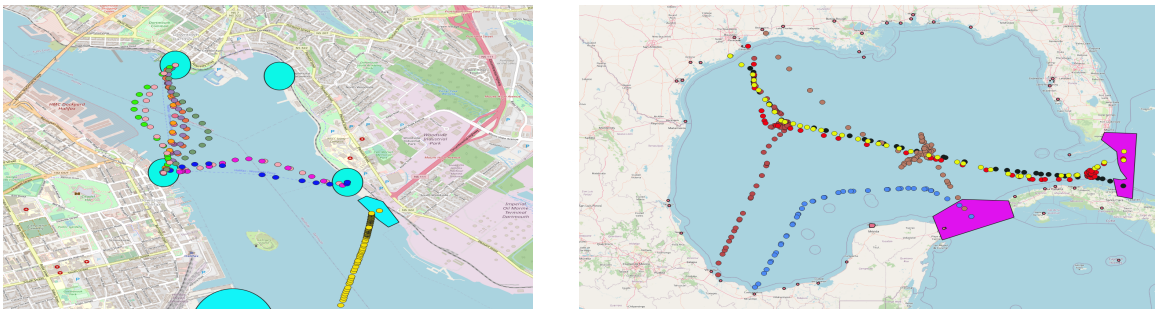
Figure 5.7: Ship trajectory segments between two ports.

Let *segment* denotes a trajectory segment. In this context, $segment_0$ represents the segment's first data record, and $segment_{-1}$ represents the segment's last data record. We employ two variables, $ind1$ and $ind2$, to store the indices of the port data corresponding to the segment's origin and destination points, respectively. To maintain the original data sequence of the segment: (1) We duplicate the first data record from the segment and replace the latitude and longitude coordinates with the central points of the associated port, $[lat, lon]$; signifying the origin point of the segment. This duplicate is then inserted as the first data record within the segment's data records. (2) Similarly, we replicate the last data record from the segment, substitute the latitude and longitude coordinates with the central points of

Algorithm 3: Ship routes extraction

Input: preprocessed labelled trajectories $T = \{\tau_1, ..., \tau_n\}$, and Ports data $Ports = \{(name_1, lat_1, lon_1), ..., (name_m, lat_m, lon_m)\}$;
Output: Ship routes (trajectory segments);
/ Trajectory segments are obtained by extracting AIS points between the starting and ending ports, with the centre coordinates of the ports added as the start and end points for each segment. */*
$segment \leftarrow \{\}$
for i *in each* τ **do**
    if $\tau_i[label] \neq \tau_{i+1}[label]$ **then**
        $segment.append(\tau_{i-1}, \tau_{i+2})$;
    end
    $ind1 \leftarrow index(segment_0(label) \leftarrow Ports(name))$;
    $ind2 \leftarrow index(segment_{-1}(label) \leftarrow Ports(name))$;
    $segment_0[lat, lon] \leftarrow Ports_{ind1}(polygons\_centers)$;
    $segment_{-1}[lat, lon] \leftarrow Ports_{ind2}(polygons\_centers)$;
end
save segment to a file;

the corresponding port, $[lat, lon]$, signifying the destination point of the segment, and insert this duplicate as the last data record within the segment's data records, see appendix A.3. Figure 5.8 displays some trajectory segments on a map using QGIS 3.26. Table 5.1 presents the number of segments generated for each vessel trajectory.



(a) Segments from Halifax Port dataset. (b) Segments from Gulf of Mexico dataset.

Figure 5.8: Overview of the segmentation result. Different colours represent different segments

Table 5.1: Vessels Trajectory Data Statistics.

Vessel Type	# trajectory Points	# trajectory Segments
Ferry (Halifax)	103162	4263
Cargo (Halifax)	38853	23
Cargoes (Gulf of Mexico)	218213	4847
Tankers (Gulf of Mexico)	229471	5750

5.5 Path Finding

This is the first clustering step in the SPTCLUST approach. The trajectory segments are clustered according to their position and directions between port pairs. To cluster the segments according to their similarity direction, the bearing is calculated between only the endpoints of the segments, which significantly decreases computational complexity. This procedure identifies the number of common shipping lanes between ports. Basically, our procedure clusters segments that follow the same direction between two ports into one group. The number of resultant groups (i.e., clusters) equals the number of common lanes between related ports in the studied area. Finally, it visualizes the groups of directions (i.e., clusters) to help the operator discover interesting patterns (see Figure 5.9). This visualization is also used to determine whether a direction group dg_i has multiple patterns from the origin port P_O to the destination port P_D, which aids in the next clustering step.

Let $\mathbf{s}_1$ and $\mathbf{s}_2$ be trajectory segments with endpoints $\mathbf{l}_1^O$, $\mathbf{l}_1^D$ and $\mathbf{l}_2^O$, $\mathbf{l}_2^D$, respectively. The coordinates of these endpoints are given by $\mathbf{l}_i^O = \begin{pmatrix} \text{lat}_i^O \\ \text{lon}_i^O \end{pmatrix}$ and $\mathbf{l}_i^D = \begin{pmatrix} \text{lat}_i^D \\ \text{lon}_i^D \end{pmatrix}$, where $\mathbf{l}_i^O$ and $\mathbf{l}_i^D$ denote the origin and destination points of the segment $\mathbf{s}_i$, respectively.. Let N be the cardinality of the set of all trajectory segments and B be the cardinality of the set of all common lanes between ports. Let $\mathbf{DG} = \{\mathbf{dg}_1, \mathbf{dg}_2, \ldots, \mathbf{dg}_m\}$ be the set of direction groups, where each group contains segments following the same direction between port pairs ($\mathbf{P_O}$, $\mathbf{P_D}$) and $m = B$. Let $\mathbf{B\_list} = \{(\beta_{\mathbf{l_1^O} \to \mathbf{l_i^O}}, \beta_{\mathbf{l_1^D} \to \mathbf{l_i^D}}) \mid 1 \leq i \leq n\}$ be the set of unique bearing pairs of the segments' endpoints. This set facilitates the mapping of segments to their corresponding clusters in $\mathbf{DG}$ and supports efficient data storage and retrieval, similar to the role of keys in Python dictionaries. Algorithm 4 presents the procedure for

grouping segments based on their direction similarity. The bearing is represented as a floating-point number rounded to the nearest hundredths to ensure consistency and conserve memory in large datasets.

Algorithm 4: Group based on Direction Algorithm

Input: A set of trajectory segments $S = \{s_1, s_2, ..., s_n\}$
Result: A set of groups of segments $DG = \{dg_1, dg_2..., dg_m\}$

$DG \leftarrow \{\{\}\}$ // list of lists; each inner list represents a group of segments. Each group has segments with the same origin and destination points;
$B\_list \leftarrow \{\}$// list of the segments' endpoint similarity, it is a list of pairs of bearings as in definition 5.2.2;
for *each segment* s_i *in* S **do**
    $\beta_{l^O} \leftarrow (\beta_{\mathbf{l_1^O} \rightarrow \mathbf{l_i^O}})$;
    // Calculate the bearing between origin points: from $\mathbf{l}_1^O$ in $\mathbf{s}_1$ to $\mathbf{l}_i^O$ in $\mathbf{s}_i$, as in Definition 5.2.1;
    $\beta_{l^D} \leftarrow (\beta_{\mathbf{l_1^D} \rightarrow \mathbf{l_i^D}})$;
    // Calculate the bearing between destination points: from $\mathbf{l}_1^D$ in $\mathbf{s}_1$ to $\mathbf{l}_i^D$ in $\mathbf{s}_i$, as in Definition 5.2.1;
    $Bearing\_Pair \leftarrow (\beta_{l^O},\ \beta_{l^D})$;
    // Create a pair of bearings;
    if *Bearing_Pair NOT in B_list* **then**
        $B\_list.add\{Bearing\_Pair\}$;
        // Append the pair of bearings to the B_list;
        $Pos \leftarrow$ index of $Bearing\_Pair$ in B_list;
        // Return the index of the newly added pair of bearings in the B_list;
        $DG(Pos).add\{empty\ list\}$;
        // Append empty list ''a new group'' to DG at the specified index ''Pos'';
    end
    else
        $Pos \leftarrow$ index of $Bearing\_Pair$ in B_list;
        // Return the index of the matched pair of bearings in the B_list;
    end
    $DG(Pos).add\{s_i\}$;
    // Add the segment to a particular list ''group'' within DG at the specified index ''Pos'';
end
return DG;

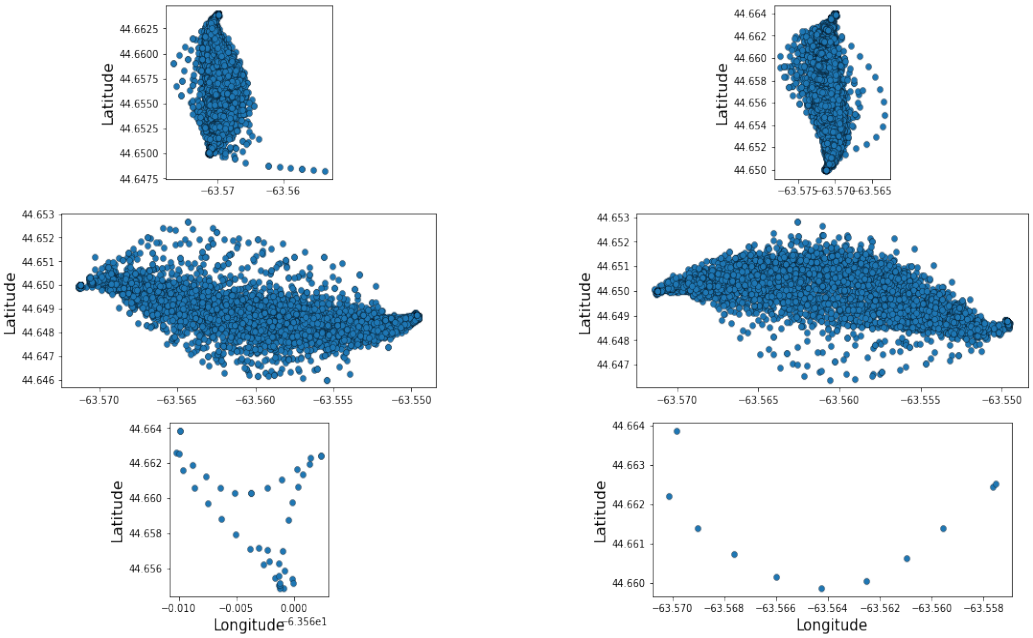


Figure 5.9: Direction groups of transit ferry trajectory data.

5.6 Movement Patterns Extraction

Since trajectory segments are grouped based on their direction similarity, the skeleton of the maritime traffic network is identified. At this point, we attempt to observe any discernible trends within each cluster to identify different possible movement patterns. Discovering multiple sailing lanes from port A (i.e., node A) to port B (i.e., node B) helps navigators decide one lane over another for navigation according to multiple factors: fuel consumption, avoiding undesirable conditions such as weather, traffic, geopolitical tensions, and other external factors. To distinguish potential patterns, a similarity measure with threshold values is defined to group similar patterns according to the desired lower limit for the similarity of two segments within the same cluster (Section 5.6.2). The threshold values are determined according to the distribution of the similarity values of the segments in each cluster (Section 5.6.1). After each cluster is constructed, segments within it are aggregated to form smooth, continuous lanes that comprise the maritime traffic network (Section 5.7).

5.6.1 Identify Threshold

We observed that, for trajectory clustering algorithms, threshold selection is highly sensitive to the data. Threshold selection is the key to effective clustering. To mitigate the manual selection of the threshold, we propose a method that uses pooled standard deviations (SD_pooled) [71, 45] to find the appropriate threshold. It is a method for estimating a single standard deviation to represent two trajectory segments' spatial coordinates in a direction group (i.e., cluster). This is because segments within a group are assumed to come from populations with a common standard deviation. Hence, the pooled standard deviations can be used to identify segments with similar patterns. A histogram is generated to visualize the distribution of the calculated SD_pooled values.

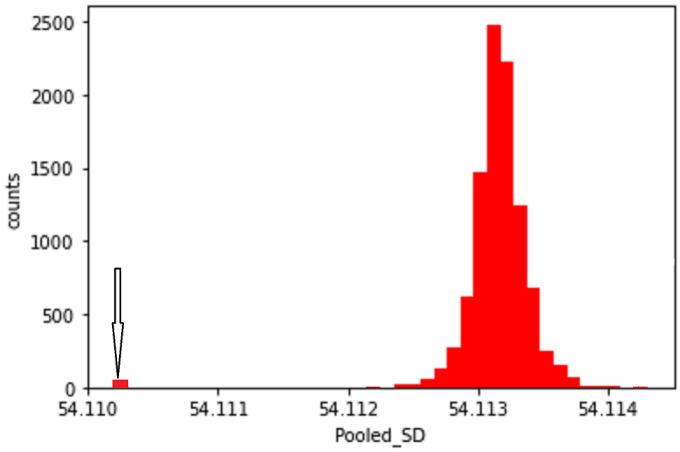


Figure 5.10: Histogram of pooled standard deviation values distribution and the arrow points to the outliers.

As shown in Figure 5.10 the arrow refers to extreme values, while most values cluster on the right of the histogram. If the user decides that there are two clusters, then we can use a threshold value equal to 54.111.

Let dg_1 be a direction group with segments travelling from P_O to P_D. Such that $dg_1 = \{s_1, s_2, .., s_k\}$, where $1 \leq k \leq N$, and N is the total number of trajectory segments. Θ is a set of pooled standard deviations SD_pooled between each two segments' points in dg_1, $\Theta = \{SD\_pooled_1, SD\_pooled_2, ..., SD\_pooled_{k-1}\}$. Because trajectory segments vary in length, linear interpolation is used to unify segments' lengths, Algorithm 5: lines (3, 4). The built-in Python function is used to interpolate

Algorithm 5: Identify Threshold Algorithm

Input: A group of trajectory segments following same direction
$dg_i = \{s_1, s_2, ..., s_k\}$

Result: A list of thresholds between the segments within the group dg_i
$\Theta = \{SD\_pooled_1, SD\_pooled_2, ..., SD\_pooled_{k-1}\}$

```
Θ ← {} // list of pooled standard deviation values between
 segments within dg_i; possible threshold/s, as in Equation 5.2;
for each segment s_i in dg do
    interp_lat ← np.interp(s_i.latitude) ;
    interp_lon ← np.interp(s_i.longitude);
    // linear interpolation to uniformly sample spatiotemporal
    positions along the length of s_i;
    coord ← (interp_lat, interp_lon);
    // get the interpolated coordinates of s_i;
    if s_i = s_1 then
        SD_1 ← np.std(coord);
        // Compute the standard deviation of the given coordinates
        of s_1 ;
        n_1 ← len(coord);
        // get the length of the interpolated coordinates of s_1;
    end
    else
        SD_2 ← np.std(coord);
        n_2 ← len(coord);
        SD_pooled(SD_1, SD_2, n_1, n_2);
        // Calculate the pooled standard deviation between s_1 and
        s_2; as detailed in Equation 5.2;
    end
    Θ.add(SD_pooled);
    // Append the pooled standard deviation value to the
    thresholds list;
end
return Θ;
```

the spatial coordinates of each segment. Then, the pooled standard deviation between the first segment's SD_1 and each other segment's SD_2 in the set dg_1 is calculated as follows:

$$SD_{pooled} = \sqrt{\frac{(n_1 - 1) * SD_1^2 + (n_2 - 1) * SD_2^2}{n_1 + n_2 - 2}} \tag{5.2}$$

Where n_1 is the length of the interpolated coordinates of s_1 and n_2 is the length of the interpolated coordinates of the compared segment s_i. Algorithm 5 is proposed to

identify the threshold/s of each group of directions. The histogram of Θ set values, will be used to determine the threshold(s) to cluster the segments in each direction group if it is possible. Otherwise, segments in a group of directions will be aggregated to create one reference route from one port to the next.

5.6.2 Second Clustering step

Once the threshold has been determined, the second step of clustering can be applied to extract multiple possible lanes between two ports using algorithm 6. The clustering algorithm 6 follows the same steps as the identifying threshold algorithm (Algorithm 5), except that it utilizes the pooled standard deviation to match the determined threshold and create clusters. By using this approach, the clustering algorithm is able to group trajectory segments that are similar in terms of their movement patterns, providing valuable insights into the common lanes taken by vessels.

5.7 Constructing Continuous and Smooth Summarized Network Lanes

To build a summarized maritime traffic network with continuous and smooth aggregated lanes connecting ports, a reference route is generated for each resulting cluster. This is achieved using the Reference Routes of Trajectory (RRoT) algorithm (Algorithm 1, explained in Section 4.4). The reference route is a mean segment that summarizes the segments within a cluster. The RROT function returns a set of potential reference routes, $R_{i,j} = \{r_1, r_2, ..., r_m\}$. The constructed reference routes are geometric objects formed as sequences of averaged interpolated locations of the segments within each cluster. The generated reference routes form a directed graph-based representation with continuous, smooth, and complete lanes, constituting the summarized maritime traffic network.

Algorithm 6: Cluster trajectory segments within same Direction group Algorithm

Input: A set of trajectory segments following same direction
$dg_1 = \{s_1, s_2, ..., s_k\}$ and Threshold/s

Result: A set of clusters $C = \{c_1, ..., c_u\}$

$C \leftarrow \{\{\}\}$;
for *each segment* s_i *in* dg **do**
  $interp\_lat \leftarrow np.interp(s_i.latitude)$;
  $interp\_lon \leftarrow np.interp(s_i.longitude)$;
  $coord \leftarrow (interp\_lat, interp\_lon)$;
  if s_1 **then**
    $SD_1 \leftarrow np.std(coord)$;
    $n_1 \leftarrow len(coord)$;
  end
  else
    $SD_2 \leftarrow np.std(coord)$;
    $n_2 \leftarrow len(coord)$;
    $SD\_pooled(SD_1, SD_2, n_1, n_2)$ `// calculations detailed in Equation 5.2`;
    if $SD_{pooled} \leq Threshold_1$ **then**
      C(0).add$\{s_i\}$;
    else if $SD\_pooled \leq Threshold_2$ **then**
      $C(1).add\{s_i\}$;
    end
    else if $SD\_pooled \leq Threshold_j$ **then**
      $C(r).add\{s_i\}$;
    end
    else
      $C(u).add\{s_i\}$;
    end
  end
end
return C;

5.8 Evaluation Metrics

To evaluate qualitatively the goodness of the clustering results; homogeneity [76], completeness [76], and the V-measure [76] are used here. Therefore, we applied the definitions of these metrics to evaluate our method. "Accurate clustering" is contingent upon each cluster $\mathbf{c}_i$ precisely identifying a group of segments possessing equivalent labels for their respective origin and destination points. The notion of *Completeness* is formally defined as the extent to which segments possessing equivalent labels for their start and end points are contained within a solitary cluster.

The property of *Homogeneity* dictates that the distribution of segment labels within each cluster should exhibit a bias toward a single class, with a consequent minimization of entropy.

The *V-measure* is a composite metric that reflects a harmonious balance between the measures of homogeneity and completeness, calculated as the harmonic mean of the aforementioned scores [76].

A paramount concern is the attainment of elevated levels of completeness and V-measure. The absence of substantial completeness and V-measure scores would render the extraction of maritime routes infeasible. Should a cluster contain segments traversing opposite directions between a specified pair of ports, the averaging of such segments would result in a minuscule arc or even a single point situated within the aquatic region on a map.

In this work, *completeness (cm), homogeneity (h), and V-measure (V)*, as shown in Equation 5.3, Equation 5.4, and Equation 5.5, respectively, are used. For the purposes of the following discussion, assume a dataset comprising N segments, and two partitions of these: a set of clusters $C = \{c_i | i = i, 2, ..., K\}$, where K is the total number of clusters generated by the algorithm, and a set of segments' labels within each cluster c_i, $\Lambda_i = \{\lambda_j | j = 1, 2, ..., n\}$, n is the total number of segments in c_i, $\widehat{\Lambda_i}$ is a set of desired labels whose elements are equal to the maximum occurring label of the segments within a cluster c_i. Formally, the Homogeneity score (h), completeness score (cm), and V-measure (v) are defined as [76]:

$$h_K = \frac{1}{K} \sum_{i=1}^{k} (1 - \frac{H(\widehat{\Lambda_i} | \Lambda_i)}{H(\widehat{\Lambda_i})}) \tag{5.3}$$

$$cm_K = \frac{1}{K}\sum_{i=1}^{k}(1 - \frac{H(\Lambda_i|\widehat{\Lambda}_i)}{H(\Lambda_i)}) \tag{5.4}$$

$$V_K = \frac{1}{K}\sum_{i=1}^{k}(2\frac{h_i.cm_i}{h_i + cm_i}) \tag{5.5}$$

$H(\widehat{\Lambda}_i|\Lambda_i)$ indicates the conditional entropy of the desired labels given the cluster assignments and $H(\widehat{\Lambda}_i)$ means the entropy of the desired labels [76].

5.9 Experiments on Clustering Trajectory Segments

This study examines the performance of the proposed clustering approach, SPTCLUST, in clustering segments of vessel trajectory and generating their summarized traffic network. The integration of ports' information with AIS data makes it easier to evaluate clustering by assigning reference labels to each cluster based on the origin and destination points of its segments [73]. These clusters are then assessed using completeness, homogeneity, and validity measures [76]. Initially, we evaluate SPTCLUST's clustering performance against four well-known standard clustering methods: Kmedoids, DBSCAN, OPTICS, and BIRCH, described in Section 3.2.2.1. Subsequently, we assess the effectiveness of constructing smooth, continuous, and interpretable reference routes from the clusters generated by each method. This process allows us to further evaluate the effectiveness of each clustering method in creating comprehensive maritime traffic networks with smooth, continuous lanes tailored to various vessel types.

5.9.1 Results and Discussions

Table 5.2 displays the clustering quality of each method using three indicators: homogeneity, completeness, and V-measure across the four datasets.

Clustering Quality. Table 5.2 and Figure 5.11 reveal low completeness and homogeneity homogeneity scores in the baselines, reflecting the struggle of standard clustering methods in identifying non-linearly separable clusters due to their parameter constraints and similarity calculations. These algorithms primarily cluster segments based on proximity, often resulting in similar but distant segments forming multiple

Table 5.2: The clustering quality results comparing the SPTCLUST approach with baselines.

Clustering Method	Dataset	Homogeneity	Completeness	V-measure
Kmedoids	Ferry(Halifax)	0.000	0.000	0.000
	Cargo(Halifax)	0.033	0.000	0.000
	Cargoes(Gulf of Mexico)	0.030	0.000	0.000
	Tankers(Gulf of Mexico)	0.021	0.000	0.000
DBSCAN	Ferry(Halifax)	0.0833	0.000	0.000
	Cargo(Halifax)	0.025	0.000	0.000
	Cargoes(Gulf of Mexico)	0.127	0.000	0.000
	Tankers(Gulf of Mexico)	0.122	0.000	0.000
OPTICS	Ferry(Halifax)	0.000	0.000	0.000
	Cargo(Halifax)	0.020	0.000	0.000
	Cargoes(Gulf of Mexico)	0.019	0.000	0.000
	Tankers(Gulf of Mexico)	0.012	0.000	0.000
BIRCH	Ferry(Halifax)	0.167	0.000	0.000
	Cargo(Halifax)	0.050	0.000	0.000
	Cargoes(Gulf of Mexico)	0.128	0.000	0.000
	Tankers(Gulf of Mexico)	0.073	0.000	0.000
SPTCLUST	Ferry(Halifax)	0.916	0.946	0.929
	Cargo(Halifax)	0.929	0.951	0.938
	Cargoes(Gulf of Mexico)	0.899	0.9398	0.9197
	Tankers(Gulf of Mexico)	0.883	0.941	0.907

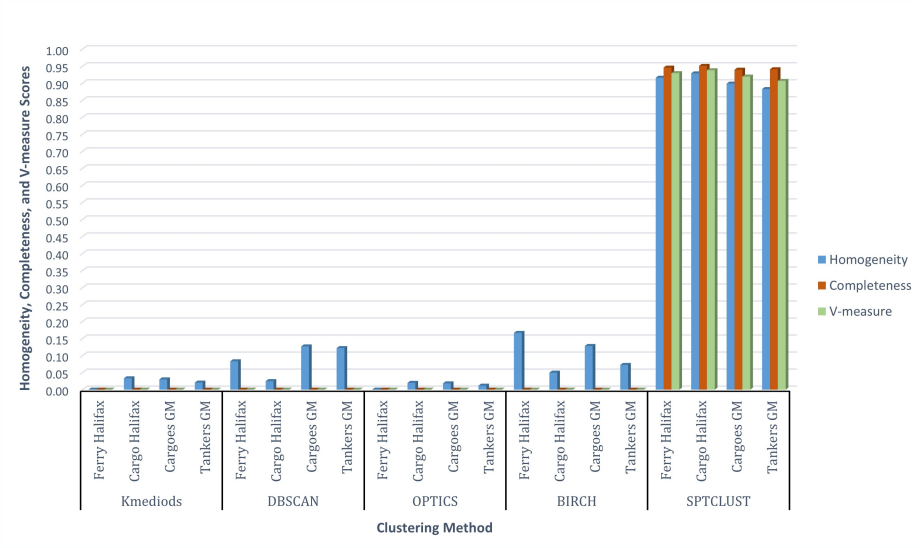


Figure 5.11: Visualization of Clustering Quality Results: SPTCLUST vs. Baselines across Four Datasets.

unnecessary clusters, i.e., low completeness, instead of being appropriately grouped in one cluster. Moreover, considering the proximity and the segments' shape results in segments with similar patterns but moving in opposite directions between two ports being clustered together, i.e., low homogeneity. In contrast, the SPTCLUST approach outperforms baseline methods in clustering performance, achieving higher scores in homogeneity, completeness, and V-measure across all four datasets. By clustering

segments solely based on the similarity of their endpoints, SPTCLUST effectively captures both positional and directional similarities. This results in well-separated clusters that accurately reflect real-world traffic flow.

Computational Efficiency. To evaluate the efficiency of our clustering method, we compared SPTCLUST with four baseline methods across four datasets from different water areas with varying traffic densities. The results in Table 5.3 and Figure 5.12 show that SPTCLUST surpasses all other methods in computational efficiency across all datasets.

Table 5.3: Runtime comparison of clustering methods: SPTCLUST approach and baselines across four datasets.

Dataset	#Points	#Seg.	Size	Standard Clustering				Proposed
				Kmedoids	DBSCAN	OPTICS	BIRCH	SPTCLUST
Ferry(Halifax)	103162	4263	16.6MB	06:20:35	06:20:37	06:26:13	06:20:37	00:00:06
Cargo(Halifax)	38853	23	1.04MB	00:00:04	00:00:04	00:00:04	00:00:04	00:00:00:056
Cargoes(Gulf of Mexico)	218213	4847	35.3MB	11:12:00	11:12:02	11:31:14	11:14:03	00:00:07
Tankers(Gulf of Mexic)	229471	5750	46.9MB	14:39:31	14:39:34	15:09:45	14:40:20	00:00:09

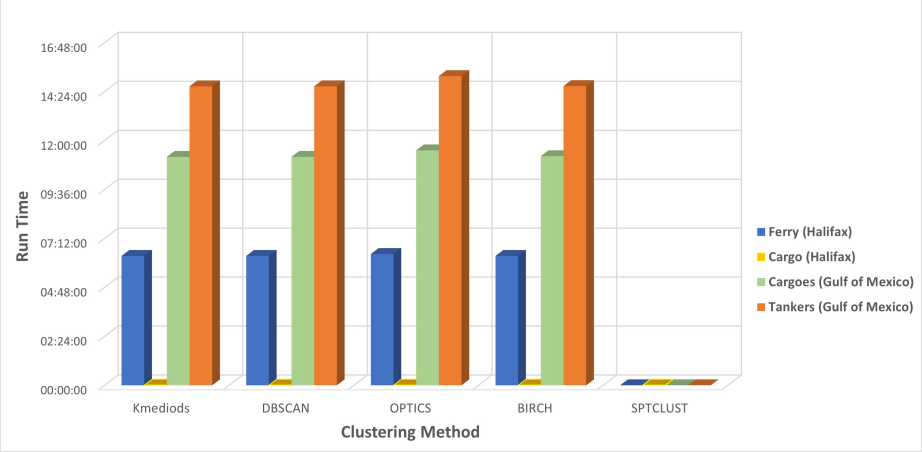


Figure 5.12: Visualization of the runtime comparison between SPTCLUST and baselines across four datasets.

This superiority can be attributed to SPTCLUST's reliance exclusively on the segments' endpoints' similarity, specifically considering the pair of bearings associated with these endpoints. This leads to a time complexity of $O(N)$ for clustering N segments. In contrast, Hausdorff distance for similarity measures have high time complexity, with at least $O(m^2)$ time cost for computing distances between segments with

m points each. This indicates that SPTCLUST achieves a lower computation complexity with a linear time complexity, making it effective in processing large datasets efficiently.

Impact of Clustering Parameters. Table 5.4 displays parameter values and resulting number of clusters.

Table 5.4: Clustering methods, their parameter values, and the number of generated clusters.

Clustering Method	Dataset	Parameters	Parameters values	#Clusters
Kmedoids	Ferry(Halifax)	k	6	6
	Cargo(Halifax)		6	6
	Cargoes(Gulf of Mexico)		265	265
	Tankers(Gulf of Mexico)		289	289
DBSCAN	Ferry(Halifax)	(ϵ, $MinPts$)	(0.01, 20)	12
	Cargo(Halifax)		(0.01, 3)	3
	Cargoes(Gulf of Mexico)		(5, 3)	252
	Tankers(Gulf of Mexico)		(5, 3)	254
OPTICS	Ferry(Halifax)	($min\_samples$, $min\_cluster\_size$, xi)	(3, 100, 0.01)	11
	Cargo(Halifax)		(2, 3, 0.01)	5
	Cargoes(Gulf of Mexico)		(3, 10, 0.01)	162
	Tankers(Gulf of Mexico)		(3, 10, 0.01)	170
BIRCH	Ferry(Halifax)	($branching\_factor$, $n\_clusters$, $threshold$)	(50, 12, 0.01)	12
	Cargo(Halifax)		(50, 6, 0.01)	6
	Cargoes(Gulf of Mexico)		(100, 265, 0.5)	265
	Tankers(Gulf of Mexico)		(50, 289, 0.5)	289
SPTCLUST	Ferry(Halifax)	No parameters		6
	Cargo(Halifax)			7
	Cargoes(Gulf of Mexico)			265
	Tankers(Gulf of Mexico)			289

Parameter tuning for baseline methods involves iterative adjustments to achieve optimal outcomes, using visualizations and metrics. Despite these tuning efforts, clusters generated by baseline methods still contain segments moving in opposite directions. Increasing the number of clusters tends to increase homogeneity but lower completeness. For example, assigning each segment to its own cluster maximizes homogeneity but minimizes completeness. When determining the number of clusters for DBSCAN, OPTICS, and BIRCH on the Halifax ferry dataset, we observed that these methods tend to classify a substantial portion of the data as noise. Nevertheless, we find that this number of clusters adequately preserves the fundamental trajectory structure. In contrast, SPTCLUST eliminates the need for input parameters, automatically determining both the number of clusters and their shapes.

Clustering Granularity-Interpretability Trade-off. Table 5.5 demonstrates that SPTCLUST surpasses baselines by autonomously generating clusters that define traffic flow between port pairs. This removes subjective bias and trial and error,

yielding quicker, more objective outcomes. Conversely, parameter tuning in baselines reveals a trade-off between granularity and interpretability. Granularity in clustering pertains to the level of detail in segments' partitioning, with higher granularity indicating more clusters, each with smaller, more homogeneous subsets. Increasing cluster numbers may augment granularity but might also introduce noise or less meaningful clusters. Conversely, reducing clusters could obscure distinctions between different data groups, impacting both cluster detail and interpretability.

Table 5.5: The influence of input parameters on clustering results.

Clustering Method	**Parameters**	**Clustering Outcomes**
Kmediods	k	Increasing (k) generally leads to more clusters, resulting in smaller cluster sizes as segments are partitioned into more groups. While decreasing k results in fewer clusters, resulting in larger cluster sizes as more segments are grouped together.
DBSCAN	(ϵ, $MinPts$)	Increasing (ϵ) decreases the compactness of clusters, as more neighbouring segments can be included in the same cluster despite being less related in direction. Increasing ($MinPts$) typically results in larger and fewer clusters.
OPTICS	($min\_samples$, $min\_cluster\_size$, xi)	Increasing the value of ($min\_samples$) reduces the number of core segments, resulting in larger cluster sizes and fewer clusters. Similarly, raising ($min\_cluster\_size$) typically leads to fewer clusters, as smaller clusters may fail to meet the minimum size requirement and merge into larger ones. Higher values of (xi) can also decrease the number of clusters by permitting greater density fluctuations, potentially merging smaller clusters into larger ones.
BIRCH	($branching\_factor$, $n\_clusters$, $threshold$)	Increasing the ($branching\_factor$) tends to increase the number of clusters while reducing their sizes, as it allows for more branching in the tree structure. Conversely, a higher number of clusters parameter ($n\_clusters$) generally leads to clusters of smaller sizes and more of them. Raising the ($threshold$) can decrease the number of clusters by allowing for greater similarity between clusters, potentially merging smaller clusters into larger ones.
SPTCLUST	-	Automatically clusters trajectory segments between port pairs, identifying the number of clusters and defining the traffic flow between them.

SPTCLUS' Second Clustering Step. Here, we evaluate the effectiveness of the second clustering step in partitioning clusters to potentially yield multiple reference routes within the same network lane. Figure 5.13j displays varying numbers of reference routes among different clusters, while Figures 5.15j and 5.16j display a single reference route per cluster. These routes serve as summarized representations of their respective clusters. Our analysis reveals that the second clustering step performs well with shorter trajectory segments from AIS data captured at the Halifax port (Figure 5.13j). However, it struggles with longer segments displaying complex motions from Gulf of Mexico datasets (Figures 5.15j and 5.16j). This challenge arises from the algorithm's reliance on spatial distribution variability among trajectory segments, utilizing pooled standard deviation, which oversimplifies relationships among longer segments and fails to capture nuanced differences in their patterns.

The Construction of Continuous and Smooth Lanes of Traffic Network. Here, we evaluate the efficacy of constructing complete summarized maritime traffic networks by averaging trajectory segments within clusters generated by each clustering method. This process reduces trajectory data complexity, aiding in identifying common traffic routes, i.e., reference routes, and capturing overall vessel movement trends between specific origin and destination pairs. High-quality clusters accurately represent real-world traffic flow, resulting in continuous and smooth reference routes between ports, thereby forming a comprehensive summarized maritime traffic network that preserves trajectory fundamental shape. Conversely, low-quality clusters yield an incomplete summarized maritime traffic network, missing the fundamental shape of the trajectory. Figures 5.13, 5.14, 5.15, 5.16 depict the results of clustering algorithms, with clusters of trajectory segments shown on the left and their corresponding reference routes forming summarized maritime traffic networks on the right. Baseline methods (K-medoids, DBSCAN, OPTICS, and BIRCH), as shown in the results table 5.2, produce low-quality clusters, resulting in incomplete summarized maritime traffic networks (on the right) that fail to capture trajectories fundamental shapes. However, SPTCLUST outperforms baselines in generating high-quality clusters, as highlighted in the results table 5.2, resulting in continuous and smooth reference routes between ports, thereby constructing a comprehensive summarized maritime traffic network that preserves trajectories fundamental shapes.

5.10 Limitations

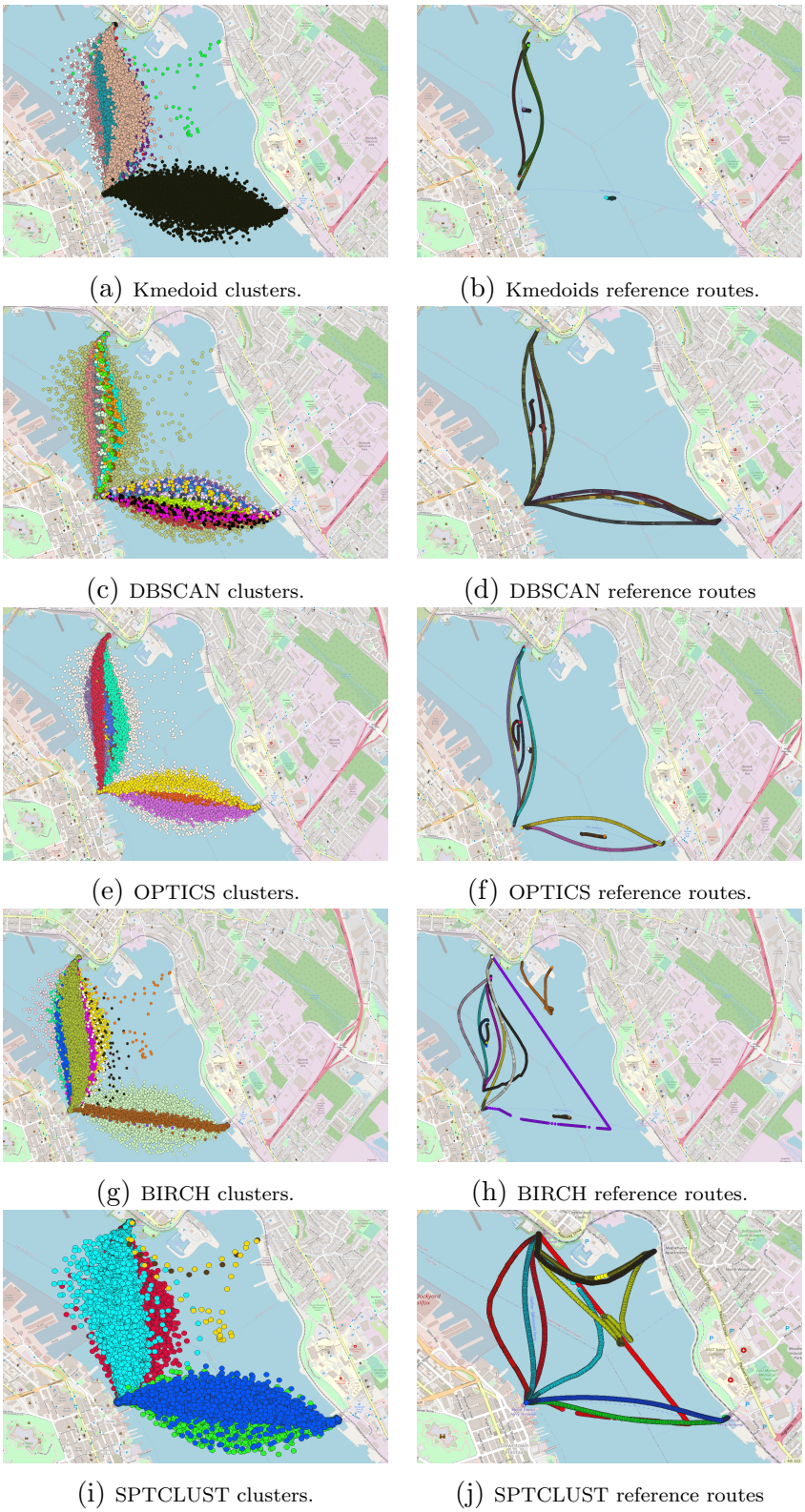
The primary limitation of our proposed semi-supervised clustering approach lies in the second clustering step. This step relies on statistical measures, which are uncertain measurements of hidden noise and complex patterns. Employing a more robust similarity measure in this step would be advantageous. This adjustment would enhance the identification and filtering of non-normal segments, thereby facilitating the recognition of typical and expected routes, maintaining data quality, and improving the accuracy of traffic pattern prediction.

Another limitation is highlighted with red circles in Figures 5.15j and 5.16j, where segments separated by unsailable area should yield two reference routes. However, the extracted reference routes show only one lane crossing the prohibited sailing area,

which diminishes the quality of the summarized maritime traffic network and poses a safety risk. To construct a reliable and safe map of maritime routes within the traffic network, it is essential to account for environmental obstacles such as islands and Marine Protected Areas (MPAs).

5.11 Conclusions

In this chapter, we introduce an advanced method for constructing maritime traffic network, utilizing a novel two-step clustering technique to capture potential multiple movement patterns for the same shipping lane. Trajectory segmentation based on port points (origins and destinations) and clustering of similar trajectory segments allows identifying established maritime routes, vessel paths, and port-to-port traffic flows directly from AIS trajectories. Our proposed semi-supervised clustering method to cluster trajectory segments can identify non-linearly separable clusters with irregular shapes and varied densities, outperforming four traditional methods by efficiently identifying complex vessel motion patterns in linear time without sacrificing accuracy. Additionally, it automatically determines the number of clusters without relying on random initialization and is not sensitive to outliers, providing more interpretable results. Our experiments demonstrate the efficacy of the proposed clustering approach in automatically detecting maritime routes that accurately represent real-world traffic flow. This facilitates the creation of a summarized traffic network that preserves the essential shape of vessels' trajectories while potentially reducing complexity. This allows focusing on the fundamental spatial patterns without getting bogged down in noise or minor deviations. The method performs well across four datasets, including dense and challenging water areas, demonstrating its competitive performance. Preserving the fundamental shape of the trajectory is crucial for accurate representation of vessel movement patterns in maritime operations. These representations serve as reliable inputs for predictive models and algorithms, enabling more accurate forecasts of future vessel movements and enhancing operational effectiveness and safety.



(a) Kmedoid clusters. (b) Kmedoids reference routes.

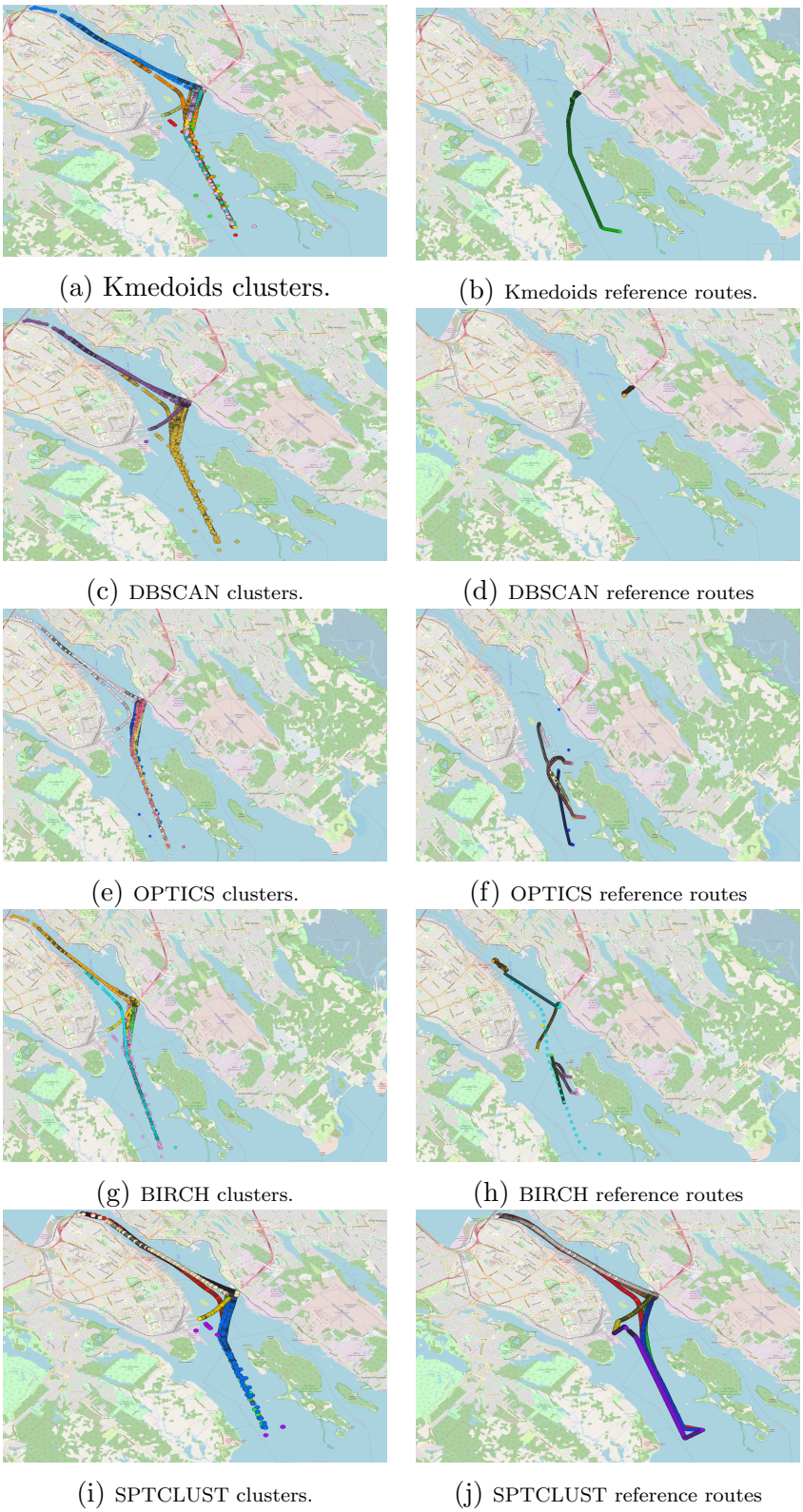
(c) DBSCAN clusters. (d) DBSCAN reference routes

(e) OPTICS clusters. (f) OPTICS reference routes.

(g) BIRCH clusters. (h) BIRCH reference routes.

(i) SPTCLUST clusters. (j) SPTCLUST reference routes

Figure 5.13: Visualization comparing SPTCLUST clusters and reference routes with those of Kmedoids, DBSCAN, OPTICS, and BIRCH, for Transit Ferry trajectory data from Halifax port.



(a) Kmedoids clusters.

(b) Kmedoids reference routes.

(c) DBSCAN clusters.

(d) DBSCAN reference routes

(e) OPTICS clusters.

(f) OPTICS reference routes

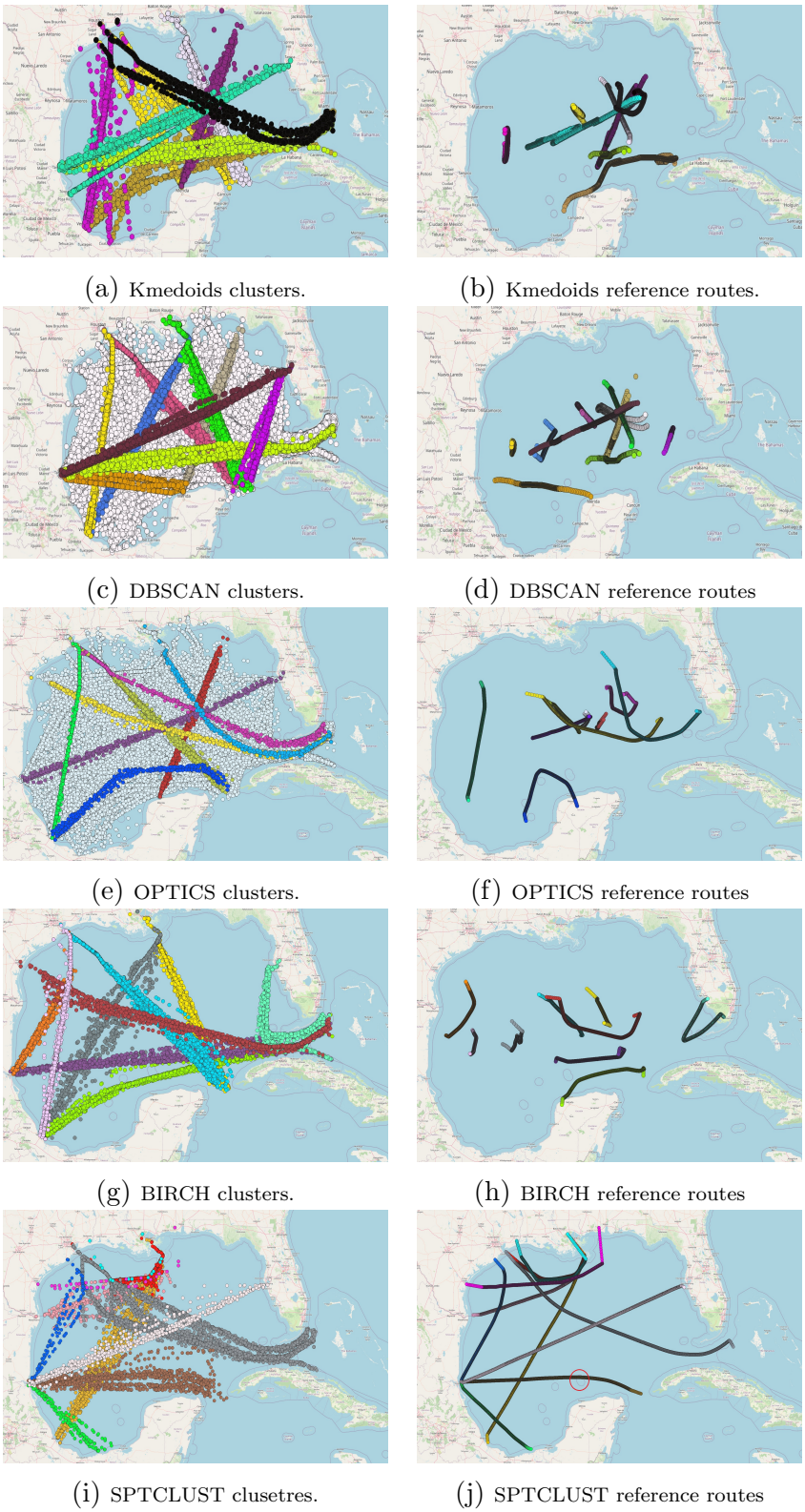
(g) BIRCH clusters.

(h) BIRCH reference routes

(i) SPTCLUST clusters.

(j) SPTCLUST reference routes

Figure 5.14: Visualization comparing SPTCLUST clusters and reference routes with those of Kmedoids, DBSCAN, OPTICS, and BIRCH, for Cargo Vessel trajectory data from Halifax port.



(a) Kmedoids clusters. (b) Kmedoids reference routes.

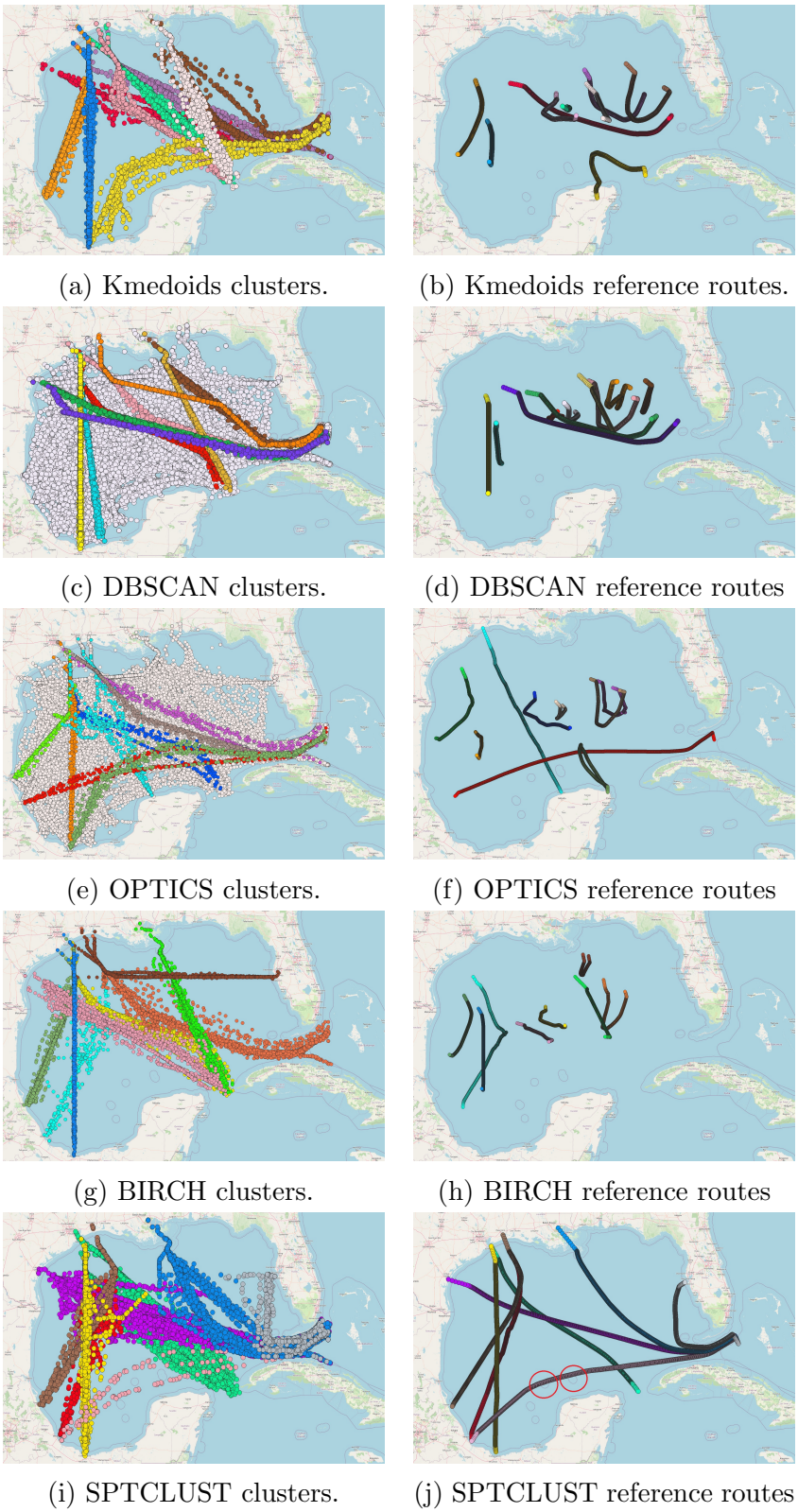
(c) DBSCAN clusters. (d) DBSCAN reference routes

(e) OPTICS clusters. (f) OPTICS reference routes

(g) BIRCH clusters. (h) BIRCH reference routes

(i) SPTCLUST clusetres. (j) SPTCLUST reference routes

Figure 5.15: Visualization comparing SPTCLUST clusters and reference routes with those of Kmedoids, DBSCAN, OPTICS, and BIRCH, for cargo vessels from the Gulf of Mexico basin.



(a) Kmedoids clusters. (b) Kmedoids reference routes.

(c) DBSCAN clusters. (d) DBSCAN reference routes

(e) OPTICS clusters. (f) OPTICS reference routes

(g) BIRCH clusters. (h) BIRCH reference routes

(i) SPTCLUST clusters. (j) SPTCLUST reference routes

Figure 5.16: Visualization comparing SPTCLUST clusters and reference routes with those of Kmedoids, DBSCAN, OPTICS, and BIRCH, for tanker vessels from the Gulf of Mexico basin.

Chapter 6

Enhancing the SPTCLUST Approach to Optimize Port Destination Predictions

Incorporating normal movement patterns improves the accuracy of vessel destination predictions. Aligning predicted routes with typical vessel movements reduces the likelihood of making predictions based on exceptional instances [51]. Extracting these normal patterns entails identifying and filtering trajectory segments that deviate significantly from expected norms. Current outlier detection methods, relying on fixed thresholds, often overlook variations in different sailing scenarios. To bridge this gap[1], we propose enhancing the second clustering step of the SPTCLUST approach by integrating an adaptive threshold algorithm to detect and filter outlier segments. This algorithm provides user assistance in selecting threshold values tailored to filter segments within each cluster, accommodating changes in motion states across various sailing scenarios. Consequently, this refinement leads to cleaner and more accurate representations of historical routes. Additionally, we propose enhancing the summarization algorithm to prevent aggregated routes from crossing prohibited sailing areas by incorporating Marine Protected Areas' (MPAs) and islands' centerpoints as intermediary points, resulting in two distinct reference routes avoiding prohibited sailing areas. Finally, we explore trajectory prediction tasks leveraging a detailed traffic network with clusters of maritime routes and a summarized maritime traffic networks to evaluate their efficacy in destination prediction.

6.1 Introduction

Normal movement patterns are crucial for predicting vessel trajectories accurately. In the literature, various outlier detection algorithms have been proposed to identify normal trajectory segments [88, 100]. However, a major limitation of these methods is

[1]This chapter is based on the publications [29, 31]

their reliance on constant threshold values, potentially limiting their ability to assign distinct outlier scores, given the diverse nature of vessel movements. This lack of adaptability may hinder the algorithms' effectiveness across different vessel behaviours and scenarios. To address this constraint, we propose an alternative approach using adaptive thresholding techniques to enhance the robustness of identifying normal trajectory segments [29]. Initially, the method automatically selects a representative segments within each cluster, capturing the typical movement patterns of vessels of a specific type. Then, Dynamic Time Warping (DTW) [13, 42] is utilized to evaluate the similarity between the chosen representative segment and other segments within the same cluster. The resulting similarity values are visualized through a histogram, allowing users to set an appropriate threshold value for each cluster to filter its outliers. Additionally, we propose utilizing knowledge of marine protected areas (MPAs) and islands to summarize normal segments within each cluster. This helps refine the representation of the generated reference routes by accurately accounting for deviations caused by non-sail areas. As a result, two reference routes, one on each side of the prohibited sailing area, are extracted, providing a more realistic depiction of the summarized maritime traffic network.

Our main contributions are: (1) We propose improving the second step of our SPTCLUST approach (Chapter 5) by developing an adaptive thresholding technique that takes into account the local context of each cluster to identify and filter outlier segments. (2) We propose considering marine protected areas and islands in maritime traffic network summarization to enhance safety and the overall precision of the summarized routes. (3) We propose exploring vessel destination prediction by leveraging detailed traffic networks with clustered trajectory segments along their edges, as well as summarized maritime traffic networks, to evaluate their efficacy in predicting vessel destinations based on recent movements.

6.2 Definitions

Definition 6.2.1 (Fragment of a Trajectory Segment)**.** A fragment of a trajectory segment, fr, can be defined as follows: Let $s = \{(l_1, t_1), (l_2, t_2), \ldots, (l_n, t_n)\}$ denote a trajectory segment, where l_i represents the location at time t_i. A fragment fr of the trajectory segment s is a subsequence $\{(l_j, t_j), (l_{j+1}, t_{j+1}), \ldots, (l_k, t_k)\}$, where j is the

index of the first location in the segment and k is the index of the last location before reaching the destination port. Thus, a fragment starts from the origin port but does not extend to the destination port, capturing a portion of the trajectory segment.

6.3 Normal Routes Extraction

In Chapter 5, when using the pooled standard deviation as a similarity measure to distinguish movement patterns within a cluster, we encountered difficulty discerning patterns in longer trajectory segments (definition 4.2.3). This limitation hindered SPTCLUST's effectiveness in identifying outlier segments. Therefore, we propose replacing the pooled standard deviation with dynamic time warping (DTW) to enhance the identification and filtering of outlier segments.

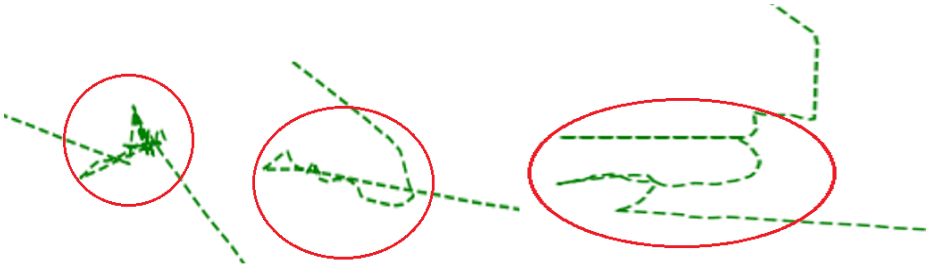
The DTW measure is chosen for its parameter-free nature, tolerance to noise and outliers inherent in motion patterns, and popularity in pattern matching tasks (Section 6.3.1). To capture normal routes, a representative segment within a cluster is selected (Section 6.3.2). Following that, the similarity among segments within the cluster is calculated using Dynamic Time Warping (DTW) to eliminate dissimilar segments (Section 6.3.2).

6.3.1 Similarity Measures

DTW, an unbounded similarity measure, assigns a value of 0 for identical segments, while larger values indicate greater dissimilarity [13, 87]. It measures similarity by identifying the optimal global alignment between two segments (definition 4.2.3) and exploring all their points' alignments to find the minimum distance [13, 87]. DTW offers advantages: It's parameter-free and robust to outlier observations [51]. However, a drawback of warping-based distance is its one-to-one comparison of trajectory points (definition 4.2.1). Thus, selecting a reference segment representing the normal motion trend is necessary to effectively measure the similarity of trajectory segments within each cluster and detect outliers. The following subsection describes the method for selecting such a reference segment.

6.3.2 Detecting and Filtering Outlier Segments

We identify two primary anomalous motion patterns among outlier segments: "sharp turning" and "self-crossing"[88]. Our trajectory segments contain these outliers, as shown in Figure 6.1.



(a) Self-crossing (b) Sharp turning

Figure 6.1: Example of anomalous movement patterns in trajectory segments.

The outlier segment detection process comprises two main steps: (1) the selection of the normal pattern, often referred to as the *"representative segment"*, and (2) the subsequent classification and filtering.

1. In Figure 6.1, anomalous movement patterns appear as sharp peaks within a time series. To select a *representative segment* within each cluster, each segment's spatial information is converted into a one-dimensional signal. This conversion is critical because the peak identification function only works with 1D arrays. The segment with the fewest or no peaks is then chosen to represent the normal movement pattern. This process is illustrated in Figure 6.2.

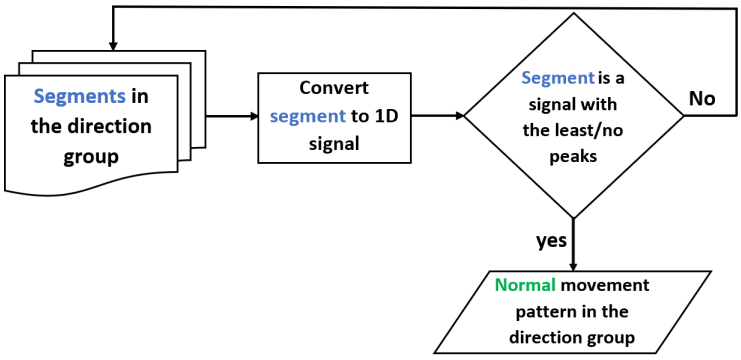


Figure 6.2: Normal pattern selection process.

2. To filter outlier segments, a similarity threshold ϵ is required, indicating the minimum desired similarity between segments and their representative segment in the same cluster. Segments with similarity scores exceeding this threshold are filtered out. The selection of ϵ is highly data-dependent, so the DTW similarity between trajectory segments and their representative segment is calculated, as detailed in Algorithm 7. The results, shown in a histogram and a list of similarity scores (Figure 6.4), assist users in determining the appropriate threshold. This calculation is performed for each cluster, considering distinct movement patterns between pairs of ports. Once the threshold is identified, the procedure outlined in Figure 6.3 is implemented to remove outlier segments.

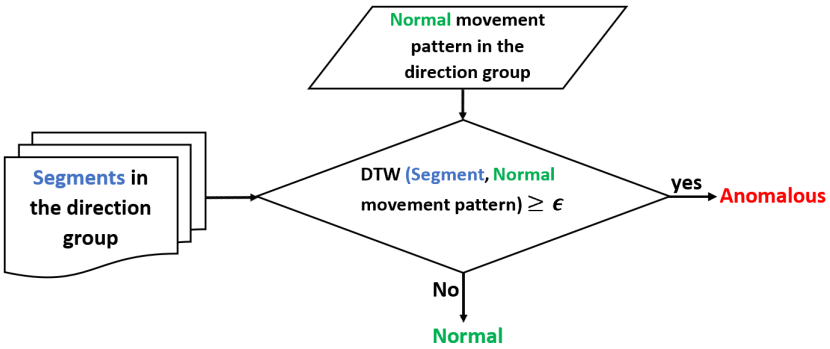


Figure 6.3: Segments classification process.

Algorithm 7: Choose epsilon

Input: A set of trajectory segments following same direction
$cluster_i = \{s_1, s_2, ..., s_k\}$

Result: A list of similarity values between the segments
$DTW\_values = \{DTW_1, DTW_2, ..., DTW_{k-1}\}\}$

```
DTW_values ← {} // list of variability values to choose ϵ
for each segment s_i in cluster_i do
    P ← (s_normal.latitude, s_normal.longitude);
    Q ← (s_i.latitude, s_i.longitude);
    DTW_values.add(DTW(P, Q));
end
print(sort DTW_values ascending);
print(sort DTW_values descending);
plot Histogram of DTW_values;
```

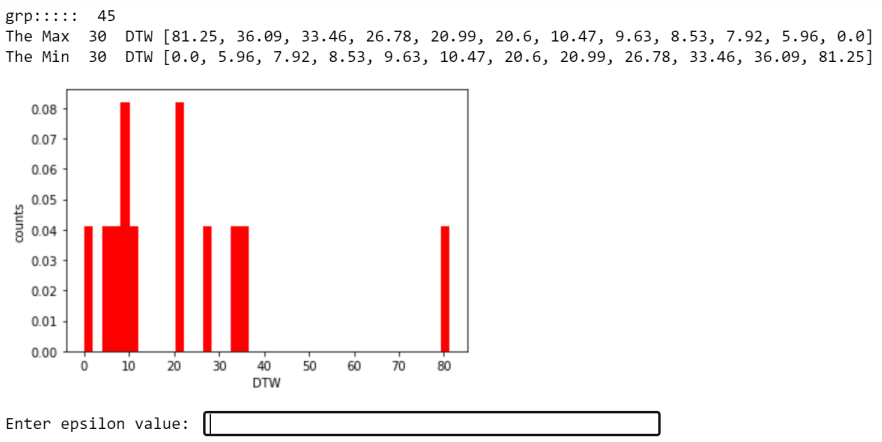


Figure 6.4: The ordered values of the similarity measure and the histogram represent variations in similarity between trajectory segments in a direction group.

6.4 Constructing Continuous and Smooth Summarized Traffic Network

To establish a complete summarized maritime traffic network with continuous and smooth aggregated lanes, a reference route for each cluster needs to be generated using the Reference Routes of Trajectory (RRoT) algorithm, discussed in Section 5.7. However, RRoT has a limitation: it can only extract a single reference route for segments within a cluster. In scenarios where obstacles like islands or Marine Protected Areas (MPAs) separate trajectory segments with identical origin and destination points, the aggregated route may traverse the restricted area. This limitation could be addressed by introducing intermediary points between two ports to highlight the distinctions between routes. To enhance the construction of the reference routes, a method called "Island_search" is developed. This method employs a semantic layer of islands and MPAs to separate trajectory segments within clusters separated by islands or MPAs (as illustrated in Figure 6.5). By using this approach, we can identify the trajectory segments on either side of the islands, or MPAs. The steps involved in this method are:

First, identify the coordinates of the center of the vicinity surrounding islands or MPAs. Next, create a list (e.g., islands_list) of port pairs separated by islands or MPAs. **Then, for each cluster:**

1. If its segments have start and endpoints in the list (islands_list):

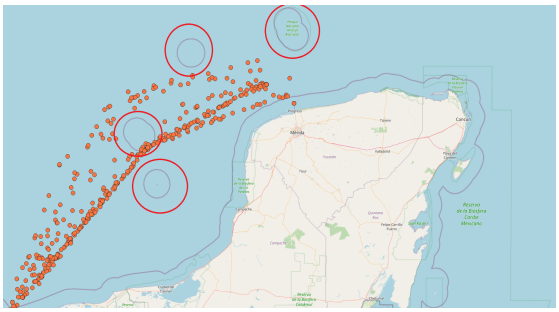


Figure 6.5: Trajectory segments following the same direction separated by MPAs, indicated by red circles.

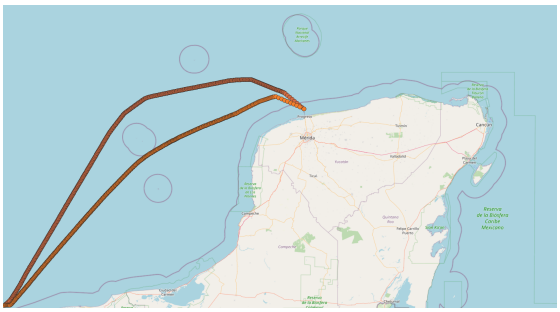


Figure 6.6: Reference route extraction of trajectory segments following the same direction separated by MPA.

(a) Group the segments based on their direction to the island's center point by calculating the compass bearing between these segments and the center points of the non-navigable areas (i.e., N, E, W, S). The standard compass bearing divides the compass into sixteen directions, spaced 11.25° apart.

(b) Each group should contain segments passing on one side of the non-navigable area.

(c) Construct the reference route for each group by utilizing the RROT algorithm 1, as shown in Figure 6.6.

2. Else if the cluster's segments have start and endpoints not in the list (islands_list): Construct the reference route for segments in the cluster using the RROT algorithm.

6.5 Vessel Destination Prediction

Maritime Situational Awareness (MSA) relies on leveraging extracted maritime knowledge, particularly maritime traffic networks inferred from trajectory patterns, to assist in vessel destination prediction based on recent movement patterns. Our proposed destination port prediction approach, detailed in Section 3.2 of Chapter 3, will be employed to evaluate how these representations can improve prediction accuracy. Detecting destination ports involves comparing the summarized traffic network with current trajectories, aiming to enhance computational efficiency. Meanwhile, destination prediction based on detailed traffic networks, with clusters of trajectory segments

along each connection, aims to facilitate the development of predictive models for accurately forecasting future ship locations up to the next port. Based on findings from Chapter 4, Discrete Fréchet Distance (DFD) and Dynamic Time Warping (DTW) are utilized for matching.

6.5.1 Destination Port Prediction based on Detailed Network

Clusters, representing normal movement patterns of maritime routes that accurately reflect real-world traffic flow, are expected to enhance the efficacy of a tracking method for predicting vessels' upcoming locations en route to their next destination. This method leverages clustered trajectory segments, aligning the recent route with historically clustered segments to forecast vessel movements based on similarity in direction and location, as illustrated in Figures 6.7 and 6.8. This classification technique employs two similarity measures; specifically, (i) Discrete Fréchet Distance, and (ii) Dynamic Time Warping (DTW), to align a test fragment of a trajectory segment with clusters of trajectory segments. A fragment of trajectory segment, representing a sequence of locations starting from the origin port but not extending to the destination port (e.g., an incomplete route).

Figure 6.7: Ship track prediction using utilizing clusters of maritime routes.

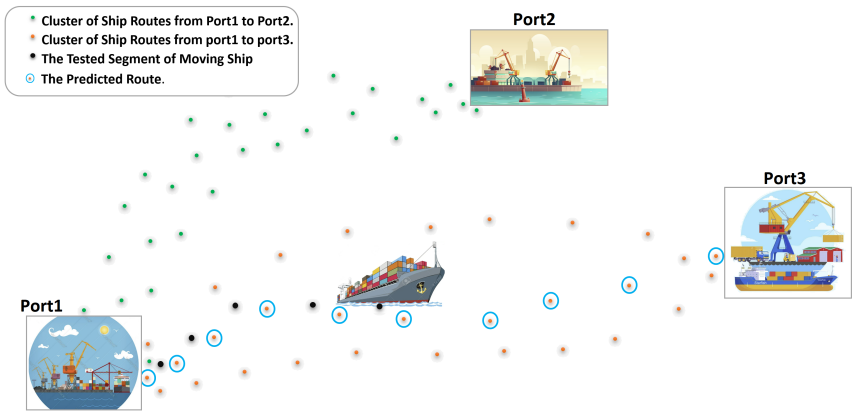


Figure 6.8: Prediction Process Overview.

6.5.2 Destination Port Prediction based on Summarized Network

The summarized traffic network, which summarizes trajectories, is anticipated to enhance the efficiency of destination port prediction. The aim is to predict vessel movements globally, taking into account the departure point, and the current ongoing route. Predicting a vessel's destination port entails identifying the reference route that closely aligns with the predicted ongoing route, as illustrated in Figure 6.9. This prediction technique matches reference routes with fragments of trajectory segments using two similarity measures: (i) Discrete Fréchet Distance (DFD), and (ii) Dynamic Time Warping (DTW).

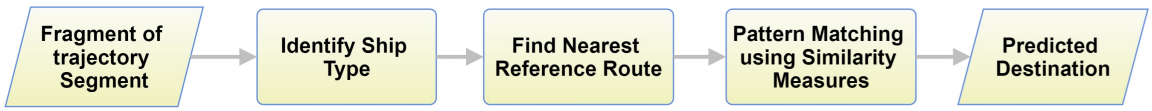


Figure 6.9: Ship destination prediction using maritime traffic network.

6.5.3 Evaluation Metrics for Prediction

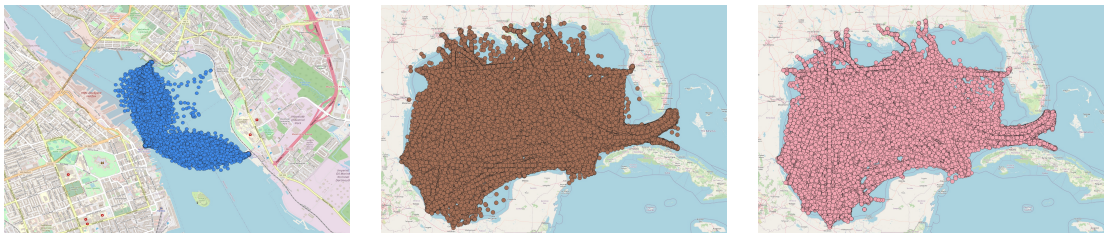
To evaluate the performance of our prediction models, we use accuracy, and the F1 score, which are two commonly used metrics for prediction accuracy. The scikit-learn metrics API are utilized to calculate these two metrics.

6.6 Experiments

This section evaluates the effectiveness and efficiency of the enhanced "SPTCLUST" clustering approach in extracting accurate representations of established maritime routes. Furthermore, it evaluates the performance of similarity-based prediction models that utilize these extracted representations to predict vessel routes and destinations.

6.6.1 Dataset

In our experiments, we utilize three AIS datasets captured in two distinct water regions, as illustrated in Figure 6.10. Specifically, we employ datasets from Halifax Harbour, including Transit Ferry, and two datasets from the Gulf of Mexico ocean basin, encompassing cargo vessels and tanker vessels AIS datasets.



(a) Transit ferry. (b) Cargo vessels. (c) Tanker vessels.

Figure 6.10: Visualization of Vessels' Trajectories for the three Datasets.

6.6.2 Filtering Outlier Segments

This subsection evaluates the removal of outlier segments from the clusters. It is a critical step, as these outliers can significantly affect the derived prediction models. Figure 6.11 compares randomly selected clusters of trajectory segments before and after filtering. These clusters include transit ferry data from Halifax port and trajectories of cargo vessels and tanker vessels from the Gulf of Mexico area.

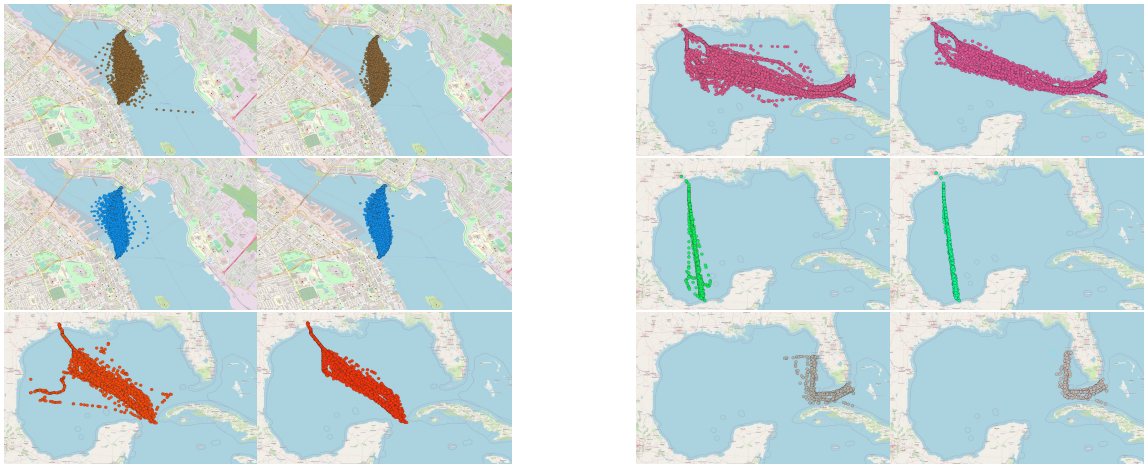


Figure 6.11: side by side Visualization of clusters of trajectory segments before and after outliers removal.

6.6.3 The Construction of Continuous and Smooth Summarized Lanes

Here we evaluate the construction of more precise reference routes, focusing on generating separate routes for segments within the same cluster, particularly those separated by non-navigable areas. Figure 6.12 illustrates 44 reference routes extracted

from cargo vessels' clusters, derived from 22 clusters with segments separated by non-navigable areas. The results highlight three instances where the extracted reference routes cross island borders, indicated by red circles. Additionally, Figure 6.13 shows 32 reference routes from tanker vessels' clusters, obtained from 16 clusters with segments separated by non-navigable areas, with two instances where the routes cross non-navigable areas, marked by red circles.

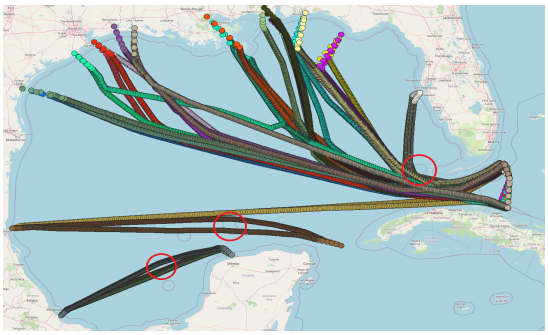


Figure 6.12: Cargo vessels' reference routes around obstacles.

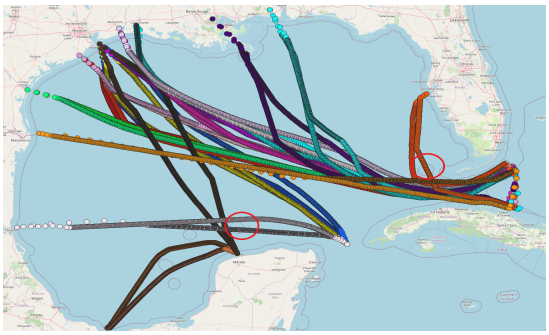


Figure 6.13: Tanker vessels' reference routes around obstacles.

Zoom in on the inaccurate cases depicted in the figures above and overlay them on their respective segments in the clusters. Figure 6.14 illustrates the cluster on the right (pink dots), containing segments separated by an MPA. The numbers of segments passing through the MPA's south and those in the north are mostly equal. As the segments approach their destination port, it becomes evident that the trajectory points are located on the MPA's border. Consequently, the resulting reference routes in Figure 6.14 are accurate and coincide with the original trajectory points. The same phenomenon occurs in the clusters on the left side of the picture.

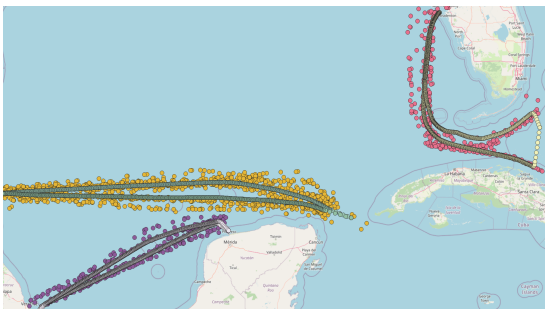


Figure 6.14: Cargo vessels reference routes atop their clustered segments.

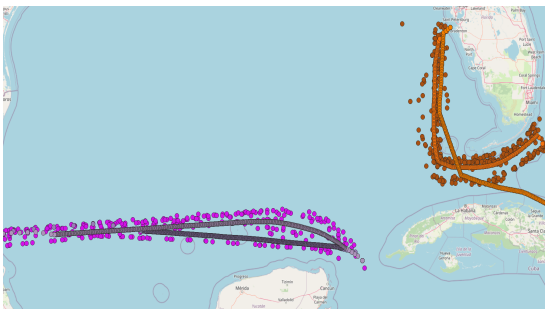


Figure 6.15: Tanker vessels reference routes atop their clustered segments.

Figure 6.15 shows the extracted reference routes overlaid on their respective clusters. In the cluster on the right, a reference route seems to cross the MPA, but the actual segment points don't pass through it. This deviation occurs because the arithmetic mean used in the RROT algorithm (algorithm 1) places the reference route away from the middle of the three segments due to data variability. In the cluster on the left, the reference route passing through the MPA's north aligns accurately with the segment points. However, the reference route passing the MPA's south is slightly shifted, again due to the use of the average in the RROT algorithm.

The evaluation of reference route extraction involves manual assessment through visualization and binary classification. Routes are categorized as either avoiding the prohibited sailing area (category 1) or passing through it (category 0). Accuracy and F1-score metrics evaluate the model's overall performance in constructing reference routes. Table 6.1 summarizes the accuracy and F1-scores for both cargo and tanker vessels' routes.

Table 6.1: Accuracy and F1-score of the extracted reference routes.

Vessel_Type	Accuracy	F1-score
Cargoes	98.1%	99%
Tankers	97%	98.5%

6.6.4 Destination Port Prediction based on Detailed Network

In this evaluation, we investigate predicting vessels' upcoming locations en route to their next destination, as detailed in section 6.5. The prediction accuracy is assessed using clusters of trajectory segments generated by our SPTCLUST approach and compared against clusters from four baseline methods: Kmedoids, DBSCAN, OPTICS, and BIRCH (see Section 3.2.2.1). To ensure robustness, we randomly sample 500 fragments and repeat this process five times, utilizing clusters from each method across three datasets. The averaged results from these five runs, including accuracy and F1-score metrics, are presented in Table 6.2 and Figure 6.16.

As depicted in Table 6.2 and Figure 6.16, predictive models utilizing clusters generated by SPTCLUST demonstrate higher prediction accuracy compared to those using clusters from standard methods. This superiority stems from SPTCLUST's

Table 6.2: Accuracy and F1-score, along with their 95% confidence intervals, are reported for two similarity-based predictive models: Discrete Fréchet Distance (DFD) and Dynamic Time Warping (DTW), using clusters.

Clustering Method	Dataset	DFD		DTW	
		Accuracy	F1-score	Accuracy	F1-score
Kmedoids	Ferry	50.50% ± 0.026	47.30% ± 0.023	47.05% ± 0.013	46.51% ± 0.010
	Cargoes	51.79% ± 0.026	48.03% ± 0.022	45.70% ± 0.013	45.30% ± 0.011
	Tankers	43.33% ± 0.013	45.61% ± 0.019	47.47% ± 0.019	46.10% ± 0.021
DBSCAN	Ferry	39.06% ± 0.020	36.54% ± 0.026	38.53% ± 0.025	36.94% ± 0.028
	Cargoes	39.08% ± 0.023	37.67% ± 0.023	41.56% ± 0.026	40.62% ± 0.030
	Tankers	40.23% ± 0.019	36.24% ± 0.015	38.45% ± 0.024	35.82% ± 0.022
OPTICS	Ferry	39.74% ± 0.011	37.60% ± 0.010	39.62% ± 0.012	37.71% ± 0.005
	Cargoes	37.82% ± 0.025	36.03% ± 0.028	37.02% ± 0.024	36.21% ± 0.023
	Tankers	29.04% ± 0.032	23.99% ± 0.022	29.62% ± 0.033	27.33% ± 0.023
BIRCH	Ferry	53.22% ± 0.037	49.19% ± 0.015	53.78% ± 0.017	49.88% ± 0.016
	Cargoes	54.17% ± 0.036	50.68% ± 0.037	53.81% ± 0.022	50.49% ± 0.025
	Tankers	45.39% ± 0.027	43.66% ± 0.020	46.28% ± 0.013	45.44% ± 0.010
SPTCLUST	Ferry	95.16% ± 0.014	94.54% ± 0.013	**97.28% ± 0.011**	**96.60% ± 0.013**
	Cargoes	76.14% ± 0.012	76.22% ± 0.014	**88.60% ± 0.012**	**87.81% ± 0.016**
	Tankers	69.96% ± 0.003	68.40% ± 0.005	**78.02% ± 0.012**	**77.75% ± 0.015**

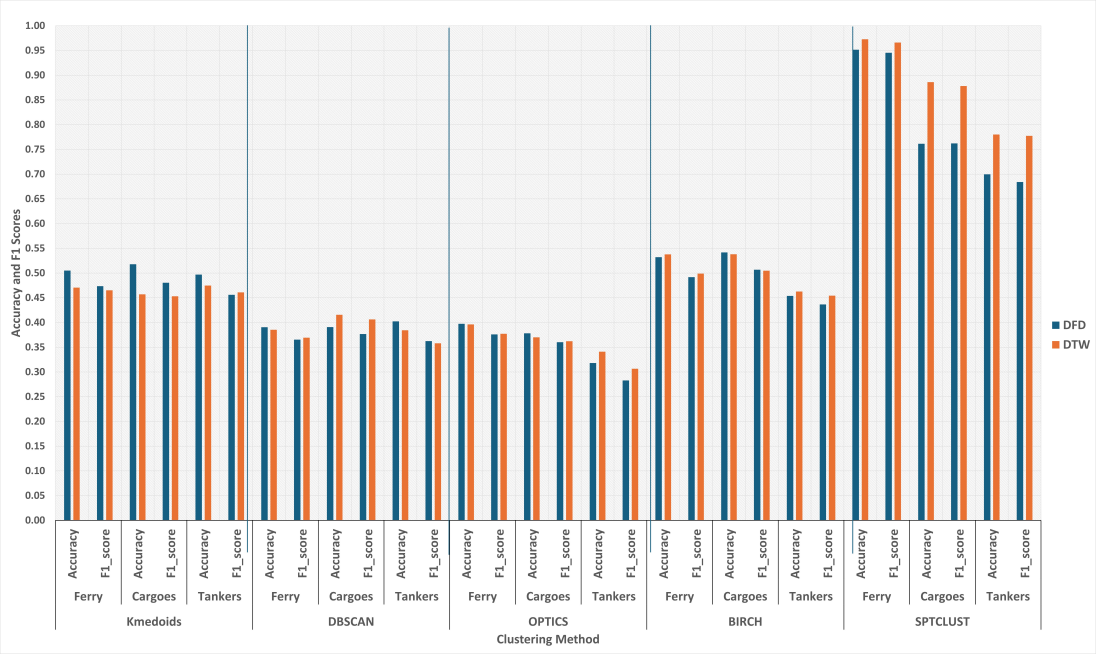


Figure 6.16: Visualization comparing prediction performance using SPTCLUST clusters versus baselines' clusters across three datasets.

capability to form comprehensive clusters, each containing all segments with consistent directions between two ports, as opposed to being scattered and mixed across multiple clusters. Consequently, this enables more precise matching with the ongoing trajectory.

However, the accuracy of the predictions relies on correctly classifying test fragments to their clusters. Overlapping clusters can complicate classification, leading to occasional misclassifications. Figure 6.17 illustrates this scenario, where accurate predictions are shown on the left side, while the right side shows erroneous predictions due to overlapping clusters. In the figure, the test fragment is highlighted in red, with its relevant cluster shown in lighter colors, and the ground truth is depicted in yellow. The output of the predictive model is represented by the most similar route in green, along with its cluster in lighter colors. It's noteworthy that predictions are continuously updated as vessels progress.

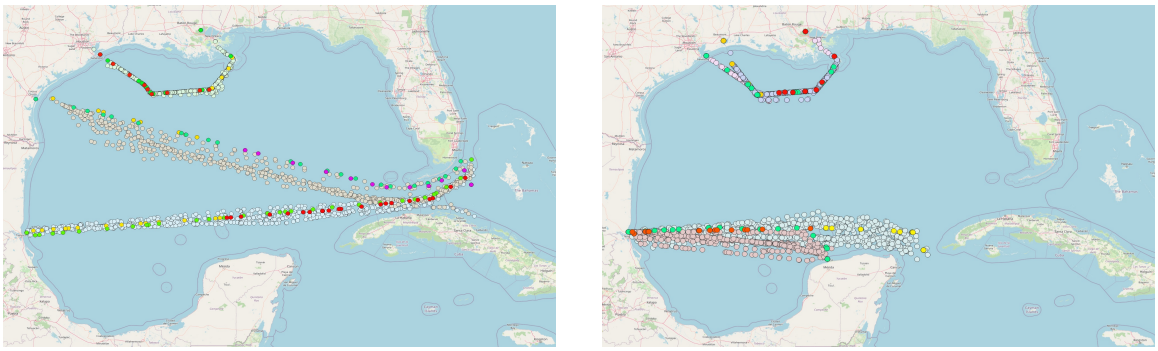


Figure 6.17: Prediction results using SPTCLUST-generated clusters: accurate predictions on the left; erroneous predictions on the right.

6.6.4.1 Destination Port Prediction based on Summarized Network

In this evaluation, we investigate the prediction of destination ports by comparing the test fragments with the generated reference routes, using two similarity measures: discrete Fréchet distance (DFD) and dynamic time warping (DTW), as elaborated in section 6.5. Prediction accuracy is evaluated using reference routes obtained from clusters generated by our SPTCLUST approach and compared with predictions using reference routes from clusters of four baseline methods: K-medoids, DBSCAN, OPTICS, and BIRCH. To ensure robustness, we randomly select 500 fragments and repeat the process five times, employing reference routes from each method across three datasets. Table 6.3 and Figure 6.18 present the averaged results from five runs, including accuracy and F1-score.

Table 6.3: Accuracy and F1-score with 95% confidence intervals for two similarity-based predictive models, Discrete Fréchet Distance (DFD) and Dynamic Time Warping (DTW), using reference routes.

Clustering Method	Dataset	DFD		DTW	
		Accuracy	F1-score	Accuracy	F1-score
Kmedoids	Ferry	0.0%	0.0%	0.0%	0.0%
	Cargoes	02.05% ± 0.007	02.07% ± 0.007	02.90% ± 0.010	03.05% ± 0.009
	Tankers	0.0%	0.0%	0.0%	0.0%
DBSCAN	Ferry	10.61% ± 0.013	10.32% ± 0.012	09.51% ± 0.006	09.65% ± 0.006
	Cargoes	09.52% ± 0.007	08.70% ± 0.006	08.79% ± 0.012	09.47 ± 0.013
	Tankers	07.11% ± 0.004	05.41% ± 0.005	04.80% ± 0.006	04.36% ± 0.003
OPTICS	Ferry	09.01% ± 0.004	08.10% ± 0.003	09.19% ± 0.002	09.56% ± 0.002
	Cargoes	0.0%	0.0%	0.0%	0.0%
	Tankers	00.57% ± 0.002	00.26% ± 0.001	00.57% ± 0.002	00.51% ± 0.001
BIRCH	Ferry	06.10% ± 0.016	05.11% ± 0.014	06.08% ± 0.010	05.58% ± 0.009
	Cargoes	05.53% ± 0.008	04.84% ± 0.006	03.24% ± 0.012	03.26% ± 0.010
	Tankers	00.47% ± 0.003	00.29% ± 0.002	00.47% ± 0.003	00.35% ± 0.003
SPTCLUST	Ferry	93.35% ± 0.015	92.84% ± 0.014	**95.18% ± 0.013**	**94.02% ± 0.011**
	Cargoes	73.14% ± 0.023	71.12% ± 0.019	**77.30% ± 0.014**	**76.68% ± 0.015**
	Tankers	**54.09% ± 0.032**	**51.53% ± 0.028**	51.14% ± 0.009	47.59% ± 0.011

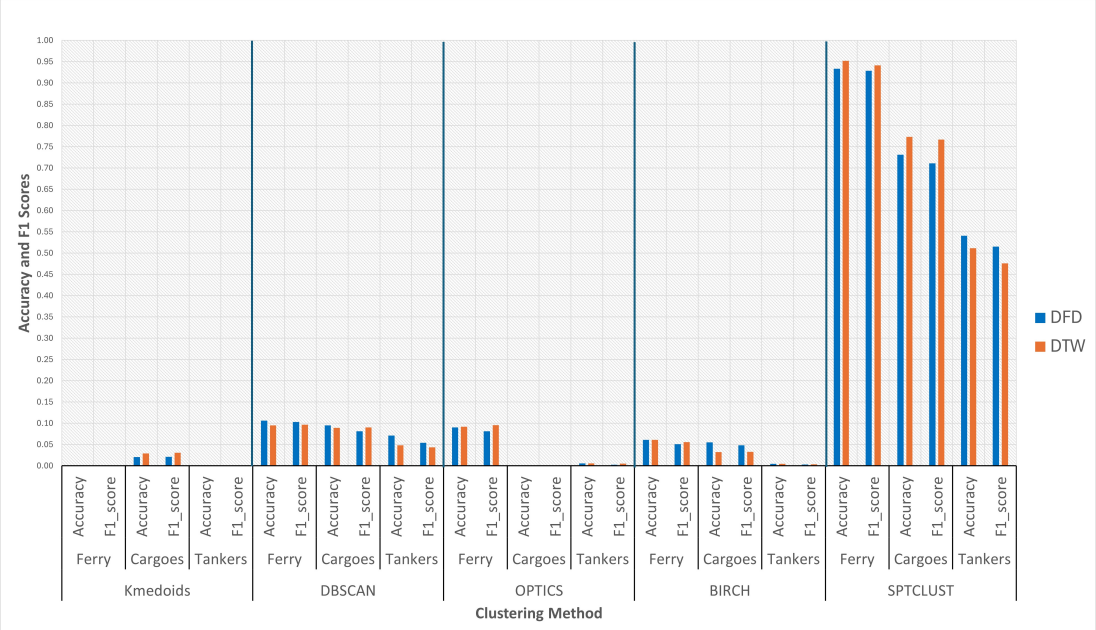


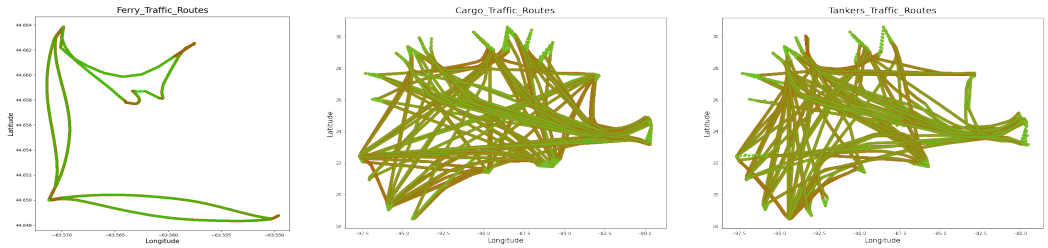
Figure 6.18: Visualization comparing prediction performance using SPTCLUST reference routes versus baselines' reference routes across three datasets.

As shown in Table 6.3 and Figure 6.18, models using reference routes from SPTCLUST-generated clusters perform better in predicting accuracy than those using routes from baselines' clusters. This is attributed to the following factors:

1. SPTCLUST-generated clusters group all segments following the same direction into a single cluster. Consequently, averaging segments with consistent directions between two ports results in comprehensive reference routes from port to port.

2. Clusters generated by baselines contain segments with conflicting directions within each cluster. Averaging segments with opposite directions between two ports leads to incomplete reference routes, which degrade the prediction.

However, when forecasting the destinations of fragments from tanker and cargo vessels' trajectories, we observe lower prediction accuracy than forecasting the destinations of fragments from transit ferry, as depicted in Figure 6.19a. This is mainly because each cluster of trajectory segments from the transit ferry has only one corresponding reference route outgoing from the ports for each direction. Moreover, instances where there are two reference routes outgoing from the same port are often distant from each other, facilitating their easy classification.



(a) Transit ferry. (b) Cargo vessels. (c) Tanker vessels.

Figure 6.19: Visualization of maritime traffic networks for the three datasets.

In contrast, the summarized traffic networks of tankers and cargo vessels depicted in Figures 6.19b and 6.19c exhibit overlapping reference routes, along with frequent instances of two reference routes for the same lane. This decreases the overall accuracy of the prediction model. The existence of multiple routes originating from the same starting point introduces ambiguity, posing a challenge for the model to determine the most appropriate match, consequently leading to incorrect predictions. Additionally, we noted that the predictive model using tanker vessels' trajectory data yielded the lowest results. This is attributed to the irregular sampling rate of this data. In general, destination predictions based on clusters exhibit greater precision compared to those based on maritime traffic networks.

6.7 Limitations

The limitation of our predictive approach is its inability to accurately forecast definitive destinations over long time horizons exceeding two hours. Specifically, the use of pattern matching may occasionally associate the ongoing route with an incorrect historical maritime route. In instances where predicted routes overlap and share similarities, this can result in inaccurate classification and diminish prediction performance, particularly when the vessel is far from its destination. Investigating advanced classification techniques that incorporate additional features and contextual information holds promise for improving the accuracy of destination prediction over longer time horizons.

6.8 Conclusions

By this chapter, we have presented novel AIS data mining approaches that overcome the limitations of traditional approaches, enabling a more comprehensive analysis of AIS datasets. These methods mitigate the reliance on the trajectory features themselves, thereby reducing potential gaps in rule generation. They also address the dynamic nature and complexity of AIS data and account for the non-independent nature of AIS data points, thus enhancing maritime situational awareness and intelligent system design. Initially, in this chapter, an adaptive threshold method is introduced to detect and filter outlier trajectory segments in the SPTCLUST clustering approach, thereby enhancing its versatility and practicality in AIS data analysis. Additionally, an enhancement to constructing the summarized maritime traffic network is proposed by incorporating intermediary points such as islands and Marine Protected Areas (MPAs) to segregate trajectory segments, ensuring that aggregated reference routes avoid crossing unsailable areas. To evaluate the effectiveness of our proposed methods in deriving representations of maritime routes that accurately reflect real-world traffic flow and enhance maritime situational awareness, the derived representations of maritime traffic networks are utilized with similarity measures to predict vessels' destinations based on their recent movement. This prediction approach involves aligning recent trajectory data of moving vessels with clusters of historical trajectory segments to predict vessels' upcoming locations en route to their

next destination. Experimental results demonstrate that prediction accuracy using clusters generated by our SPTCLUST approach outperforms prediction accuracy using clusters generated by four traditional clustering methods: K-medoids, DBSCAN, OPTICS, and BIRCH. Although using the summarized maritime traffic network for prediction instead of the detailed traffic network with clusters results in less accurate predictions, it offers greater computational efficiency.

Chapter 7

Conclusions and Future Work

This thesis presents a model for destination port prediction that leverages a newly developed maritime traffic route network structure to forecast port destinations for vessels based on their recent movements. The motivation for the design of this model is to overcome the limitations of destination prediction based on coarse-grained graph abstractions, which lack the granularity and precision of traffic patterns needed to accurately forecast vessel destinations in real-world scenarios.

A review of vessel destination prediction models and methodologies was conducted, including data preprocessing, data clustering for trajectory discretization, graph abstraction from discretized trajectories, and the application of machine learning and deep learning models for training and prediction. The key problems concerning vessel destination prediction using graph-based methods have been highlighted. The approach discretizes vessel trajectories into waypoints where vessels turn or stop but struggles with predicting non-linear dynamic trajectories and capturing full shipping route patterns, limiting real-world applicability. It relies on predefined or fixed route networks for extracting waypoints and boundaries, restricting the ability to identify meaningful patterns and traffic flows from trajectory data. Fixed route networks lack the flexibility to adapt to different regions and traffic densities. They may oversimplify or miss complex patterns, interdependencies, and the non-independent nature of AIS data points. Moreover, the key issue in abstracting a maritime traffic network into a graph is selecting a distance threshold that balances spatial relationship preservation and complexity reduction, critically influencing the graph's accuracy in representing true spatial relationships and connectivity. Considering these problems, we have developed a data-driven methodology for inferring the maritime traffic network topology directly from trajectory patterns. Our approach allows the trajectories to dictate the network structure rather than relying on fixed routes or distance

thresholds. Through trajectory segmentation based on port points (origins and destinations), we introduced a semi-supervised clustering algorithm named SPTCLUST, designed specifically for clustering trajectory segments between port pairs. This clustering approach effectively partitions maritime traffic into interpretable clusters, each representing the directional flow of maritime routes between two ports. By clustering motion patterns along each port-to-port connection, our methodology accurately identifies established maritime routes, vessel paths, and traffic flows from the AIS data. Consequently, we derive a directed graph representation, where nodes correspond to ports, and edges depict the inferred traffic routes. This adaptable network topology flexibly captures the traffic patterns observed in the data, accommodating the intricate dependencies and non-independent nature of AIS data points. Unlike predefined networks, our approach does not rely on fixed routes or distance thresholds, enabling it to adapt to different regions and traffic densities. The derived traffic network consists of two representations based on connection details: a summarized graph with aggregated trajectory segments as connections and a detailed graph with clusters of trajectory segments along connections. Using these network structures and similarity measures to predict vessel destinations, we have shown that this prediction approach captures the intricate patterns and interconnections of maritime traffic routes. By comparing a vessel's recent trajectory with this network representation, the model can infer the likelihood of different port destinations, accounting for maritime transportation systems' inherent complexities and dynamics. Experiments on four real-world AIS datasets from two areas with varying traffic densities support this finding.

This thesis uses an empirical approach and an analysis of network connections to find the most appropriate representation for accurate destination prediction.

7.1 Findings

1. Network Representation and Destination Prediction:

    - The proposed network representations, derived directly from AIS trajectory data, coupled with similarity measures, enable the prediction of vessel destinations based on recent trajectory information.

- This approach results in a probabilistic prediction framework that effectively expresses uncertainty through a probability distribution over potential port destinations.

2. Uncertainty Quantification and Decision Support:

   - As vessels are distant from their destination, multiple potential outcomes arise, and the model predicts a subset of all ports with the probability of arrival for each port.
   - As the vessel approaches its destination, the probabilities are continuously updated, capturing the evolving uncertainty.
   - Expressing predictions as probability distributions provides a more realistic and informative outlook than deterministic predictions.

3. Prediction Horizon and Limitations:

   - When the vessel is far from its destination, pattern matching may associate the current route with an inaccurate historical maritime route. This issue arises particularly when predicted routes overlap and exhibit similarities with incorrect historical routes, leading to misclassifications that impact prediction accuracy.
   - The issue of incorrectly associating a vessel's current route with an inaccurate historical maritime route when the vessel is far from its destination directly impacts the time horizon over which destination predictions can be made with high accuracy and confidence.
   - This limitation highlights the need for additional techniques or data sources to improve long-range destination prediction, especially in complex maritime environments with overlapping routes and high traffic densities.

4. Network Representation Comparison:

   - Utilizing the detailed traffic network, which includes clusters of trajectory segments along each connection, leads to more accurate destination predictions than using a summarized traffic network with aggregated trajectory segments as connections.

- The summarized traffic network offers greater computational efficiency, highlighting a trade-off between prediction accuracy and computational complexity.

These findings demonstrate the effectiveness of the proposed data-driven approach in predicting vessel destinations while quantifying uncertainty, as well as the trade-offs and limitations associated with different network representations and prediction horizons in complex maritime environments.

7.2 Future Research

This thesis is focused on developing a novel maritime traffic network representation, where the network connections represent the traffic flow and movement patterns exhibited by different vessels of the same type so that the vessel destination can be predicted based on its recent movement. Here, we discuss potential future research directions, which include:

1. **Port polygons generation.** To accurately identify a vessel's origin and destination points at ports, creating polygons that accurately represent the port areas or docks is crucial. These polygons enable us to determine whether a trajectory point falls within the designated port area. Currently, the process involves manually adjusting the radius of circles around each port location, which can be tedious and time-consuming. Therefore, exploring automated methods for identifying the circular areas around ports' locations would be beneficial. This would streamline the process and enhance efficiency in generating accurate port polygons.

2. **Reference routes construction.** Although averaging routes within each cluster provides satisfactory representations for calculating similarities, there is a need for improved accuracy in visualizations, specifically to prevent shifts of reference routes that traverse prohibited areas due to skewed spatial data distributions. Alternative methods, such as the geometric mean or harmonic mean, can be explored as substitutes for the arithmetic mean during the extraction of

reference routes to address this issue. These alternative methods have the potential to capture spatial information more accurately and can be investigated to enhance the precision and fidelity of the visualized routes.

3. **Fuse the vessels' trajectory data with some meteorological and oceanographic data.** By integrating trajectory data with meteorological and oceanographic data, the motion information of ships can be deeply mined, and important information hidden within the data can be effectively extracted. This increases the accuracy of destination prediction on a local and global scale, which can be used to perceive potential risks and ensure navigation efficiency. With reliable destination prediction, it becomes possible to make informed decisions that optimize the routes taken by ships, enhance safety measures, and minimize environmental impact. Ultimately, using these advanced data analytics techniques represents a significant step forward in marine transportation, offering a powerful tool for improving global shipping operations' efficiency and sustainability.

4. **Deep learning methods.** Deep learning algorithms offer a noteworthy advantage through their incremental learning of high-level features from data. This distinctive ability enables them to effectively capture the structural similarities inherent in complex movement patterns. As a result, these algorithms have the potential to identify highly reliable structural matches and accurately predict ships' destinations even for longer time horizons exceeding 24 hours. Moreover, it is crucial to integrate uncertainty quantification techniques into deep learning prediction models to express prediction uncertainty, thereby facilitating improved risk assessment and decision support for stakeholders. For instance, Monte Carlo simulation techniques sample input probability distributions and run computational models multiple times to quantify uncertainties in model outputs. Meanwhile, Polynomial Chaos Expansions (PCE) approximate model output uncertainty by expanding it into orthogonal polynomial basis functions of input random variables, offering an analytical alternative to sampling methods like Monte Carlo.

5. The proposed model for vessel destination prediction can be extended to **ETA research**, aiming to improve operational efficiency, safety, and supply chain visibility by providing reliable arrival time estimates for better planning and decision-making in port operations. The process involves leveraging diverse data sources, constructing vessel trajectories, calculating optimal routes, estimating travel times, and utilizing machine learning techniques to predict accurate ETAs while continuously updating predictions based on real-time data. After generating predictions by our model, these predictions can be enriched with an estimated time of arrival (ETA) by determining the optimal path for the vessel, considering factors like distance, weather, and traffic conditions, and estimating travel time based on vessel speed and other relevant factors. Subsequently, machine learning models (e.g., regression, ensemble methods) are trained on preprocessed data to learn patterns and make ETA predictions, with techniques like gradient boosting, random forests, and neural networks showing promising results. The performance of different machine learning models is evaluated using appropriate metrics (e.g., mean absolute error, root mean squared error), and the best-performer fleets' strategic positioning and their vessels' destination stated as new data becomes available, such as changes in vessel position, speed, and weather conditions.

6. Another potential avenue for future research is to apply the proposed model to **predict the destinations of tanker vessels**. Predicting tanker destinations extends beyond data points; it serves as a cornerstone for comprehending oil and gas flows, vessel availability, and global energy market dynamics. These predictions are crucial for clients, providing invaluable insights to navigate complexities with confidence and foresight. Companies engaged in oil transportation typically withhold information regarding the strategic positioning of their fleets and the destinations of their vessels on voyages. Consequently, situations often arise when oil transport vessels are oversupplied in certain regions while others face excessive demand. Committing a vessel to a voyage may thus result in a waste of time and resources, as competitors could reach the destination first and saturate the demand. Therefore, the ability to accurately forecast the future

destinations of competitors' oil tankers can confer a competitive advantage. Predicting the future positions of competitors' oil tankers can significantly enhance the strategic positioning of an oil tanker fleet, offering numerous competitive advantages. Using such forecasts, oil tankers would undertake fewer unprofitable journeys, optimizing resource allocation and reducing operating costs. This directly impacts an oil-shipping company's bottom line by lowering crew salaries and fuel consumption while potentially increasing profitability. Moreover, accurate forecasting benefits the oil supply chain by improving the overall distribution of oil tankers, thereby enhancing service for both oil providers and end-consumers.

7. Performing a larger scale evaluation of vessels' trajectories using Automatic Identification System (AIS) data over extended periods (e.g., 20–30 years) necessitates substantial computing power, such as High-Performance Computing (HPC) resources and big data analytics tools. Acquiring and storing massive amounts of AIS data would require robust data collection mechanisms, integration from various sources, and a scalable storage infrastructure. While longer time frames provide valuable insights, shorter durations (1-2 years) may be more practical for capturing recent patterns while mitigating the influence of significant changes. Various factors evolve over time, such as trading routes, operational ports, and technological advancements. For instance, world cruises typically traverse from Asia through the Red Sea and the Suez Canal. However, reaching the Suez Canal necessitates passing through the Bab Al Mandeb Strait near Yemen's coast. The ongoing conflict in the region, including incidents of Houthi rebels targeting vessels with cruise missiles, has prompted marine transportation companies to redirect their vessels. For example, some companies have recently updated their 2025 Grand World Voyage itineraries, replacing the Red Sea route with a circumnavigation of Africa to Europe. The spatial scale could be increased to transcontinental, influencing data coverage and computing requirements. Robust data quality assessment, cleaning, and preprocessing techniques are crucial for reliable analysis. Interdisciplinary collaboration among experts in maritime transportation, data science, HPC, and geospatial

analysis would be beneficial to tackle the challenges effectively. Careful planning, resource allocation, and collaboration are essential for the feasibility and impact of such a large-scale evaluation.

Bibliography

[1] Andrade-Campos A., De-Carvalho R., and Valente R. A. F. *Novel criteria for determination of material model parameters*. International Journal of Mechanical Sciences, 54(1):294– 305, 2012. ISSN 0020-7403, 2012.

[2] BADAR ABBAS. Spherical trigonometry and navigational calculations. https://badarabbas.wordpress.com/2015/01/30/spherical-trigonometry-and-navigational-calculations/#:~:text=Spherical%20trigonometry%20is%20used%20for,Inertial%20Navigation%20Systems%20(INS)., 2024.

[3] Yaseen Adnan Ahmed, Mohammed Abdul Hannan, Mahmoud Yasser Oraby, and Adi Maimun. Colregs compliant fuzzy-based collision avoidance system for multiple ship encounters. *Journal of Marine Science and Engineering*, 9(8), 2021.

[4] Alessia Albanese, Sankar K. Pal, and Alfredo Petrosino. Rough sets, kernel set, and spatiotemporal outlier detection. *IEEE Transactions on Knowledge and Data Engineering*, 26(1):194–207, 2014.

[5] Danial Alizadeh, Ali Asghar Alesheikh, and Mohammad Sharif. Prediction of vessels locations and maritime traffic using similarity measurement of trajectory. *Annals of GIS*, 27(2):151–162, 2021.

[6] Luis Alvares, Vania Bogorny, and Bart. Kuijpers. Towards semantic trajectory knowledge discovery. 10 2007.

[7] Marlene Alvarez, Virginia Fernandez Arguedas, Vincenzo Gammieri, Fabio Mazzarella, Michele Vespe, Giuseppe Aulicino, and Antonio Vollero. Ais event-based knowledge discovery for maritime situational awareness. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 1874–1880, 2016.

[8] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: Ordering points to identify the clustering structure. SIGMOD '99, page 49–60, New York, NY, USA, 1999. Association for Computing Machinery.

[9] Homayoon Arbabkhah, Atefe Sedaghat, Masood Jafari Kang, and Maryam Hamidi. Automatic identification system-based prediction of tanker and cargo estimated time of arrival in narrow waterways. *Journal of Marine Science and Engineering*, 12(2), 2024.

[10] Virginia Fernandez Arguedas, Fabio Mazzarella, and Michele Vespe. Spatio-temporal data mining for maritime situational awareness. In *OCEANS 2015 - Genova*, pages 1–8, 2015.

[11] Maike Buchin, Somayeh Dodge, and Bettina Speckmann. Similarity of trajectories taking into account geographic context. *Journal of Spatial Information Science*, 9, 11 2014.

[12] Maike Buchin, Somayeh Dodge, and Bettina Speckmann. Similarity of trajectories taking into account geographic context. *Journal of Spatial Information Science*, 9, 11 2014.

[13] Jekel C., Venter G., Venter M., Stander N., and Haftka R. *Similarity measures for identifying material parameters from hysteresis loops using inverse analysis.* International Journal of Material Forming, 2019.

[14] Jing Cao, Maohan Liang, Yan Li, Jinwei Chen, Huanhuan Li, Ryan Wen Liu, and Jingxian Liu. Pca-based hierarchical clustering of ais trajectories with automatic extraction of clusters. In *2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA)*, pages 448–452. IEEE, 2018.

[15] Samuele Capobianco, Leonardo M. Millefiori, Nicola Forti, Paolo Braca, and Peter Willett. Deep learning methods for vessel trajectory prediction based on recurrent neural networks. *IEEE Transactions on Aerospace and Electronic Systems*, 57(6):4329–4346, 2021.

[16] Emanuele Carlini, Vinicius Monteiro de Lira, Amílcar Soares Júnior, Mohammad Etemad, Bruno Brandoli Machado, and Stan Matwin. Uncovering vessel movement patterns from ais data with graph evolution analysis. In *EDBT/ICDT Workshops*, 2020.

[17] Xinqiang Chen, Jun Ling, Yongsheng Yang, Hailin Zheng, Pengwen Xiong, Octavian Postolache, and Yong Xiong. Ship trajectory reconstruction from ais sensory data via data quality control and prediction. *Mathematical Problems in Engineering*, 2020:1–9, 08 2020.

[18] Ticiana Coelho da Silva, Karine Zeitouni, and Jose Macedo. Online clustering of trajectory data stream. pages 112–121, 06 2016.

[19] Pasquale Coscia, Paolo Braca, Leonardo Millefiori, Francesco Palmieri, and P. Willett. Unsupervised maritime traffic graph learning with mean-reverting stochastic processes. 07 2018.

[20] Gerben K.D. de Vries, Willem Robert van Hage, and Maarten van Someren. Comparing vessel trajectories using geographical domain knowledge and alignments. In *2010 IEEE International Conference on Data Mining Workshops*, pages 209–216, 2010.

[21] Paul Dierckx. Curve and surface fitting with splines. In *Monographs on numerical analysis*, 1996.

[22] Shiting Ding, Zhiheng Li, Kai Zhang, and Feng Mao. A comparative study of frequent pattern mining with trajectory data. *Sensors*, 22:7608, 10 2022.

[23] Andrej Dobrkovic, Maria-Eugenia Iacob, and Jos Hillegersberg. Maritime pattern extraction and route reconstruction from incomplete ais data. *International Journal of Data Science and Analytics*, 5, 03 2018.

[24] Andrej Dobrkovic, Maria-Eugenia Iacob, and Jos Van Hillegersberg. Maritime pattern extraction from ais data using a genetic algorithm. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 642–651, 2016.

[25] David H. Douglas and Thomas K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10:112–122, 1973.

[26] Maria Bala Duggimpudi, Shaaban Abbady, Jian Chen, and Vijay V. Raghavan. Spatio-temporal outlier detection algorithms based on computing behavioral outlierness factor. *Data & Knowledge Engineering*, 122:1–24, 2019.

[27] Lubna Eljabu, Mohammad Etemad, and Stan Matwin. Anomaly detection in maritime domain based on spatio-temporal analysis of ais data using graph neural networks. *2021 5th International Conference on Vision, Image and Signal Processing (ICVISP)*, pages 142–147, 2021.

[28] Lubna Eljabu, Mohammad Etemad, and Stan Matwin. Destination port detection for vessels: An analytic tool for optimizing port authorities resources. *International Journal of Civil and Architectural Engineering*, 15(8):398 – 406, 2021.

[29] Lubna Eljabu, Mohammad Etemad, and Stan Matwin. Spatial clustering method of historical ais data for maritime traffic routes extraction. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 893–902, 2022.

[30] Lubna Eljabu, Mohammad Etemad, and Stan Matwin. Spatial clustering model of vessel trajectory to extract sailing routes based on ais data. *International Journal of Computer and Systems Engineering*, 16(10):482 – 492, 2022.

[31] Lubna Eljabu, Mohammad Etemad, and Stan Matwin. Charting the course of ship track prediction: A novel approach for maritime traffic analysis and enhanced situational awareness. In *2023 IEEE International Conference on Big Data (BigData)*, pages 2588–2597, 2023.

[32] Mica R. Endsley. *Designing for Situation Awareness: An Approach to User-Centered Design, Second Edition*. CRC Press, Inc., USA, 2nd edition, 2011.

[33] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. KDD'96, page 226–231. AAAI Press, 1996.

[34] Vladimir Estivill-Castro. Why so many clustering algorithms: A position paper. *SIGKDD Explor. Newsl.*, 4(1):65–75, jun 2002.

[35] Mohammad Etemad. Novel algorithms for trajectory segmentation based on interpolation-based change detection strategies. *Dalhousie Faculty of Graduate Studies Online Theses 2020*.

[36] Mohammad Etemad, Zahra Etemad, Amílcar Soares, Vania Bogorny, Stan Matwin, and Luis Torgo. Wise sliding window segmentation: A classification-aided approach for trajectory segmentation. In *Canadian Conference on Artificial Intelligence*, pages 208–219. Springer, 2020.

[37] Dominik Filipiak, Krzysztof Wecel, Milena Strozyna, Michał Michalak, and Witold Abramowicz. Extracting maritime traffic networks from ais data using evolutionary algorithm. *Business and Information Systems Engineering*, 62, 10 2020.

[38] Nicola Forti, Leonardo M. Millefiori, and Paolo Braca. Unsupervised extraction of maritime patterns of life from automatic identification system data. In *OCEANS 2019 - Marseille*, pages 1–5, 2019.

[39] Germán González-Almagro, Daniel Peralta, Eli De Poorter, José-Ramón Cano, and Salvador García. Semi-supervised constrained clustering: An in-depth overview, ranked taxonomy and future research directions. *arXiv preprint arXiv:2303.00522*, 2023.

[40] Racha Gouareb, Francois Can, Sohrab Ferdowsi, and Douglas Teodoro. Vessel destination prediction using a graph-based machine learning model. In *Network Science: 7th International Winter Conference, NetSci-X 2022, Porto, Portugal, February 8–11, 2022, Proceedings*, page 80–93, Berlin, Heidelberg, 2022. Springer-Verlag.

[41] Manish Gupta, Jing Gao, Charu C. Aggarwal, and Jiawei Han. Outlier detection for temporal data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 26(9):2250–2267, 2014.

[42] Cleasby I.R., Wakefield E.D., and Morrissey B.J. *Using time-series similarity measures to compare animal movement trajectories in ecology.* . Behav Ecol Sociobiol 73, 151, 2019.

[43] Berndt J. D. and J. Clifford. *Using Dynamic Time Warping to Find Patterns in Time Series*. In Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining, AAAIWS'94.AAAI Press, 1994.

[44] Witowski K., Feucht M., and Stander N. *An Effective Curve Matching Metric for Parameter Identification using Partial Mapping*. 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, 2012.

[45] Anders Kallner and Elvar Theodorsson. Repeatability imprecision from analysis of duplicates of patient samples and control materials. *Scandinavian Journal of Clinical and Laboratory Investigation*, 80(3):210–214, 2020. PMID: 31899972.

[46] Nitin Kamra, Hao Zhu, Dweep Trivedi, Ming Zhang, and Yan Liu. Multi-agent trajectory prediction with fuzzy query attention. NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[47] Leonard Kaufman and Peter J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

[48] Hyun-Suk Kim, Eunkyu Lee, Eui-Jong Lee, Jin-Won Hyun, In-Young Gong, Kyungsup Kim, and Yun-Sok Lee. A study on grid-cell-type maritime traffic distribution analysis based on ais data for establishing a coastal maritime transportation network. *Journal of Marine Science and Engineering*, 11(2), 2023.

[49] Ioannis Kontopoulos, Iraklis Varlamis, and Konstantinos Tserpes. A distributed framework for extracting maritime traffic patterns. *International Journal of Geographical Information Science*, 35(4):767–792, 2021.

[50] S. Kos, Mate Vukić, and David Brcic. Use of universal protocol for entering the port of destination in ais device. 2013.

[51] Nicolas Le Guillarme and Xavier Lerouvreur. Unsupervised extraction of knowledge from s-ais data for maritime situational awareness. In *Proceedings of the 16th International Conference on Information Fusion*, pages 2025–2032, 2013.

[52] Jeong-Seok Lee and Ik-Soon Cho. Extracting the maritime traffic route in korea based on probabilistic approach using automatic identification system big data. *Applied Sciences*, 12(2), 2022.

[53] Olga Lezhnina and Gábor Kismihók. Latent class cluster analysis: Selecting the number of clusters. *MethodsX*, 9:101747, 2022.

[54] Huaipeng Li. Typical trajectory extraction method for ships based on ais data and trajectory clustering. In *2021 2nd International Conference on Artificial Intelligence and Information Systems*, ICAIIS 2021, New York, NY, USA, 2021. Association for Computing Machinery.

[55] Ye Li and Hongxiang Ren. Visual analysis of vessel behaviour based on trajectory data: A case study of the yangtze river estuary. *ISPRS Int. J. Geo Inf.*, 11(4):244, 2022.

[56] Lingbo Liu, Zhilin Qiu, Guanbin Li, Qing Wang, Wanli Ouyang, and Liang Lin. Contextualized spatial–temporal network for taxi origin-destination demand prediction. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3875–3887, 2019.

[57] Zhao Liu, Hairuo Gao, Mingyang Zhang, Ran Yan, and Jingxian Liu. A data mining method to extract traffic network for maritime transport management. *Ocean & Coastal Management*, 239:106622, 2023.

[58] Búgvi Benjamin Magnussen, Nikolaj Bläser, Rune Møller Jensen, and Kenneth Ylänen. Destination prediction of oil tankers using graph abstractions and recurrent neural networks. In Martijn Mes, Eduardo Lalla-Ruiz, and Stefan Voß, editors, *Computational Logistics*, pages 51–65, Cham, 2021. Springer International Publishing.

[59] Yingchi Mao, Haishi Zhong, Xianjian Xiao, and Xiaofang Li. A segment-based trajectory similarity measure in the urban transportation systems. *Sensors*, 17(3), 2017.

[60] Ronaldo dos Santos Mello, Vania Bogorny, Luis Otavio Alvares, Luiz Henrique Zambom Santana, Carlos Andres Ferrero, Angelo Augusto Frozza, Geomar Andre Schreiner, and Chiara Renso. Master: A multiple aspect view on trajectories. *Transactions in GIS*, 23(4):805–822, 2019.

[61] Thomas Mestl, Dnv Gl, Høvik Norway, and Kay Dausendschön. Port eta prediction based on ais data. 05 2016.

[62] Duc-Duy Nguyen, Chan Le Van, and Muhammad Intizar Ali. Vessel trajectory prediction using sequence-to-sequence models over spatial grid. In *Proceedings of the 12th ACM International Conference on Distributed and Event-Based Systems*, pages 258–261, 2018.

[63] Lee Sang-min Nguyen Van-Suong, Im Nam-kyun. The interpolation method for the missing ais data of ship. *J Navig Port Res*, 39(5):377–384, 2015.

[64] Shengke Ni, Yao Cai, and Xin Wang. Modelling of ship's trajectory planning in collision situations by hybrid genetic algorithm. *Polish Maritime Research*, 25:14–25, 09 2018.

[65] Shengke Ni, Zhengjiang Liu, and Yao Cai. Ship manoeuvrability-based simulation for ship navigation in collision situations. *Journal of Marine Science and Engineering*, 7(4), 2019.

[66] Shem Otoi Onyango, Solomon Amoah Owiredu, Kwang-Il Kim, and Sang-Lok Yoo. A quasi-intelligent maritime route extraction from ais data. *Sensors*, 22(22), 2022.

[67] Ewa Osekowska, Henric Johnson, and Bengt Carlsson. Maritime vessel traffic modeling in the context of concept drift. *Transportation Research Procedia*, 25:1457–1476, 2017. World Conference on Transport Research - WCTR 2016 Shanghai. 10-15 July 2016.

[68] Giuliana Pallotta, Michele Vespe, and Karna Bryan. Vessel pattern knowledge discovery from ais data: A framework for anomaly detection and route prediction. *Entropy*, 15(6):2218–2245, 2013.

[69] Jinwan Park, Jungsik Jeong, and Youngsoo Park. Ship trajectory prediction based on bi-lstm using spectral-clustered ais data. *Journal of Marine Science and Engineering*, 9(9), 2021.

[70] Kikun Park, Sunghyun Sim, and Hyerim Bae. Vessel estimated time of arrival prediction system based on a path-finding algorithm. *Maritime Transport Research*, 2:100012, 2021.

[71] Joachim Pum. *A practical guide to validation and verification of analytical methods in the clinical laboratory*. 01 2019.

[72] Feiyang Ren, Yi Han, Shaohan Wang, and He Jiang. A novel high-dimensional trajectories construction network based on multi-clustering algorithm. *EURASIP Journal on Wireless Communications and Networking*, 2022, 03 2022.

[73] Mohsen Rezaie and Nicolas Saunier. Trajectory clustering performance evaluation: If we know the answer, it's not clustering.

[74] H. Rong, A.P. Teixeira, and C. Guedes Soares. Maritime traffic probabilistic prediction based on ship motion pattern extraction. *Reliability Engineering & System Safety*, 217:108061, 2022.

[75] Yu Rong, Zhong Zhuang, Zhengwei He, and Xuming Wang. A maritime traffic network mining method based on massive trajectory data. *Electronics*, 11(7), 2022.

[76] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 410–420, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[77] Hang Shao, Abhishek Kumar, and P. Thomas Fletcher. The riemannian geometry of deep generative models. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 428–4288, 2018.

[78] Helong Shen, Huang Tang, and Yong Yin. A novel method for ship trajectory clustering. *International Journal of Naval Architecture and Ocean Engineering*, 14:100474, 2022.

[79] Pan Sheng and Jingbo Yin. Extracting shipping route patterns by trajectory clustering model based on automatic identification system data. *Sustainability*, 10:2327, 07 2018.

[80] Rongxin Song, Yuanqiao Wen, Liang Huang, Fan Zhang, and Chunhui Zhou. *Data-driven cognitive modeling and semantic reasoning of ship behavior*, pages 269–276. 07 2021.

[81] Giannis Spiliopoulos, Dimitrios Zissis, and Konstantinos Chatzikokolakis. A big data driven approach to extracting global trade patterns. In Christos Doulkeridis, George A. Vouros, Qiang Qu, and Shuhui Wang, editors, *Mobility Analytics for Spatio-Temporal and Social Data*, pages 109–121, Cham, 2018. Springer International Publishing.

[82] Cynthia Sung, Dan Feldman, and Daniela Rus. Trajectory clustering for motion prediction. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1547–1552, 2012.

[83] Yongfeng Suo, Wenke Chen, Christophe Claramunt, and Shenhua Yang. A ship trajectory prediction framework based on a recurrent neural network. *Sensors*, 20(18), 2020.

[84] Eiter T. and Mannila. H. *Computing discrete Fréchet distance*. Technical report, CD-TR 94/64, Christian Doppler Laboratory for Expert Systems, Technische Universität Wien, Wien,, 1994.

[85] Abdel Aziz Taha and Allan Hanbury. An efficient algorithm for calculating the exact hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(11):2153–2163, 2015.

[86] Chunhua Tang, Meiyue Chen, Jiahuan Zhao, Tao Liu, Kang Liu, Huaran Yan, and Yingjie Xiao. A novel ship trajectory clustering method for finding overall and local features of ship trajectories. *Ocean Engineering*, 241:110108, 2021.

[87] Yaguang Tao, Alan Both, Rodrigo I. Silveira, Kevin Buchin, Stef Sijben, Ross S. Purves, Patrick Laube, Dongliang Peng, Kevin Toohey, and Matt Duckham. A comparative analysis of trajectory similarity measures. *GIScience & Remote Sensing*, 58(5):643–669, 2021.

[88] Enmei Tu, Guanghao Zhang, Shangbo Mao, Lily Rachmawati, and Guangbin Huang. Modeling historical ais data for vessel path prediction: A comprehensive treatment. *ArXiv*, abs/2001.01592, 2020.

[89] Chris Veness. Calculate distance, bearing and more between latitude/longitude points. https://www.movable-type.co.uk/scripts/latlong.html, 2022.

[90] Roberto Vettor and Carlos Guedes Soares. Detection and analysis of the main routes of voluntary observing ships in the north atlantic. *Journal of Navigation*, 68:397–410, 2015.

[91] Chao Wang, Fangzheng Lyu, Sensen Wu, Yuanyuan Wang, Liuchang Xu, Feng Zhang, Shaowen Wang, Yongheng Wang, and Zhenhong Du. A deep trajectory clustering method based on sequence-to-sequence autoencoder model. *Transactions in GIS*, 26(4):1801–1820, 2022.

[92] Guiling Wang, Jinlong Meng, and Yanbo Han. Extraction of maritime road networks from large-scale ais data. *IEEE Access*, 7:123035–123048, 2019.

[93] Sainan Wang, Suixiang Gao, and Wenguo Yang. Ship route extraction and clustering analysis based on automatic identification system data. In *2017 Eighth International Conference on Intelligent Control and Information Processing (ICICIP)*, pages 33–38, 2017.

[94] Wells Wang, Junchi Bin, Amirhossein Zaji, Richard Halldearn, Fabien Guillaume, Eric Li, and Zheng Liu. A multi-task learning-based framework for global maritime trajectory and destination prediction with ais data. *Maritime Transport Research*, 3:100072, 2022.

[95] Lin Wu, Yongjun Xu, and Fei Wang. Identifying port calls of ships by uncertain reasoning with trajectory data. *ISPRS International Journal of Geo-Information*, 9(12), 2020.

[96] Xing Wu, Aesha L. Mehta, Victor A. Zaloom, and Brian N. Craig. Analysis of waterway transportation in southeast texas waterway based on ais data. *Ocean Engineering*, 121:196–209, 2016.

[97] Zhe Xiao, Xiuju Fu, Liye Zhang, and Rick Siow Mong Goh. Traffic pattern mining and forecasting technologies in maritime traffic service networks: A comprehensive survey. *IEEE Transactions on Intelligent Transportation Systems*, 21(5):1796–1825, 2020.

[98] Zhaojin Yan, Yijia Xiao, Liang Cheng, Rong He, Xiaoguang Ruan, Xiao Zhou, Manchun Li, and Ran Bin. Exploring ais data for intelligent maritime routes extraction. *Applied Ocean Research*, 101:102271, 2020.

[99] Cheng-Hong Yang, Guan-Cheng Lin, Chih-Hsien Wu, Yen-Hsien Liu, Yi-Chuan Wang, and Kuo-Chang Chen. Deep learning for vessel trajectory prediction using clustered ais data. *Mathematics*, 10(16), 2022.

[100] Jiawei Yang, Xu Tan, and Sylwan Rahardja. Mipo: How to detect trajectory outliers with tabular outlier detectors. *Remote Sensing*, 14(21), 2022.

[101] Wang Yitao, Yang Lei, and Song Xin. Route mining from satellite-ais data using density-based clustering algorithm. *Journal of Physics: Conference Series*, 1616:012017, 08 2020.

[102] Bakht Zaman, Dusica Marijan, and Tetyana Kholodna. Interpolation-based inference of vessel trajectory waypoints from sparse ais data in maritime. *Journal of Marine Science and Engineering*, 11(3), 2023.

[103] Chengkai Zhang. *Ais data-driven general vessel destination prediction: a trajectory similarity-based approach*. PhD thesis, University of British Columbia, 2019.

[104] Chengkai Zhang, Junchi Bin, Wells Wang, Xiang Peng, Rui Wang, Richard Halldearn, and Zheng Liu. Ais data driven general vessel destination prediction: A random forest based approach. *Transportation Research Part C: Emerging Technologies*, 118:102729, 2020.

[105] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: An efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, SIGMOD '96, page 103–114, New York, NY, USA, 1996. Association for Computing Machinery.

[106] Yuanqiang Zhang and Weifeng Li. Dynamic maritime traffic pattern recognition with online cleaning, compression, partition, and clustering of ais data. *Sensors*, 22(16), 2022.

[107] Liangbin Zhao and Guoyou Shi. A trajectory clustering method based on douglas-peucker compression and density for marine traffic pattern recognition. *Ocean Engineering*, 172:456–467, 2019.

[108] Rong Zhen, Yongxing Jin, Qinyou Hu, Zheping Shao, and Nikitas Nikitakos. Maritime anomaly detection within coastal waters based on vessel trajectory clustering and naïve bayes classifier. *Journal of Navigation*, 70(3):648–670, 2017.

[109] Yu Zheng. Trajectory data mining: An overview. *ACM Trans. Intell. Syst. Technol.*, 6(3), may 2015.

[110] Xiaolei Zhu, Keiran C. Thompson, and Todd J. Martínez. Geodesic interpolation for reaction pathways. *The Journal of Chemical Physics*, 150(16):164103, 04 2019.

[111] Nikolas Zygouras, Giannis Spiliopoulos, and Dimitris Zissis. Detecting representative trajectories from global ais datasets. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 2278–2285, 2021.

Appendix A

Outputs of The Proposed Model

A.1 Overview of AIS Data Details

This thesis utilizes AIS data collected from the Halifax Harbour area in Nova Scotia, Canada, and the Gulf of Mexico, a basin and marginal sea of the Atlantic Ocean largely surrounded by the North American continent. These AIS datasets were obtained by our team at the Institute of Big Data Analytics. The collected AIS data includes vessel information such as position, speed over ground (SOG), course over ground (COG), vessel type, etc., used for maritime traffic analysis and destination port prediction.

event_time	location_coordinates_0	location_coordinate:	position_accura	mmsi	sog	cog
2019-04-01T18:20:20.619Z	-63.577255	44.66085833	1	316027034	7.5	194.9
2019-04-08T14:34:57.314Z	-63.57737667	44.66032667	1	316027034	7.4	211.7
2019-04-01T18:20:51.073Z	-63.57734833	44.65978167	1	316027034	7.7	172.8
2019-04-12T02:07:17.408Z	-63.57616167	44.66039333	1	316027034	7.8	194.4
2019-04-08T14:34:27.474Z	-63.57626333	44.66096333	1	316027034	8	240.6
2019-04-12T02:06:46.754Z	-63.57528833	44.66124333	1	316027034	7.7	238.2
2019-04-02T17:19:16.779Z	-63.57316833	44.66068	1	316027034	7.3	192.3
2019-04-08T11:34:23.954Z	-63.57401833	44.660025	1	316027034	8.2	201.5
2019-04-12T16:04:28.407Z	-63.573875	44.65994667	1	316027034	8.4	210.4
2019-04-12T16:04:49.303Z	-63.573875	44.65994667	1	316027034	8.4	210.4
2019-04-12T02:07:47.234Z	-63.57603667	44.65937167	1	316027034	7.8	156.5

(a) Example of vessels' information from AIS datasets from Halifax.

mmsi	datetime	time	lon	lat	sog	cog	imo	name	vesseltype	vesseltype	callsign	flag	gross_tc	summer	length_t	year_bu
201972208	2021-09-05 3:03	1630810999	-80.6192	23.85812	16.2	239										
203571200	2021-06-29 9:45	1624959906	-82.4997	23.08744	0	319		TALOA	Sailing Vesse	Sailing Vessel	OEX5384	Austria [AT]			10 x 3 m	
203571200	2021-07-02 21:43	1625262199	-82.8425	23.1168	3.4	262		TALOA	Sailing Vesse	Sailing Vessel	OEX5384	Austria [AT]			10 x 3 m	
203571200	2021-07-02 22:39	1625265562	-82.8919	23.11031	3.1	261		TALOA	Sailing Vesse	Sailing Vessel	OEX5384	Austria [AT]			10 x 3 m	
203571200	2021-07-06 1:59	1625536777	-86.4807	21.36436	3.2	236		TALOA	Sailing Vesse	Sailing Vessel	OEX5384	Austria [AT]			10 x 3 m	
203571200	2021-07-21 3:43	1626839026	-86.749	21.2494	0.1	58		TALOA	Sailing Vesse	Sailing Vessel	OEX5384	Austria [AT]			10 x 3 m	
203571200	2021-07-26 4:40	1627274448	-86.948	20.5177	0.1	332		TALOA	Sailing Vesse	Sailing Vessel	OEX5384	Austria [AT]			10 x 3 m	
203571200	2021-08-10 17:04	1628615067	-87.0284	20.46919	0.2	141		TALOA	Sailing Vesse	Sailing Vessel	OEX5384	Austria [AT]			10 x 3 m	
205042000	2021-07-03 21:00	1625346006	-81.042	18.14189	8.7	309		DELOS	Tanker	Crude Oil Tan	ONKR	Belgium [BI	156293	299986	336 x 60 m	2021
205042000	2021-07-04 0:11	1625357470	-81.422	18.43949	8.7	309		DELOS	Tanker	Crude Oil Tan	ONKR	Belgium [BI	156293	299986	336 x 60 m	2021
205042000	2021-07-04 4:30	1625373026	-81.9346	18.82946	8.7	309		DELOS	Tanker	Crude Oil Tan	ONKR	Belgium [BI	156293	299986	336 x 60 m	2021
205042000	2021-07-04 10:59	1625396373	-82.6985	19.42405	8.7	310		DELOS	Tanker	Crude Oil Tan	ONKR	Belgium [BI	156293	299986	336 x 60 m	2021
205042000	2021-07-04 14:59	1625410757	-83.2632	19.85002	8.9	308		DELOS	Tanker	Crude Oil Tan	ONKR	Belgium [BI	156293	299986	336 x 60 m	2021
205042000	2021-07-04 17:25	1625419505	-83.586	20.1061	8.9	307		DELOS	Tanker	Crude Oil Tan	ONKR	Belgium [BI	156293	299986	336 x 60 m	2021

(b) Example of vessels' information from AIS datasets form Gulf of Mexico.

Figure A.1: Example of vessels' information from AIS datasets captured in two distinct maritime areas.

The AIS data records, represented as rows in CSV files, can be visualized on a map as points. Each point corresponds to a moving vessel at a specific time, along with associated features described in the CSV tables in Figure A.1. These features

constitute a feature vector that defines vessel dynamics and characteristics. Due to the fact that the datasets were collected over different periods of time, the CSV files do not have the same number of columns. Notably, the destination feature is missing from both tables in Figure A.1. Specifically, location (latitude, longitude), MMSI, timestamp, and vessel type are the most important features for this work.

A.2 Semantic Trajectory

A semantic trajectory is a model for representing movement data that captures not only the spatial component (the path or route taken) but also the contextual and semantic information associated with the movement. It goes beyond just recording the geographic coordinates to include annotations about the moving vessel's situation and activities.

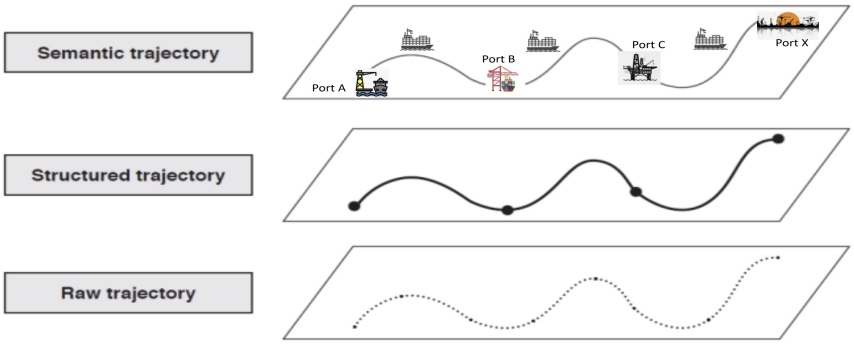
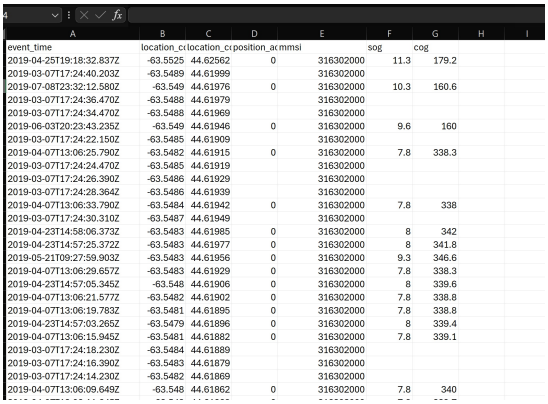


Figure A.2: An example of a semantic trajectory illustrating vessel movement.

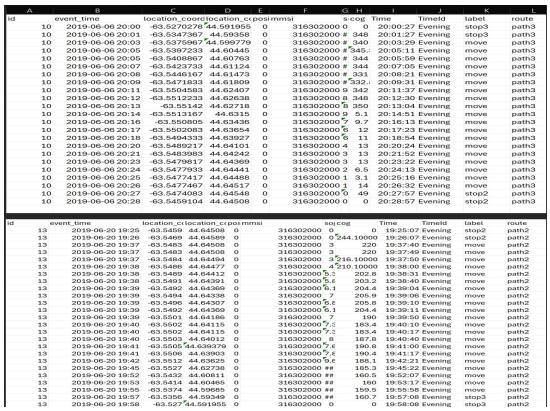
For instance, processing AIS data to create a semantic trajectory for tankers or cargo ships activities as it travels between various ports to load and unload cargo: *Departure from Port A* $\rightarrow$ *Voyage* $\rightarrow$ *Arrival at Port B* $\rightarrow$ *Departure from Port B* $\rightarrow$ *Voyage* $\rightarrow$ *Arrival at Port C* $\rightarrow$ *Departure from Port C* $\rightarrow$ *Voyage to the next port*. Figure A.2 illustrates this trajectory, which delineates the sequential phases of maritime transport operations, focusing on the movement between ports.

Semantic trajectories aim to provide a richer representation of movement by integrating spatial data with semantic information from maps and subject matter experts. This allows for deeper analysis and understanding of the moving vessel's behaviour, as shown in Figure A.3.



event_time	location_c	location_c	position_a	mmsi	sog	cog
2019-04-25T19:18:32.837Z	-63.5525	44.62562	0	316302000	11.3	179.2
2019-03-07T17:24:40.203Z	-63.5489	44.61999		316302000		
2019-07-08T23:32:12.580Z	-63.549	44.61976	0	316302000	10.3	160.6
2019-03-07T17:24:36.470Z	-63.5488	44.61979		316302000		
2019-03-07T17:24:34.470Z	-63.5488	44.61969		316302000		
2019-06-03T20:23:43.235Z	-63.549	44.61946	0	316302000	9.6	160
2019-03-07T17:24:22.150Z	-63.5485	44.61909		316302000		
2019-04-07T13:06:25.790Z	-63.5482	44.61915	0	316302000	7.8	338.3
2019-03-07T17:24:24.470Z	-63.5485	44.61919		316302000		
2019-03-07T17:24:26.390Z	-63.5486	44.61929		316302000		
2019-03-07T17:24:28.364Z	-63.5486	44.61939		316302000		
2019-04-07T13:06:33.790Z	-63.5484	44.61942	0	316302000	7.8	338
2019-03-07T17:24:30.310Z	-63.5487	44.61949		316302000		
2019-04-23T14:58:06.373Z	-63.5483	44.61985	0	316302000	8	342
2019-04-23T14:57:25.372Z	-63.5483	44.61977	0	316302000	8	341.8
2019-05-21T09:27:59.903Z	-63.5483	44.61956	0	316302000	9.3	346.6
2019-04-07T13:06:29.657Z	-63.5483	44.61929	0	316302000	7.8	338.3
2019-04-23T14:57:05.345Z	-63.548	44.61906	0	316302000	8	339.6
2019-04-07T13:06:21.577Z	-63.5482	44.61902	0	316302000	7.8	338.8
2019-04-07T13:06:19.783Z	-63.5481	44.61895	0	316302000	7.8	338.8
2019-04-23T14:57:03.265Z	-63.5479	44.61896	0	316302000	8	339.4
2019-04-07T13:06:15.945Z	-63.5481	44.61882	0	316302000	7.8	339.1
2019-03-07T17:24:18.230Z	-63.5484	44.61889		316302000		
2019-03-07T17:24:16.390Z	-63.5483	44.61879		316302000		
2019-03-07T17:24:14.230Z	-63.5482	44.61869		316302000		
2019-04-07T13:06:09.649Z	-63.548	44.61862	0	316302000	7.8	340

(a) Depiction of CSV file of the raw trajectory data.



id	event_time	location_c	location_c	position_a	mmsi	sog	cog	Time	TimeId	label	route
10	2019-06-06 20:00	-63.5270278	44.591955	0	316302000	0	0	20:00:27	Evening	stop3	path3
10	2019-06-06 20:01	-63.5347367	44.59358	0	316302000	8	348	20:01:27	Evening	stop3	path3
10	2019-06-06 20:03	-63.5375967	44.599779	0	316302000	8	340	20:03:29	Evening	move	path3
10	2019-06-06 20:05	-63.5397233	44.60445	0	316302000	8	345.:	20:05:11	Evening	move	path3
10	2019-06-06 20:05	-63.5408867	44.60763	0	316302000	8	344	20:05:59	Evening	move	path3
10	2019-06-06 20:07	-63.5423733	44.61124	0	316302000	8	344	20:07:05	Evening	move	path3
10	2019-06-06 20:08	-63.5446167	44.61473	0	316302000	8	331	20:08:21	Evening	move	path3
10	2019-06-06 20:09	-63.5471833	44.61809	0	316302000	8	332.:	20:09:31	Evening	move	path3
10	2019-06-06 20:11	-63.5504583	44.62407	0	316302000	9	342	20:11:37	Evening	move	path3
10	2019-06-06 20:12	-63.5512233	44.62638	0	316302000	8	348	20:12:30	Evening	move	path3
10	2019-06-06 20:13	-63.55142	44.62718	0	316302000	5	350	20:13:04	Evening	move	path3
10	2019-06-06 20:14	-63.5513167	44.6315	0	316302000	9	5.1	20:14:51	Evening	move	path3
10	2019-06-06 20:16	-63.550805	44.63436	0	316302000	7	9.7	20:16:13	Evening	move	path3
10	2019-06-06 20:17	-63.5502083	44.63654	0	316302000	5	12	20:17:23	Evening	move	path3
10	2019-06-06 20:18	-63.5494333	44.63927	0	316302000	6	11	20:18:54	Evening	move	path3
10	2019-06-06 20:20	-63.5489217	44.64101	0	316302000	4	13	20:20:24	Evening	move	path3
10	2019-06-06 20:21	-63.5483983	44.64242	0	316302000	3	13	20:21:52	Evening	move	path3
10	2019-06-06 20:23	-63.5479817	44.64369	0	316302000	3	13	20:23:22	Evening	move	path3
10	2019-06-06 20:24	-63.5477933	44.64441	0	316302000	2	6.5	20:24:13	Evening	move	path3
10	2019-06-06 20:25	-63.5477417	44.64488	0	316302000	1	3.1	20:25:16	Evening	move	path3
10	2019-06-06 20:26	-63.5477467	44.64517	0	316302000	1	14	20:26:32	Evening	move	path3
10	2019-06-06 20:27	-63.5474083	44.64548	0	316302000	0	49	20:27:57	Evening	stop2	path3
10	2019-06-06 20:28	-63.5459104	44.64508	0	316302000	0	0	20:28:57	Evening	stop2	path3

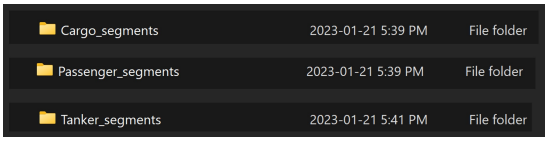
id	event_time	location_c	location_c	position_a	mmsi	sog	cog	Time	TimeId	label	route
13	2019-06-20 19:25	-63.5459	44.64508	0	316302000	0	0	19:25:07	Evening	stop2	path2
13	2019-06-20 19:26	-63.5469	44.64589	0	316302000	0	244.10000	19:26:07	Evening	stop2	path2
13	2019-06-20 19:37	-63.5483	44.64508	0	316302000	3	220	19:37:40	Evening	move	path2
13	2019-06-20 19:37	-63.5483	44.64508	0	316302000	3	220	19:37:49	Evening	move	path2
13	2019-06-20 19:37	-63.5484	44.64494	0	316302000	3	216.10000	19:37:50	Evening	move	path2
13	2019-06-20 19:38	-63.5486	44.64477	0	316302000	4	210.10000	19:38:00	Evening	move	path2
13	2019-06-20 19:38	-63.5489	44.6442	0	316302000	5.1	202.8	19:38:31	Evening	move	path2
13	2019-06-20 19:38	-63.5491	44.64391	0	316302000	5.6	203.2	19:38:40	Evening	move	path2
13	2019-06-20 19:39	-63.5492	44.64369	0	316302000	5.1	204.4	19:39:04	Evening	move	path2
13	2019-06-20 19:39	-63.5494	44.64338	0	316302000	7	205.9	19:39:06	Evening	move	path2
13	2019-06-20 19:39	-63.5496	44.64307	0	316302000	5.8	205.8	19:39:10	Evening	move	path2
13	2019-06-20 19:39	-63.5492	44.64369	0	316302000	6.1	204.4	19:39:11	Evening	move	path2
13	2019-06-20 19:39	-63.5501	44.64186	0	316302000	7	190	19:39:50	Evening	move	path2
13	2019-06-20 19:40	-63.5502	44.64115	0	316302000	7.2	183.4	19:40:10	Evening	move	path2
13	2019-06-20 19:40	-63.5502	44.64115	0	316302000	7.2	183.4	19:40:17	Evening	move	path2
13	2019-06-20 19:40	-63.5503	44.64012	0	316302000	8	187.8	19:40:40	Evening	move	path2
13	2019-06-20 19:41	-63.5505	44.639379	0	316302000	7.6	190.8	19:41:00	Evening	move	path2
13	2019-06-20 19:41	-63.5506	44.63903	0	316302000	7.6	190.4	19:41:17	Evening	move	path2
13	2019-06-20 19:42	-63.5512	44.63625	0	316302000	9.6	188.1	19:42:21	Evening	move	path2
13	2019-06-20 19:45	-63.5527	44.62738	0	316302000	##	185.3	19:45:22	Evening	move	path2
13	2019-06-20 19:52	-63.5432	44.60811	0	316302000	##	160.5	19:52:07	Evening	move	path2
13	2019-06-20 19:53	-63.5414	44.60465	0	316302000	##	160	19:53:17	Evening	move	path2
13	2019-06-20 19:55	-63.5374	44.59685	0	316302000	##	159.5	19:55:58	Evening	move	path2
13	2019-06-20 19:57	-63.5356	44.59349	0	316302000	##	160.7	19:57:08	Evening	stop3	path2
13	2019-06-20 19:58	-63.527	44.591955	0	316302000	0	0	19:58:08	Evening	stop3	path2

(b) Depiction of CSV file of the semantic trajectory data.

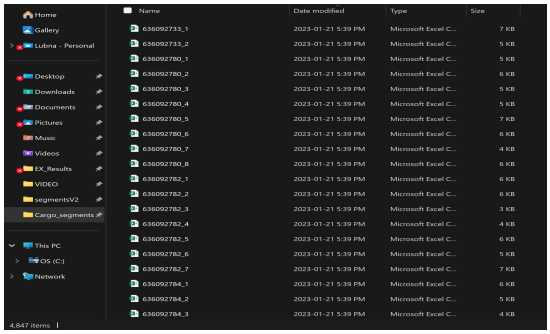
Figure A.3: Overview of the semantic trajectory.

A.3 Outputs of Segmentation

Segmenting vessel trajectories based on the intersection with port areas enables the identification of distinct voyage segments. These segments represent meaningful legs of a vessel's journey between successive port visits. Each identified segment is conveniently stored in individual CSV files for subsequent analysis, sharing, or processing. The storage approach follows a systematic naming convention using the vessel's MMSI and the segment number, such as "636092983_2.csv" and "636092983_7.csv". Figures A.4 and A.5 represent the outputs of the segmentation process.



(a) Folders containing CSV files of segmented trajectories by vessel type.

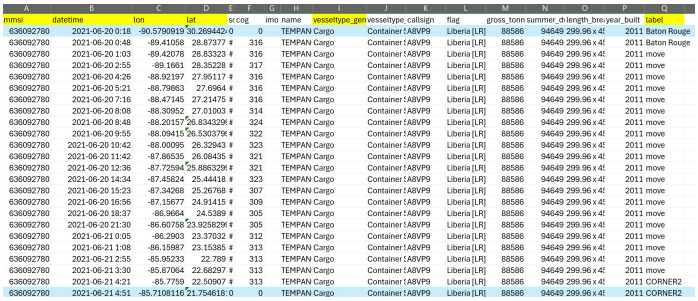


(b) CSV files of trajectory segments for cargo vessels.

Figure A.4: Overview of the segmentation outputs.

Segmenting trajectories and saving them as separate CSV files promotes better data organization, efficient storage and access, parallel processing capabilities, easier

data sharing, and improved fault tolerance, making it a useful approach for managing and analyzing large trajectory datasets. Figure A.5 shows the contents of the CSV file for a trajectory segment. The first and last rows of each segment, highlighted in light blue, represent the latitude and longitude coordinates of port areas' central points (i.e., "lon" and "lat" columns). These points remain consistent across segments that start and end at the same ports.



mmsi	datetime	lon	lat	sc	cog	imo	name	vesseltype_gen	vesseltype_callsign	flag	gross_tonn	summer_dr	length_bre	year_built	label
636092780	2021-06-20 0:18	-90.5790919	30.269442	0	0		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 Baton Rouge
636092780	2021-06-20 0:48	-89.41058	28.87377	#	316		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 Baton Rouge
636092780	2021-06-20 1:03	-89.42078	28.83323	#	316		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 2:55	-89.1661	28.35228	#	317		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 4:26	-88.92197	27.95117	#	316		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 5:21	-88.79863	27.6964	#	316		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 7:16	-88.47145	27.21475	#	316		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 8:08	-88.30952	27.01003	#	314		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 8:48	-88.20157	26.834329	#	324		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 9:55	-88.09415	26.530379	#	322		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 10:42	-88.00095	26.32943	#	323		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 11:42	-87.86535	26.08435	#	321		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 12:36	-87.72594	25.886329	#	321		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 14:34	-87.45824	25.44418	#	323		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 15:23	-87.34268	25.26768	#	307		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 16:56	-87.15677	24.91415	#	309		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 18:37	-86.9664	24.5389	#	305		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-20 21:30	-86.60758	23.925829	#	305		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-21 0:05	-86.2903	23.37032	#	312		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-21 1:08	-86.15987	23.15385	#	313		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-21 2:55	-85.95233	22.789	#	313		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-21 3:30	-85.87084	22.68297	#	313		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 move
636092780	2021-06-21 4:21	-85.7759	22.50907	#	313		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 CORNER2
636092780	2021-06-21 4:51	-85.7108116	21.754618	0	0		TEMPAN	Cargo	Container ! A8VP9	Liberia [LR]	88586	94649	299.96 x 4		2011 CORNER2

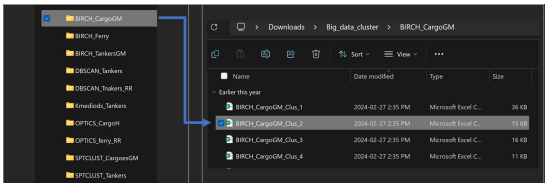
Figure A.5: Overview of the CSV file for a trajectory segment.

A.4 Outputs of Clustering

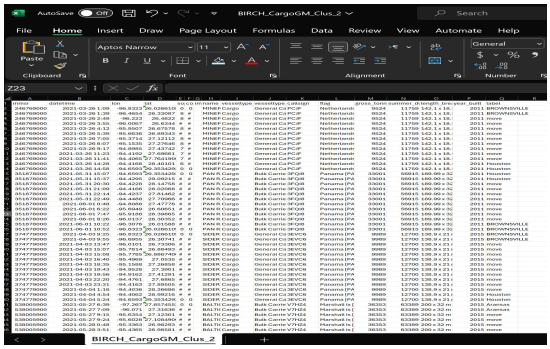
Clustering trajectory segments between pairs of ports enables the identification of typical or common route patterns observed by ships for specific origin-destination pairs. This analysis provides insights into maritime traffic flows and navigation behaviours and facilitates modelling of maritime traffic, all of which are valuable for enhancing maritime safety, efficiency, and analytics applications. The generated clusters of each clustering approach are saved as CSV files. CSV files containing the clustered segments of trajectories can be easily imported into other data analysis tools, visualization software, or geographic information systems (GIS) for further processing, analysis, or visualization of the clusters.

Figures A.6 and A.7 depict folders containing CSV files generated by the BIRCH clustering approach and our proposed SPTCLUST clustering approach, respectively. The CSV file generated by the BIRCH approach reveals clustering of segments that traverse in opposite directions. For instance, a segment starting from "Brownsville" ends at "Houston", followed by another segment starting from "Houston" to "Brownsville", and subsequently, segments with different port names. In contrast, the CSV file generated by the SPTCLUST approach demonstrates clustering of segments following a

single direction. For example, segments starting from “CORNER2” end at “Baton Rouge”.

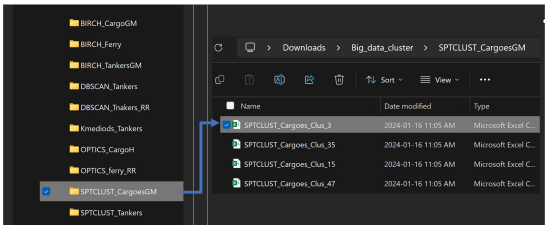


(a) Folders containing CSV files of the generated clusters.

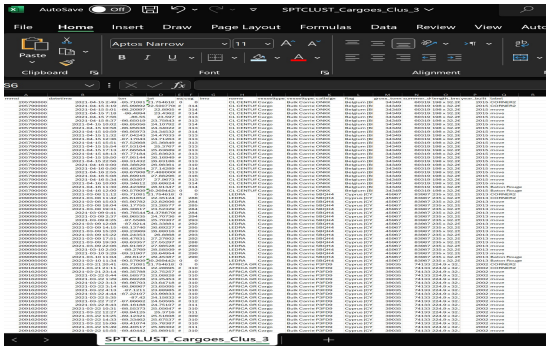


(b) CSV file of a generated cluster.

Figure A.6: Overview of the clustering outputs of BIRCH clustering approach.



(a) Folders containing CSV files of the generated clusters.



(b) CSV file of a generated cluster.

Figure A.7: Overview of the clustering outputs of SPTCLUST clustering approach.

Appendix B

B.1 Interpolation Methods

What is the preferred interpolation approach for filling gaps between each pair of consecutive trajectory points to generate a smooth and continuous reference route?

In interpolating sea vessel trajectory points, the choice between geodesic and linear interpolation is dictated by the distance and specific requirements of the application. Table B.1 summarizes the pros and cons of geodesic and linear interpolation methods. In our work, we use linear interpolation for filling the gap between successive trajectory points spaced minutes apart, as it effectively approximates routes over short distances, minimizing curvature effects and simplifying computations for real-time or local processing.

Table B.1: Interpolation methods for sea vessel trajectories: Geodesic vs. Linear.

Method	Description	Pros	Cons
Geodesic Interpolation	Calculates the shortest path between two points on the Earth's surface, considering the Earth's curvature [110].	Accurate for long-distance or global trajectories. Reflects realistic navigation paths that align with maritime practices [77].	Computationally intensive. Overhead is unnecessary for short distances where curvature is negligible.
Linear Interpolation	Connects points with a straight line in Cartesian coordinates [63].	Simple and computationally efficient. Effective for short distances where the Earth's curvature impact is minimal [63].	Can introduce significant errors over large distances due to ignoring Earth's curvature. Less accurate for global or long-distance trajectories.

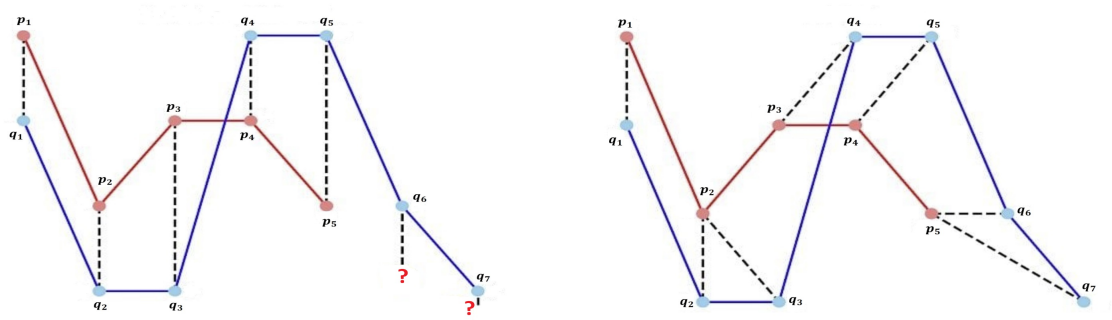
B.2 Similarity Measures of Trajectories

Is it possible to compute the Euclidean distance using two sequences with varying lengths?

In this example, we have two trajectories, P and Q, with different lengths. $P = [p_1, p_2, p_3, p_4, p_5]$, and $Q = [q_1, q_2, q_3, q_4, q_5, q_6, q_7]$. **We cannot calculate the Euclidean distance between P and Q since they don't have equal lengths.** Table B.2 and Figure B.1 comparing Euclidean distance, Dynamic Time Warping (DTW), and Discrete Fréchet distance (DFD) for calculating similarity between two trajectories. Euclidean distance is simple but requires aligned trajectories. DTW allows flexible matching but is more expensive. Discrete Fréchet distance is robust to outliers but sensitive to sampling.

Table B.2: Euclidean distance, Dynamic Time Warping (DTW), and Discrete Fréchet distance (DFD) for calculating similarity between two trajectories.

Method	Description	Pros	Cons
Euclidean distance	Sums the point-wise Euclidean distances between time-aligned trajectory points.	(i) Simple to compute. (ii) Efficient; interpretable.	Sensitive to misalignment and variations in trajectory length.
Dynamic Time Warping (DTW)	Finds an optimal non-linear alignment between trajectories that minimizes the summed distances between matched points.	Handles differences in speed and misalignment; captures similarities despite temporal variations.	(i) More computationally expensive than Euclidean distance. (ii) Assumes monotonic alignment.
Discrete Fréchet distance (DFD)	Finds an optimal coupling between trajectory vertices that minimizes the maximum distance between matched points.	Captures the overall shape and structure similarity; less sensitive to point-wise noise and variations.	(i) More computationally expensive than DTW. (ii) sensitive to outliers and might require simplification.



(a) Matching using Euclidean Distance. (b) Matching using DTW or DFD.

Figure B.1: DTW and DFD accommodate variations in trajectory lengths, allowing for better alignment of points and representing more sophisticated distance measures than Euclidean distance.