NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MARITIME CRAFT NAVIGATION PATTERN CHARACTERIZATION AND PREDICTION BASED ON AUTOMATED IDENTIFICATION SYSTEM DATA FROM THE PERSIAN GULF

by

Justin Knisely

September 2023

Thesis Advisor: Robert A. Koyak
Second Reader: Ruriko Yoshida

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2023	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE
MARITIME CRAFT NAVIGATION PATTERN CHARACTERIZATION AND PREDICTION BASED ON AUTOMATED IDENTIFICATION SYSTEM DATA FROM THE PERSIAN GULF

5. FUNDING NUMBERS

6. AUTHOR(S) Justin Knisely

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release. Distribution is unlimited.

12b. DISTRIBUTION CODE
A

13. ABSTRACT (maximum 200 words)

Maritime planners and analysts seek to characterize vessel behavior and predict future motion in highly congested and contested waterways. This thesis examines historical Automated Identification System (AIS) data from the Persian Gulf over six months. Extensive data preprocessing and outlier detection facilitates the construction of vessel "sub-tracks" between an origin and destination point. Next, cluster analysis of vessel sub-tracks is used to develop a Bayesian probability of cluster membership for each successive point in the sub-track. This probability, combined with dynamic and static vessel data including ship coordinates, identification number, ship size, and ship type, are used as features in a recurrent neural network model for predicting future ship trajectory. Finally, this thesis identifies potential modeling approaches that may be used to more accurately predict vessel trajectories or identify anomalous vessel behavior near contested waters.

14. SUBJECT TERMS
Navy, maritime planning, Persian Gulf, AIS, maritime traffic, maritime patterns, neural network, machine learning, clustering, regression

15. NUMBER OF PAGES
69

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

MARITIME CRAFT NAVIGATION PATTERN CHARACTERIZATION AND PREDICTION BASED ON AUTOMATED IDENTIFICATION SYSTEM DATA FROM THE PERSIAN GULF

Justin Knisely
Lieutenant, United States Navy
BS, United States Naval Academy, 2015

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2023

Approved by: Robert A. Koyak
Advisor

Ruriko Yoshida
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Maritime planners and analysts seek to characterize vessel behavior and predict future motion in highly congested and contested waterways. This thesis examines historical Automated Identification System (AIS) data from the Persian Gulf over six months. Extensive data preprocessing and outlier detection facilitates the construction of vessel "sub-tracks" between an origin and destination point. Next, cluster analysis of vessel sub-tracks is used to develop a Bayesian probability of cluster membership for each successive point in the sub-track. This probability, combined with dynamic and static vessel data including ship coordinates, identification number, ship size, and ship type, are used as features in a recurrent neural network model for predicting future ship trajectory. Finally, this thesis identifies potential modeling approaches that may be used to more accurately predict vessel trajectories or identify anomalous vessel behavior near contested waters.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

AIS Automatic Identification System

ARG Amphibious Ready Group

BN Bayesian network

CSV comma-separated values

FAA Federal Aviation Administration

GPS Global Positioning System

IMO International Maritime Organization

ITU International Telecommunication Union

LSTM long short-term memory

MMSI Maritime Mobile Service Identity

NN neural network

NOLH Nearly Orthogonal Latin Hybercube

PAM Partitioning Around Medoids

POI points of interest

ReLU Rectified Linear Unit

RNN recurrent neural network

TREAD Traffic Route Extraction and Anomaly Detection

USCG U.S. Coast Guard

USN U.S. Navy

VHF very high frequency

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

Military and governmental strategic guidance emphasizes the importance of freedom of navigation in strategic waterways, ranging from the Straits of Malacca in the Pacific to the Persian Gulf in the Middle East and the Panama Canal in Central America. The increase in global maritime trade presents navigational challenges for vessels, and opportunities for malign actors and adversaries to exert influence in surreptitious ways. Widespread adoption of regulations that mandate Automatic Identification System (AIS) usage by vessels, however, provides analysts a way to increase understanding of vessel movements and patterns in key maritime regions. AIS consists of ship, satellite, and ground-based transceivers that communicate static, dynamic, and voyage-related information of maritime vessels. This data includes but is not limited to dynamic information such as location coordinates, speed, and bearing, and static information such as vessel type and activity.

AIS data, however, is difficult to work with due to occasional missing, erroneous, or duplicate information, especially in manually-entered static reports. Analysts must devote substantial effort to clean usable data and remove inaccurate reports. Despite these obstacles, many modeling techniques have proven successful in predicting vessel motion. Researchers have leveraged AIS data to predict destinations, predict trajectories, and identify anomalous ship behavior. Our research uses AIS data from the Persian Gulf to build recurrent neural network (RNN) models that predict future locations of cargo and tanker vessels at 45 minutes and 120 minutes in the future.

In this thesis, we obtain AIS data from the Persian Gulf over a six-month period and undergo a structured process to clean and prepare it for analysis. First, we subset reports to geographic area that includes the Persian Gulf, Strait of Hormuz, and Arabian Sea. Next, we implement algorithms that remove outlier AIS reports, and build tracks from an individual vessel's time-sequential dynamic AIS reports. We break up tracks into smaller voyages called sub-tracks that include a ship's movement from one port or anchorage location to another. We finally select only sub-tracks that cross the Strait of Hormuz and enter the Persian Gulf for input to our trajectory prediction models. We conduct this final step as we are less interested in open ocean trajectory prediction, and more intrigued in the prediction of ship trajectories in an area with a high concentration of destination locations.

Most past research involving trajectory prediction clusters spatial data before predicting motion on a single cluster. Our approach instead clusters sub-tracks and then computes a probability of cluster membership at each successive coordinate on a vessel's trajectory using a Bayesian approach. We then build recurrent neural network (RNN) models to predict future locations, as RNNs are apt for capturing the dependencies of time-series data. These models include dynamic and static data, in addition to cluster membership probability, as features. We compare the accuracy of models with different feature combinations using median, mean, and standard deviation of miss distance, or the distance between the predicted and actual vessel coordinates.

We experiment with many combinations of hyperparameters and find that models that use a single long short-term memory (LSTM) and dense layer, and include a window of coordinates, speed, and bearing, produce the most accurate models. Including the probability of cluster membership as a feature improves model performance, while including the feature of binary cluster membership does not significantly alter model performance. Our best models predict vessel location 45 minutes and 120 minutes in the future with a median miss distance under 400 meters and under 800 meters respectively.

We find that RNN trajectory prediction models predict trajectories with reasonably small miss distances. Our models are less accurate in close proximity to ports and anchorage locations as predictions tend to lag when vessels make abrupt turns or stops. We propose that future research leverages distances to ports, oil rigs and fields, and anchorage locations in building RNN models. These features would allow for the model to account for proximity to the coast when making predictions. Furthermore, we propose that future work incorporates known maritime landmarks, leveraging the strengths of both RNN and geographic-based approaches in building regional trajectory prediction models.

Acknowledgments

I would like to thank my wife, Annie-Norah, for her patience and encouragement throughout my time in Monterey. I would like to thank Professor Robert Koyak for his input and guidance throughout this process. Lastly, I would like to thank the NPS Operations Research department for equipping our cohort with the tools and resources to make future impacts in our professions.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

Maritime shipping accounts for greater than 80 percent of global trade, as shipping remains the most efficient means of transportation for commerce according to the International Maritime Organization (2023a). Maritime security presents complex challenges across military, diplomatic, and economic lines. Recent global events, notably the COVID pandemic and Russian invasion of Ukraine, provide a cautionary tale urging corporations and nations to emphasize security on all waterways.

President Biden's recent National Security Strategy highlights the United States' commitment to affirming maritime freedom across the globe (White House 2022). It notes America's dedication to a free Indo-Pacific and emphasizes the importance of freedom of navigation in the Middle East's bodies of water including the Strait of Hormuz. Furthermore, Chief of Naval Operations Admiral Gilday's 2022 Navigation Plan directs the U.S. Navy (USN) to protect key maritime choke points (United States Navy 2022) as indicated in Figure 1.1. With a global refocus on maritime security and increased shipping demand comes the swelling of maritime traffic density, more chances for collisions, and enlarged opportunities for malign maritime actors. To combat these challenges, maritime analysts must leverage data sources that enable accurate and timely tracking of vessels.

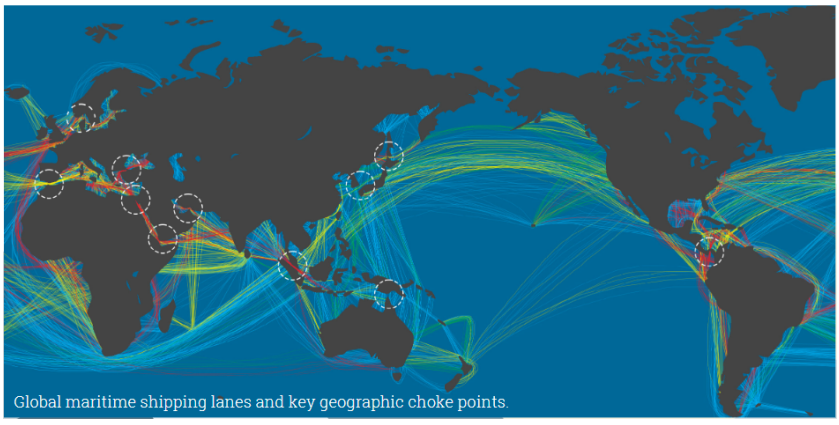

Figure 1.1. Global Maritime Shipping Lanes and Choke Points. Source: United States Navy (2022).

The most common data source of vessel tracking is the Automatic Identification System (AIS). Born out of the oil tanker Exxon Valdez running aground in Alaska and spilling millions of oil into Prince William Sound, AIS was created with the intent of improving navigational situational awareness, providing shore-based maritime tracking capabilities, and enabling coastal surveillance (Cutlip 2017). Figure 1.2 depicts a common vessel display system with AIS data. In 2002, the International Maritime Organization (IMO) mandated that participating countries require AIS transmitters and receivers on larger tonnage vessels (Cutlip 2017). Since then, different countries have endorsed additional rules for flagged vessels for not only collision avoidance and navigational awareness, but also marine environmental protection.

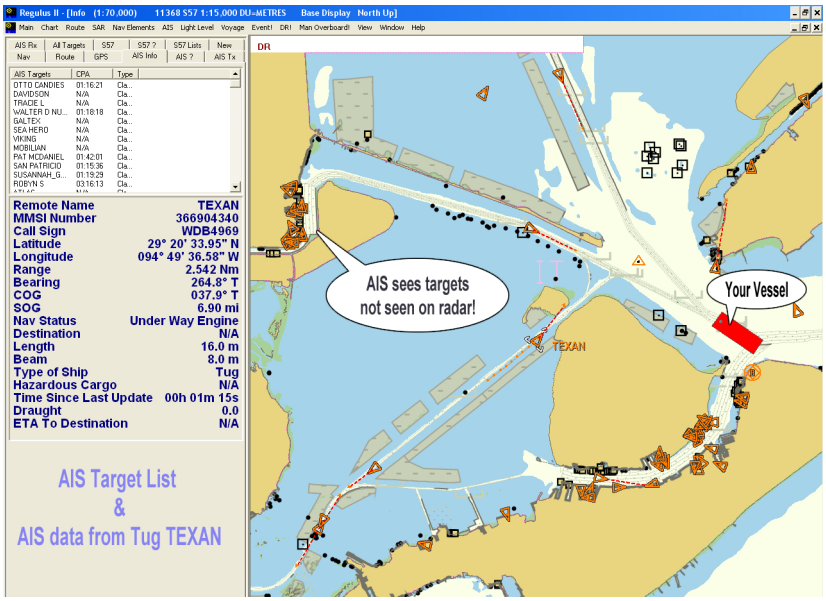

Figure 1.2. Example of a common vessel display system that includes AIS information. Source: United States Coast Guard (2023a).

According to the U.S. Coast Guard (USCG), AIS is a communications systems designed for maritime navigational safety (United States Coast Guard 2023a). The USCG notes that standards are maintained by the International Telecommunication Union (ITU) and adopted by the IMO. AIS data consists of static, dynamic, and voyage-related information but for the purposes of this research, only static and dynamic data are used. Static data includes vessel identification, call sign, type, and use and per IMO standards, is transmitted

every sixth minute. Dynamic data includes course, speed, and location coordinates. Vessel speed dictates the frequency of dynamic data, though most vessels transmit every 2–10 seconds when not at anchor.

Static Field	Description
MMSI	Ship Identification Number
CallSign	VHF radio communication name
Name	Owner designated name
ShipType	Cargo, Tanker, Passenger, Tug, High-Speed Craft, Pilot, Port Tender, Search & Rescue, Wing in Ground
ShipUse	Cargo, Diving, Dredging/Underwater, Military, Fishing, General, Pleasure, Sailing, Towing
DBow	Distance (m) from GPS transmitter to Bow
DStern	Distance (m) from GPS transmitter to Stern
DPort	Distance (m) from GPS transmitter to Port
DStbd	Distance (m) from GPS transmitter to Stbd
Destination	Destination Location
Filedate	Date of Static transmission

Table 1.1. Example of data included in a static AIS transmission. Adapted from Spire Maritime Documentation and Knowledge Base (2023) and United Nations Statistics Division (2020).

Base stations typically receive AIS data from a distance of approximately 50–100 kilometers, dependent upon receiving antenna height and other conditions (United Nations Statistics Division 2020), while ship to ship AIS range is usually less than 35–40 kilometers. Thus, maritime stations have difficulty tracking vessels in open ocean and large bodies of water via horizontal AIS reception. Satellite based receivers have attempted to fill this coverage gap as AIS transmits hundreds of kilometers vertically. Correlation of optical and radar imagery with satellite based AIS data has improved the accuracy of coverage, but challenges in reliable reception remain especially in congested waterways (United States Coast Guard 2023b).

Dynamic Field	Description
Latitude	-90° to 90°
Longitude	-180° to 180°
Msgtype	Message category from 1 to 27 (1-3: Position Report, 5: Static/Voyage-related data)
MMSI	Ship Identification Number
NavStat	Navigational status from 0 to 15 (0: underway using engine, 1: at anchor)
TurnRate	Rate of turn from -128 to 128 (°/min)
Speed	Speed over ground (knots)
Course	Course over ground from 0 to 360 (°)
Heading	Direction vessel is facing from 0 to 360 (°)
Time	Timestamp of Dynamic transmission

Table 1.2. Example of data included in a dynamic AIS transmission. Adapted from Spire Maritime Documentation and Knowledge Base (2023) and United Nations Statistics Division (2020).

Despite improvements in AIS-related technology and efforts to increase AIS use among vessels, several inaccuracies and issues still exist. Global Positioning System (GPS) errors may occur due to satellite speed changes, variations in satellite orbits, or receiver noise (Omholt-Jensen 2021). Congested channels may result in two vessels broadcasting in the same time slot, rendering the decoding process unfeasible. Other errors result from erroneous manual input. Vessel destination may not be updated as a vessel makes multiple port stops or some fields may be left blank (Omholt-Jensen 2021). Most egregious, vessel identification numbers, called Maritime Mobile Service Identity (MMSI), are often duplicated by different ships. MMSIs consist of nine numerical digits and are assigned locally by flag state authorities. Beyond the unintentional duplication of MMSIs, vessels from the same geographic region may intentionally enter an incorrect MMSI while navigating at the same time. Often, the same nine numbers are entered such as "111111111," rendering the tracking of vessels with this MMSI nearly impossible.

In this thesis, static and dynamic AIS data is used from a geographically constrained box encompassing the Persian Gulf and Strait of Hormuz. Figure 1.3 depicts AIS reports plotted in the Persian Gulf from our data set. With 21 percent of global oil trade transiting the Strait of Hormuz, the narrow channel remains a major strategic choke point (Decis and

Breton 2021). Despite U.S. and internationally-led efforts to deter violent incidents, attacks on commercial vessels have occurred frequently since 2019 and are often in response to tensions between Iran and Israel. On July 5, 2023, an Iranian corvette attacked the merchant oil tanker *Richard Voyager* off the coast of Oman. In response, the U.S. mobilized the *Bataan* Amphibious Ready Group (ARG) to join three U.S. destroyers already patrolling the region (LaGrone 2023).

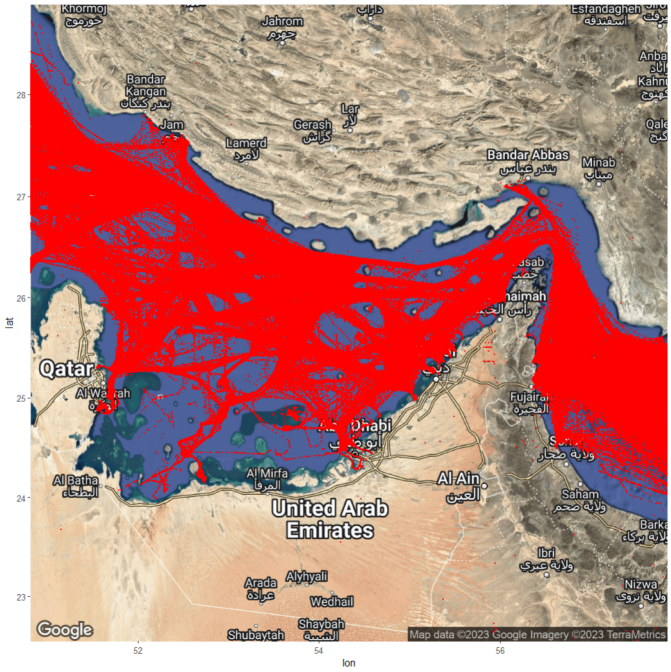

Figure 1.3. Coordinates from dynamic AIS data set plotted in Persian Gulf. Each coordinate is plotted as a small, red circle using the "ggmap" package in R. Source: Kahle and Wickham (2013).

Military leadership, commercial shipping vessels, and maritime security analysts all have a vested interest in understanding the maritime environment in highly-trafficked waterways like the Strait of Hormuz. Leveraging AIS data, models can predict trajectories of vessels and the uncertainty associated with these predictions to aid vessels and monitoring stations in several key ways. First, trajectory prediction may aid in identifying potential collisions and notifying vessels and stations in advance. In search and rescue operations where a vessel stops AIS transmission, trajectory prediction may identify the most likely current missing vessel location. For vessels transiting in maritime regions with frequent

violent incidents, trajectory prediction may aid in identifying anomalous behavior and cue vessels and security analysts to potential risks.

1.1 Thesis Structure

Our research uses AIS data to build neural network (NN) and recurrent neural network (RNN) models that predict future locations of cargo and tanker vessels at particular time intervals in the future. We demonstrate the accuracy of model predictions in a geographic location, the Persian Gulf, with well-defined transit routes but occasionally erroneous AIS transmissions. The remainder of this thesis is organized into four chapters. Chapter 2 reviews literature related to the topics of destination prediction, trajectory prediction, and anomaly detection. Chapter 3 details the methodology of data acquisition, preparation, and analysis techniques used. Chapter 4 describes the performance of our models and further analysis. Chapter 5 offers conclusions from our research and identifies topics of interest for future work.

CHAPTER 2:
Literature Review

Recent technological advances in computing power have reinvigorated interest in machine learning over the past decade. Autonomous ground vehicles, most commonly known as self-driving cars, have brought trajectory prediction to the forefront of discussion. Most autonomous ground vehicle motion prediction involves well-defined roads that restrict feasible routes. Air motion, on the other hand, involves a three dimensional space where deconfliction is often conducted by altering altitudes. Furthermore, the Federal Aviation Administration (FAA) requires all aircraft to carry airborne weather radar equipment and collision avoidance equipment. Of all forms of motion, maritime vessel deconfliction remains particularly challenging due to congested channels, varying vessel speeds, constant bearing changes, and dynamic weather. Although predetermined shipping routes and traffic separation schemes were adopted for safety reasons (International Maritime Organization 2023b), vessel trajectory prediction remains paramount in preventing collisions.

Maritime motion prediction can generally be classified into three categories: destination prediction, trajectory prediction, and anomaly detection.

2.1 Destination Prediction

Most research on destination prediction involves leveraging past historical trajectories and matching current vessel or vehicle trajectories to most similar historical routes. For example, the popular website MarineTraffic (2023) offers an API service that provides the top five most likely destination areas based on factors including past vessel movements, other vessels associated with the ownership company, and past vessel movements of a similar ship type.

de Brébisson et al. (2015) predict taxi destinations using GPS points, departure time, driver identification, and client information. The model leverages multi-layer perceptrons and bidirectional RNN to output a probability of membership to several thousand predetermined cluster centers for each trajectory and combines them in a weighted average for a final prediction.

von Eiff (2018) uses random forest models to conduct multiple predictions involving future destinations. First, she uses predictors such as weekday, port stay length, previous ports visited, and vessel country registration to predict if a vessel will stay in the nearby vicinity (e.g., the Persian Gulf). She then uses similar predictors to determine the most likely next port of visit, and investigates how well a Markov model or higher order network can be used to model the probabilities of next port of visit.

2.2 Anomaly Detection

Tester (2013) uses spatiotemporal clustering to gain knowledge of vessel behavior and attempts to screen vessel interactions for further investigation. His approach uses k-means clustering to refine clusters based on similarity of vessel speeds, headings, and distance. He concludes that this approach is viable for classifying paralleling and following movement that vessels may exhibit when engaged in nefarious activity.

Mascaro et al. (2014) detect anomalies in vessel tracks using static and dynamic Bayesian networks (BNs). They conclude that BNs offer decision making capabilities that other uncertainty models do not and suggest variations on using track probabilities to identify anomalies as future work.

Pallotta et al. (2013) use a methodology called Traffic Route Extraction and Anomaly Detection (TREAD) that leverages historical AIS data to detect anomalies, project future trajectories and identify patterns. The authors identify points of interests (POIs), typically entry and exit points, and create routes by clustering vessel movement between POIs. Elementary trajectories between POIs enable assessment of routes that deviate from the standard trajectory. The authors suggest that considering vessel interaction can help improve the interpretation of vessel behavior and understand potential anomalous motion.

Yan et al. (2022) use support vector machines and random forest models trained on satellite AIS data that leverage behavior characteristics and geometric features to classify vessel type. Their approach achieves over 92 percent accuracy in classifying a ship as cargo, tanker, fishing, passenger, or tug. Additionally, they present experimental results detect ships purposefully passing forged AIS transmissions to avoid observation.

Wolsing et al. (2022) review approaches to anomaly detection by maritime AIS tracks. The authors observe that most research attempts to model standard vessel behavior in specific regional geographic contexts, and that a lack of known anomalous acAIS situations often forces researchers to simulate anomalies or manually draw tracks.

2.3 Trajectory Prediction

Pallotta et al. (2014) further their 2013 research by using Ornstein-Uhlenbeck stochastic processes to predict vessel trajectories. The authors' methodology demonstrates simplicity in modelling of traffic routes and well-represented prediction errors several hours forward.

Young (2017) builds both random forest and NN models for trajectory prediction. His models use speed, latitude, longitude, course, a naive prediction using current speed and coordinates, and reference distances derived from evenly distributed points in the geographical space of the route as predictors. Latitude and longitude are predicted separately in this research and used to build prediction region bounds. It concludes that random forest models predict future location across all time intervals more accurately than NN models.

Li et al. (2019) predict long-term vessel motion by combining trajectory classification with a long short-term memory (LSTM) network. A longest common subsequence algorithm identifies trajectory similarity in clustering and then grouped trajectories are modeled by LSTM networks to improve prediction accuracy for long term motion where other parametric models perform poorly.

Wang et al. (2022) build convolutional NN, deep NN, and RNN models to predict vessel trajectories and destinations from multiple ports around the world. Their research investigates linkages between trajectory and destination prediction, concluding that the RNN model, based on LSTM layers, performs the best among all models. Additionally, the authors posit that multi-task learning architecture improves destination prediction accuracy compared to prior methods without this mechanism.

Liraz (2018) builds regression and classification RNN models to predict vessel movements at three future time intervals. He uses Nearly Orthogonal Latin Hybercube (NOLH) to aid in hyper-parameter selection and finds that a classification RNN approach produces

more accurate predictions. Finally, he investigates model use for anomaly detection and briefly investigates potential future work including geohashing.

CHAPTER 3:
Methodology

This chapter outlines the process we use to obtain and process AIS data into functional form for analysis, and the methods and approaches we use to examine research objectives. Unless otherwise stated, we use the R programming language (R Core Team 2021) for all data processing and analysis.

3.1 Data Description and Ingestion

Vessels transmit raw AIS messages to shore-based receivers, other vessels, or space-based satellite receivers. These messages typically contain six "positions," format, message count, message number, sequence identification, very high frequency (VHF) channel, the actual AIS data, and number of fill bits required. Upon receipt, messages must be parsed into usable form.

```
!AIVDM,1,1,,A,L4EgIM8FRcWr>jP00000,2*76>jp00000
```

Figure 3.1. Example of a raw AIS transmission before decoding.

Over 575,000 dynamic AIS messages were received daily in a region encompassing Southern and Eastern Africa and the Indian Ocean according to a study by Eriksen et al. (2018). Over a six month period, similar geographic regions can expect over one hundred million dynamic messages. For large geographic regions, the volume of AIS transmissions is difficult to manage. Our research uses a data set from January through June 2014 containing AIS messages across the entire globe, downsampled to approximately ten minute transmission intervals. We use a similar process to Bay (2017) and Young (2017) in converting raw AIS messages to convenient form, first using a regular expression script to convert raw messages to comma-separated values (CSV) files for each individual day in our date range. We then convert both static and dynamic CSV files to two dataframes in R. We geographically constrain our dynamic AIS data of interest to an area including the Persian Gulf, Strait of Hormuz, and Gulf of Oman, from (21, 31) degrees latitude and (48, 62) degrees longitude.

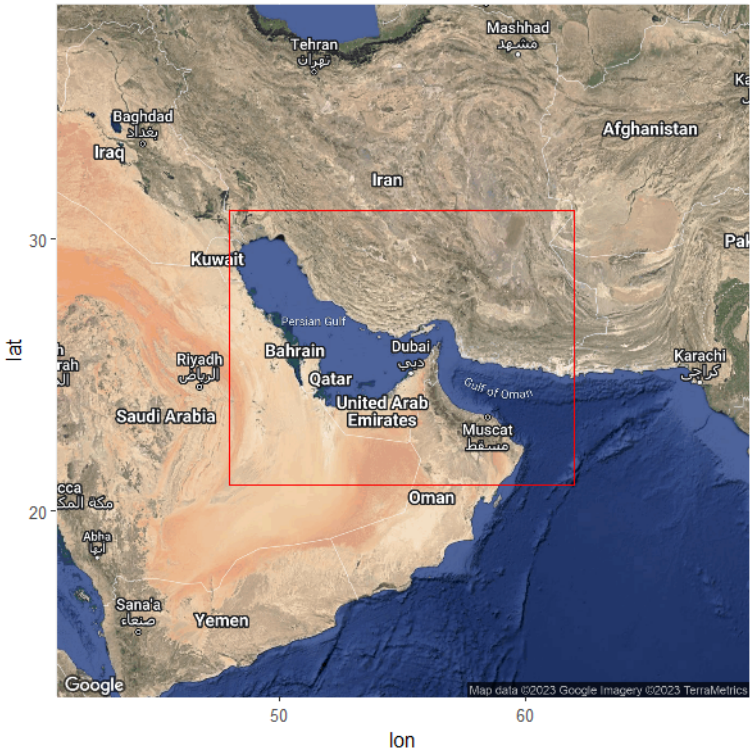

Figure 3.2. Geographic map of research area of interest. Only AIS data from within the red rectangle is considered in our research. This map was rendered using the "ggmap" package in R. Source: Kahle and Wickham (2013).

We use this dynamic AIS data as the foundation of our data set as static AIS data is not available for a specific location. Static records are cleaned by removing duplicate records and matched with dynamic records possessing the same unique MMSI value. We remove static and dynamic records without values for time or less than nine digit MMSI values and merge the records, using dynamic records as the base. The merged dataframe includes latitude and longitude coordinates, message type, MMSI, navigational status, turn rate, speed, course, bearing, and date/time stamp from the original dynamic transmission and IMO, call sign, name, ship type, ship use, bow distance, stern distance, port distance, starboard distance, destination, and date of transmission from the original static transmission (refer to Table 1.1 and Table 1.2 for detailed description of AIS fields). Although the following fields may have use in different analyses, navigational statistic, message type, static date of transmission, and call sign are removed as this information is not useful for trajectory prediction. Additionally,

sea state, current, vessel traffic, and other factors will likely change the required heading to travel a specific course. Thus, we remove the heading field as we anticipate course to be a more accurate predictor of vessel trajectory. Lastly, we only remove reports with missing input for key fields including MMSI, geographic coordinates, time stamp, or ship type.

For ease of use, we convert speed from knots to meters per minute. We create a proxy for ship size by calculating the surface area of the ship, using distance across the ship length and width. The ship length is calculated by summing the stern and bow distances, and is multiplied by the ship width, calculated by summing the starboard and port distances. We then remove the four distance columns from our data set. Although later in our process we convert ship coordinates to Universal Transverse Mercator format, for now we keep coordinates in latitude and longitude decimal format for plotting purposes. We conduct preliminary outlier detection by examining values in each data field, removing rows with values clearly outside of reasonable bounds. For example, many speed values exceeded sixty knots for cargo vessels or fell outside of 0 to 360 degrees for bearing. We save the brunt of outlier detection until vessel tracks are defined. We remove reports for all vessel types other than cargo and tanker in order to focus on larger vessels. Finally, we convert the data set into a simple features dataframe using the "sf" package in R (Pebesma 2018). This data structure allows us to combine geometric information and attribute data into a single object, streamlining spatial operations including spherical distance computation and geographic plotting. Our resulting simple features dataframe includes 9,548,734 records.

Our first step in preparing AIS data for further analysis is the construction of vessel tracks. A track is defined as a set of time sequential coordinates representing the voyage of a single vessel. During the six months of records, one vessel may cover thousands of miles while transiting to multiple destinations, anchoring, or halt transmission of AIS data for an extended period of time. We use a simple algorithm to generate vessel tracks that encompass the entirety of a vessel's time sequential coordinates for this period.

3.2 Outlier Detection

As referenced in the introduction, literature highlights the presence of erroneous and missing data in AIS records. We first remove tracks with less than twenty successive reports or with greater than twenty minutes in between reports. We use an algorithm developed by

Koyak (2017) to remove coordinates that indicate unreasonably large distance traveled over the time period between successive reports (Young 2017). We find the reported speed to sometimes not accurately represent the distance traveled by a vessel over the reported time period. In place of the reported speed, we substitute a computed speed using the distance and time gap from the previous to current report. Similar inconsistencies exist with the reported course, so we substitute a computed bearing using the bearing from previous to current report. To visually verify outlier detection, we plot tracks as geometric paths on a rasterized map of the Persian gulf using the "ggmap" package in R (Kahle and Wickham 2013). Although our methodology does capture most outliers, to conduct an additional sweep we import shapefiles for the Persian Gulf, Arabian Sea, and Gulf of Oman (Marine Regions 2017). Shapefiles provide a geospatial vector data format that stores geographic information including points, lines, or polygons that represent geographic features. The three shapefiles referenced above contain polygons that enclose the respective geographic region. We keep only reports with coordinates within either of the three shapefiles.

3.3 Sub-track Creation

The development of trajectory prediction models requires AIS data shaped into an appropriate format and constrained to a geographic area of interest. Liraz (2018) and Young (2017) select ports of interest and filter AIS data based on vessel tracks that pass within a specified distance of port coordinates to build sub-tracks. Their research mainly focuses on ports in the United States. This methodology works well for major ports adjacent to open ocean and without competing ports nearby. The Persian Gulf, however, contains 66 ports according to the World Port Index data set obtained from the National Geospatial Agency Maritime Safety Information (2019) web page and displayed in Table 3.1.

PORT NAME	COUNTRY	LATITUDE	LONGITUDE	HARBOR SIZE
MINA JABAL ALI	UAE	25.017	55.050	L
BUSHEHR	IRAN	28.983	50.833	M
ABADAN	IRAN	30.333	48.283	M
KHAWR FAKKAN	UAE	25.350	56.383	M
ABU ZABY	UAE	24.500	54.333	M
JABAL AZ ZANNAH/RUWAYS	UAE	24.200	52.700	M
SITRAH	BAHRAIN	26.167	50.667	M
MINA SALMAN	BAHRAIN	26.200	50.633	M
MINA ASH SHUAYBAH	KUWAIT	29.033	48.167	M
MINA AL AHMADI	KUWAIT	29.067	48.167	M
AL BASRAH	IRAQ	30.517	47.833	M
KHORRAMSHAHR	IRAN	30.433	48.183	M
RAS LAFFAN	QATAR	25.917	51.583	M
ZIRKUH PETROLEUM PORT	UAE	25.017	53.000	M
BANDAR ABBAS	IRAN	27.150	56.200	M
KHALIFA BIN SALMAN	BAHRAIN	26.250	50.750	M
BANDAR-E MAHSHAHR	IRAN	30.467	49.183	S
BANDAR KHOMEYNI	IRAN	30.433	49.083	S

Table 3.1. Relevant information for 18 of 66 ports in the Persian Gulf. Source: National Geospatial Agency Maritime Safety Information (2019).

The volume of ports in the Persian Gulf combined with the limited time frame of our AIS data set suggests that a port-focused approach may not provide sufficient number of tracks to build useful NN or RNN models. Preliminary data analysis, including the generation of basic NN and RNN models, confirms this speculation. Instead, we elect to treat the Strait of Hormuz as a virtual gate. We filter AIS tracks that cross within a specified distance of this gate, as depicted in Figure 3.3. For all distance calculations, unless explicitly specified, we compute the haversine distance. The haversine distance represents the shortest distance between two points on the surface of a sphere, such as the Earth. We use the "geosphere" package in R which contains a built-in function to compute haversine distance (Hijmans 2022).

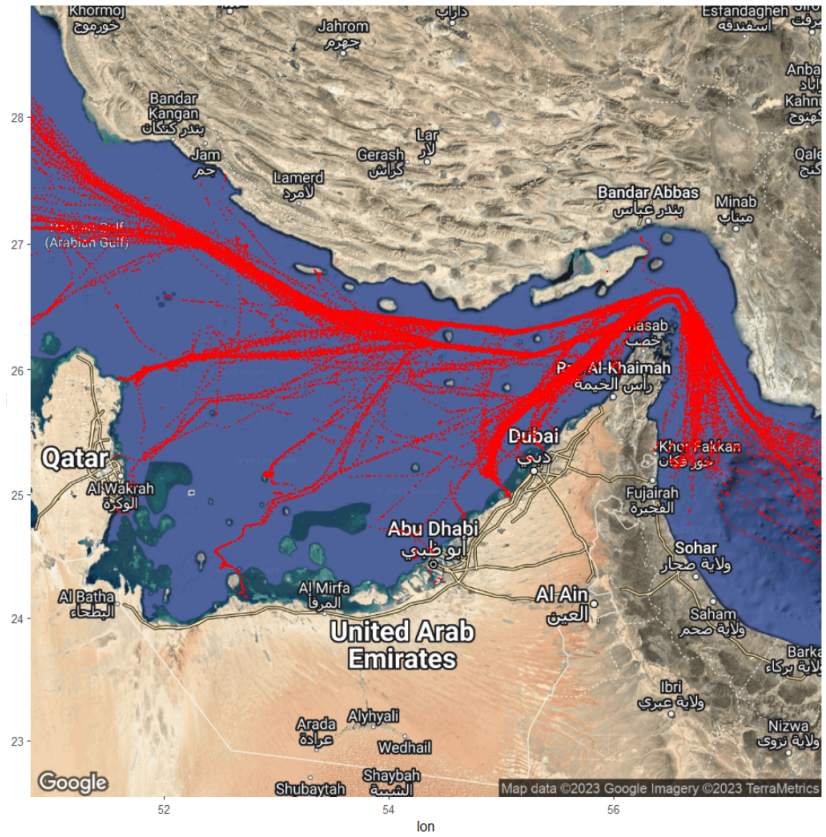

Figure 3.3. Map of vessel tracks that cross the Strait of Hormuz and are considered in our analysis. Each track coordinate is plotted as a small, red circle using the "ggmap" package in R. Source: Kahle and Wickham (2013).

As Figure 3.4 indicates, a vessel transiting across the Strait of Hormuz may make multiple trips between ports or anchorage points. To separate each transit from one POI to another we use an algorithm that extracts these sub-tracks. We label sub-tracks crossing the Strait of Hormuz gate in a west-bound direction as incoming sub-tracks, and those crossing the Strait of Hormuz gate in an east-bound direction as outgoing sub-tracks. We remove outgoing sub-tracks as we are interested in vessels' trajectories within the Persian Gulf, not in open ocean transit.

Figure 3.4. Example of a vessel transiting to multiple destinations in the Persian Gulf. Each AIS transmission for this vessel track is plotted as a small, red circle using the "leaflet" package in R. Source: Cheng et al. (2022).

Both time and distance between successive AIS reports for a sub-track vary. As Nieto-Barajas and Sinha (2014) assert, comparative analysis of time-series data requires equally spaced observations. Young (2017) uses linear interpolation at specified cumulative track distances to standardize AIS data before clustering while Liraz (2018) elects to use a time-based linear interpolation method. In the next section, we attempt both approaches and find that distance-based interpolation produces more definitive clusters and greater average silhouette width. Thus, we interpolate every 15 kilometers of cumulative distance per sub-track until 300 kilometers of total distance.

3.4 Cluster Analysis

Cluster analysis is a statistical technique used in unsupervised machine learning to partition similar data points together based on features or characteristics. In essence, it intends to minimize the dissimilarity between data points while maximizing the dissimilarity between cluster groups. Clustering attempts to discover patters and structures within a data set without prior knowledge of class labels or outcomes. We use a specific clustering method

called Partitioning Around Medoids (PAM) (Kaufman and Rousseeuw 1990). Instead of using the centroid of data points as the medoid in the K-Medoids method, PAM uses the actual data point within the cluster that minimizes the sum of distances to all other points in that cluster as the medoid (Kaufman and Rousseeuw 1990). PAM is robust to outliers and uses interpretable cluster centers but is computationally intensive compared to K-means and other methods. It requires specification of K clusters in advance. We first compute a distance matrix by calculating the distance between respective coordinates for two sub-tracks, summing the distances, and repeating for each combination of sub-tracks in our previously distance-interpolated AIS data. If N represents the number of sub-tracks, then the resulting distance matrix is $N \times N$. The "cluster" package in R includes a PAM algorithm that we use to compute clustering information on our distance matrix for K values from 2 to 30. The To help visually determine the optimally number of clusters, we build a silhouette plot, as depicted in Figure 3.5. This plot displays the silhouette coefficient for each number of clusters, K. The silhouette coefficient measures the average similarity of each data point to its respective cluster compared to other clusters. We aim to choose values of K that maximize the silhouette coefficient and generally fall above 0.5, as suggested by Kaufman and Rousseeuw (1990). Thus, we use $K = 2$ and $K = 3$, and keep corresponding clustering information for use in our models.

Figure 3.5. Silhouette plot indicating silhouette width for clusters sized 2 to 30. This plot suggests that two or three clusters are optimal for grouping sub-tracks. This plot was rendered using the "ggmap" package in R. Source: Kahle and Wickham (2013).

Prior research involving trajectory prediction has traditionally clustered spatial data and then predicted motion on a specific cluster of data. For example, Figure 3.6 depicts vessel sub-tracks categorized by membership to one of two clusters. Past research has typically only built models using data from only cluster one or cluster two, not both simultaneously. Other methods have used cluster membership as a predictor for trajectory prediction models. Our approach, that treats the Strait of Hormuz as a virtual gate and only considers Persian Gulf inbound tracks, produces similar initial trajectories for nearly all tracks, as displayed in Figure 3.3. Thus, we propose a Bayesian approach that computes a probability of cluster membership at each successive coordinate point on the trajectory.

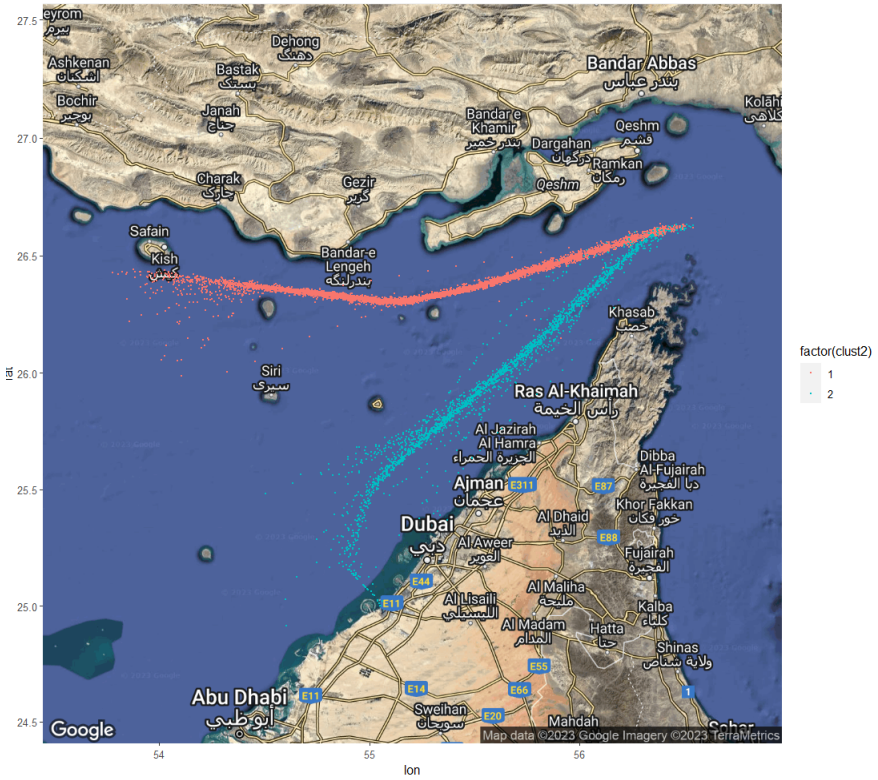

Figure 3.6. Vessel sub-tracks categorized by cluster membership. Coordinates depicted in red are members of the first cluster, and coordinates shown in blue are members of the second cluster. This plot was rendered using the "ggmap" package in R. Source: Kahle and Wickham (2013).

Suppose we have K clusters, each defined for a vector of length $2N$, with N defined as the number of coordinate points. Cluster j has a mean vector μ_j (a $2N$ - vector) and covariance matrix V_j (a $2N \times 2N$ matrix). Suppose we have a randomly observed trajectory that we denote $x = (x_1, ..., x_N)$ which could have come from any of the K clusters.

Before the trajectory starts, we assume something about the prior probabilities that we would assign to the clusters. For example, we might say the trajectory could have come from any of the clusters with the same probability $1/K$, or we might assign different probabilities based on a belief that some clusters are more likely than others. Let $\pi_1^{(0)}, \ldots, \pi_K^{(0)}$ denote these prior probabilities that must sum to 1.

Next, suppose that the vessel is underway and that we are able to see the partial

trajectory $x^{(n)} = (x_1, ..., x_n)$ where n is an integer between 1 and N. The following equation shows how the prior probabilities $\pi_1^{(n-1)}, ..., \pi_K^{(n-1)}$ are updated to produce the new set of posterior probabilities $\pi_1^{(n)}, ..., \pi_K^{(n)}$. It is a recursive process, using Bayes' Rule, starting at $n = 1$, using the previous posterior probabilities as the new prior probabilities. To make the notation simpler, let C_j denote the event that the trajectory came from cluster j:

$$\pi_j^{(n)} = P(C_j|x^{(n)}) = \frac{\pi_j^{(n-1)} \cdot p(x_n|C_j, x^{(n-1)})}{\sum_{i=1}^{K} \pi_i^{(n-1)} \cdot p(x_n|C_i, x^{(n-1)})}, j = 1, \ldots, K. \tag{3.1}$$

We assume that, conditional on the cluster, the trajectory vectors have multivariate normal distributions with means and covariance matrices relevant to that cluster. As x_n is a two-vector of longitude and latitude, $p(x_n|C_j, x^{(n-1)})$ is then understood to be a bivariate normal density conditional on the cluster being C_j and conditional on all previous measurements of the trajectory. We find the conditional mean (two-vector) and covariance matrix (2×2) to evaluate this density and use these quantities as follows.

Let $\mu_{j,n} = (\mu_{j,n,lon}, \mu_{j,n,lat})$ be considered as a row vector, containing the expected values of longitude and latitude of observation n for cluster C_j. Let $V_j^{(n-1)}$ represent the covariance matrix of the previous observation $(n-1)$ for cluster C_j. Let $W_j^{(n)}$ represent the $(2n-2)\times 2$ matrix in the upper-right of matrix $V_j^{(n)}$ and let $(W_j^{(n)})^T$ represent the $2\times(2n-2)$ matrix in the lower-left of matrix $V_j^{(n)}$. Let Q_n represent the 2×2 matrix in the lower-right of matrix $V_j^{(n)}$.

$$\mu_j^{(n)} = \begin{bmatrix} \mu_{j,1} \\ \mu_{j,2} \\ \vdots \\ \mu_{j,n} \end{bmatrix} \qquad V_j^{(n)} = \begin{bmatrix} V_j^{(n-1)} & W_j^{(n)} \\ (W_j^{(n)})^T & Q_n \end{bmatrix} \tag{3.2}$$

Conditional on $x^{(n-1)}$ and C_j, x_n has a bivariate normal distribution with the following mean vector and covariance matrix:

$$\mu_j^{(n|n-1)} = \mu_{j,n} - (W_j^{(n)})^T (V_j^{(n-1)})^{-1} (x^{(n-1)} - \mu_j^{(n-1)})$$
$$V_j^{(n|n-1)} = Q_n - (W_j^{(n)})^T (V_j^{(n-1)})^{-1} W_j^{(n)} \tag{3.3}$$

We implement the algorithm above assuming equal initial probability of membership to each cluster for all trajectories. As displayed in Table 3.2, the algorithm computes a percentage probability that sub-track i is a member of cluster 1, cluster 2, and cluster 3 at each standardized coordinate distance.

		Cluster Probability % at Trajectory Coordinate				
Track	Cluster	Coord1 (0km)	Coord2 (15km)	Coord3 (30km)	Coord4 (45km)	Coord5 (60km)
1	1	33.33	71.12	90.04	97.25	100.00
1	2	33.33	28.88	9.96	2.75	0.00
1	3	33.33	0.00	0.00	0.00	0.00

Table 3.2. Example of the output produced by our Bayesian cluster membership algorithm.

For example, at a coordinate 15 kilometers along the trajectory of sub-track 1, there is a 71.12 percent probability that sub-track 1 is a member of cluster 1. We hypothesize that this algorithm produces a more accurate indication of cluster membership at each coordinate point in a trajectory. We keep the cluster membership probabilities to use as features in NN and RNN models.

3.5 Introduction to Recurrent Neural Networks

NNs attempt to take inputs like positional coordinates and vessel speed and predict a future position of a vessel. Each NN layer consists of connected perceptrons, otherwise known as neurons, and the model adjusts the weights of connections between neurons during training in order to minimize loss (Goodfellow et al. 2016). NN models typically use a version of stochastic gradient descent to optimize loss, and may use a variety of loss functions including mean absolute error, mean squared error or a customized loss function. NNs are usually described by the depth and type of their layers. For example, deep NNs tend to have more than three hidden layers, or layers between the input layer and output

layer (Goodfellow et al. 2016). Other types of NNs include convolutional NNs, designed to process grid-like data contained in images. For further information on NNs, see books by Goodfellow et al. (2016) and Zhang et al. (2021).

RNNs have similar structure but are more well suited to time-series information. These networks not only take in current information to make predictions, but also past information that may help capture relationships between past, current, and future information. In the context of trajectory prediction, RNNs may use past positional coordinates in addition to the current vessel position, to better predict future trajectories.

Neural network models are capable of learning complex, non-linear relationships within trajectory data that are difficult for manual feature engineering to produce. Furthermore, RNNs by nature capture time-related dependencies in trajectory data and predict future positions with greater accuracy. Although neural network models may require significant compute power and extensive hyperparameter tuning, they present a viable option for trajectory prediction models.

Some research methods establish points of interest and build routes from one to another, assessing vessel trajectory based on similarity to these preestablished routes. Although useful for vessels strictly exhibiting normal behavior in adhering to preestablished routes, they may prove overly rigid for vessels exhibiting atypical behavior. Other research methods build RNNs on a single cluster of trajectories with reasonable success. These models may be useful when trajectories quickly branch out into defined channels but may not generalize well to multiple clusters of trajectories.

We propose a method that builds RNN models with clustering information included as features. Our RNN model consists of at least one LSTM layer that aims to capture long range dependencies in our time-series sequence. A common challenge in training NNs with long sequences is the vanishing gradient problem. Gradients used to update the weights of a neural network become very small during backpropagation, causing optimization to take prohibitively long (Bengio et al. 1993). Conversely, the exploding gradient problem may also occur when training a long sequence. In this case, the gradient grows exceedingly large during backpropagation, leading to numerical instability and lack of convergence. LSTM layers are derived from the concept of creating paths through time that have non-vanishing and non-exploding derivatives (Goodfellow et al. 2016). LSTM layers contain multiple

gates that determine what information should be discarded from prior cell states and what new information should be added to the cell state, as depicted in Figure 3.7 and noted by Zhang et al. (2021). The updated cell state reflects the "forgotten" and "added" information, allowing the network to keep gradients from vanishing or exploding.

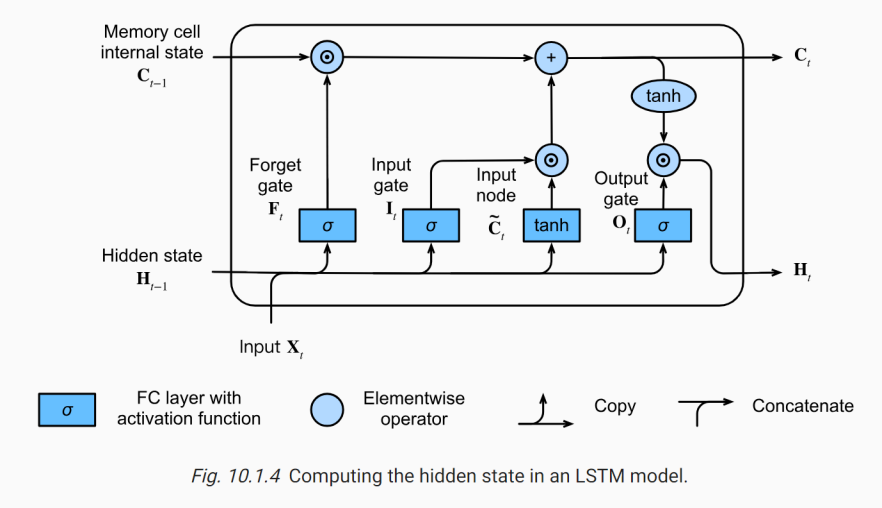

Figure 3.7. LSTM layer basic diagram. Source: Zhang et al. (2021).

Although our clustering approach relies on distance-interpolated AIS reports, we feed both time-interpolated (5 min intervals) and distance-interpolated (15 kilometer intervals) AIS data into our models and find that time-interpolated reports perform slightly better. In the case of the time-interpolated AIS data, we interpolate cluster membership probability, keeping in mind that our most definitive clusters came from distance-interpolated AIS data.

3.6 RNN Model Approach

In this section, we describe in detail our approach to shaping input representation, model architecture, and hyper-parameter selection and tuning for our RNN model.

3.6.1 Input Structure

Our model input consists of time-series dynamic and static data for each vessel sub-track. We use distance and time interpolated sub-track data in our models and scale numeric inputs to mean zero and unit variance, so that all numeric input data ranges from $[-1, 1]$

as recommended by Zhang et al. (2021) except for bearing. Bearing is scaled from 0 to 2π. This step ensures that our models converge faster, as NNs are sensitive to the scale and distribution of input data.

We shape input data to include a window of dynamic information for each point. For example, if i represents the current data location, a window size of five would include the latitude and longitude coordinates, and in some models the speed and bearing for i, $i-1$, $i-2$, $i-3$ and $i-4$ locations as input to the model. This recurrent input window allows the model to better forecast turns and speed changes.

In addition to the input window described above, some models include the average sub-track speed and either cluster membership probability for each of the three clusters or cluster membership. This information is only included for the current trajectory location i. The dynamic data in totality includes at minimum $2N$ features for models only including coordinate windows, and at maximum $4N+4$ features.

We include ship size, MMSI, and ship type as static input. We partition ship size (square meters) into 12 intervals of same length (0,2000],. . .,(22000,24000]. Figure 3.8 portrays a histogram of ship sizes before we partition into intervals.

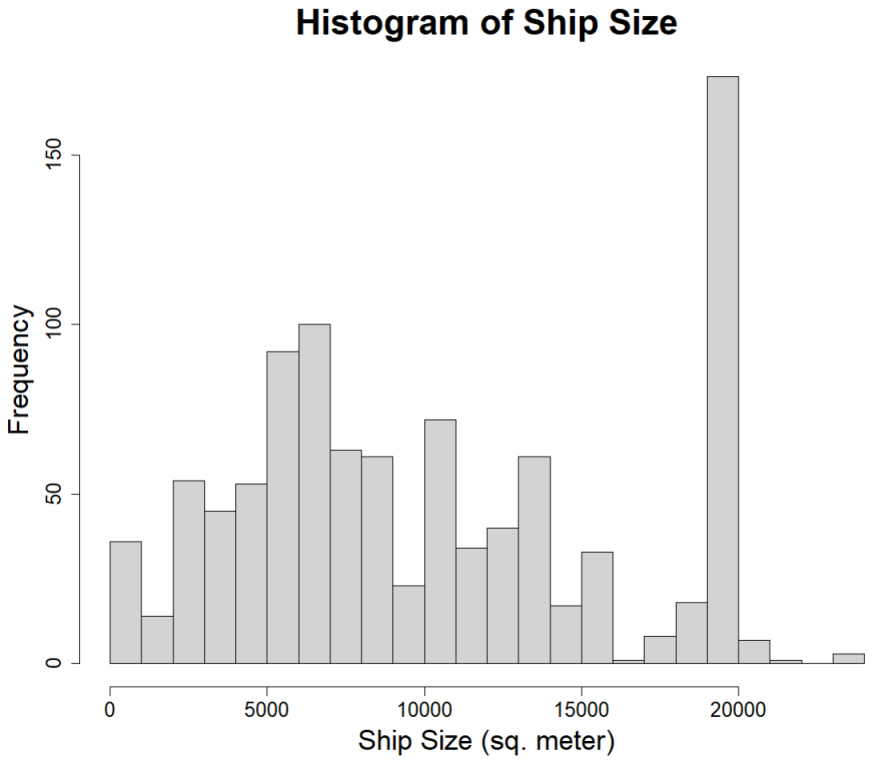

Figure 3.8. Histogram of ship size for inbound vessels crossing the Strait of Hormuz.

Ship type consists of only cargo and tanker vessels, and values for MMSI are unique for most sub-tracks. These three features are categorical and thus do not have a natural order among classes. As NNs require numerical inputs, we use a method called one-hot encoding to transform categorical features to numerical without creating a natural order (Zhang et al. 2021). We build a vector with as many components as classes and set the value to 1 to denote membership to that particular class. For example, a one-hot vector for a cargo ship would be represented by a (1,0) vector while a tanker would be represented by a (0,1) vector. Next, to reduce the complexity and dimensionality of this sparse one-hot encoded data, we use an embedding layer. The corresponding embedding size for each feature is listed in Table 3.3. The embedding layer, combined with dynamic information, constitutes the input data we feed into our models.

Feature Name	Number of Values	Embedding Size
MMSI	472	8
Ship Size	12	4
Ship Type	2	2

Table 3.3. Feature Embedding Sizes for Static Data

3.6.2 Model Design

Past research by Liraz (2018), Li et al. (2019), and Wang et al. (2022) provides us with reasonable starting points for model design considering the number of sub-tracks in our AIS data. In general, more elaborate tasks require deeper networks, but we elect to start with basic models and slowly add complexity. Most past research involves a classification approach that either assigns probabilities to a set of distance and bearing categories or to small geographic subsections. Instead, we use a regression approach that directly predicts latitude and longitude at a fixed time period in the future. Our models that predict the change in latitude and longitude from the current AIS report are significantly more accurate than those that directly predict latitude and longitude.

Although we built deeper models, our models with lowest median miss distance consisted of one LSTM layer and one dense layer, both immediately followed by dropout layers as depicted in Figure 3.9. We use a form of weight decay in our LSTM, dense, and output layers called L2 regularization, commonly known as ridge regression, to introduce a penalty term that helps prevent model overfitting (Zhang et al. 2021). We experiment with different values of the regularization parameter that control the influence of the penalty term and with different activation functions in our dense layer, obtaining best results using a Rectified Linear Unit (ReLU) activation function. These functions serve to introduce non-linearity to the networks computations (Zhang et al. 2021). We also vary the number of units in our LSTM and dense layers.

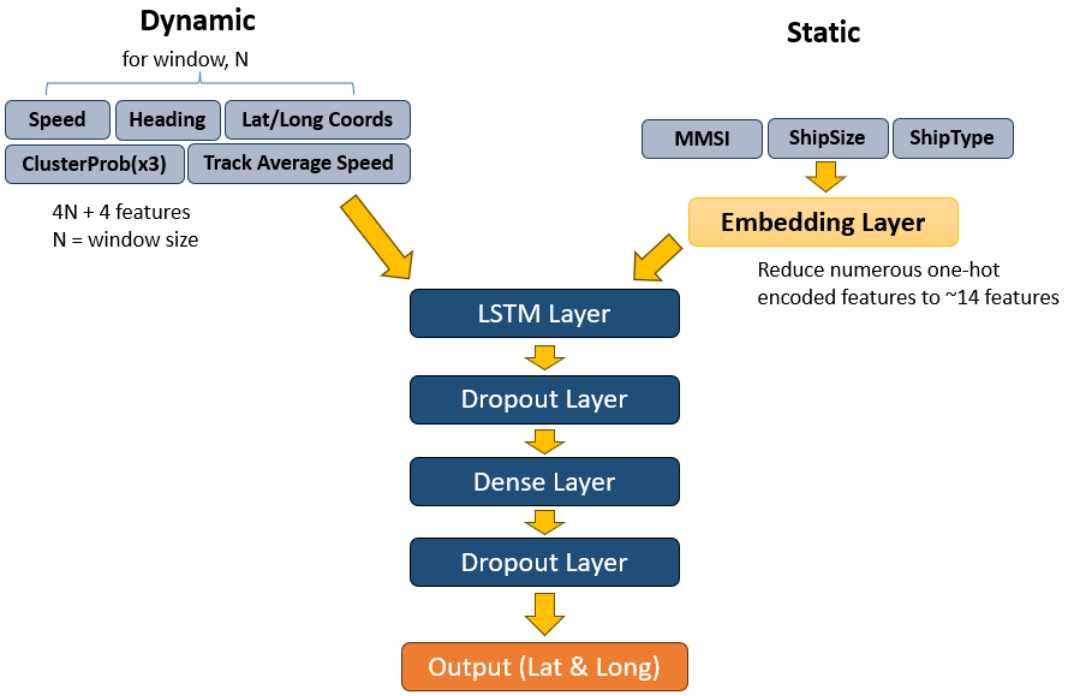

Figure 3.9. RNN regression model architecture. Hidden layers are colored dark blue, output layer is colored orange, embedding layer is colored light orange, and features are colored light blue.

Optimizers and loss functions play a crucial role in the efficient development of RNN models. Based on past trajectory prediction research, we experiment with several optimizers and find that Adam converges our training the fastest and produces the models with highest accuracy. Adam, developed by Kingma and Ba (2017), combines momentum, Adagrad, and RMSProp techniques into an efficient optimization algorithm (Zhang et al. 2021). Although no optimizer is superior in all scenarios, Adam maintains adaptable per parameter learning rates and tracks gradient history to reduce the chance of large weight updates that cause overfitting while still accelerating convergence. Because we intend to minimize the distance between predicted and actual points, we use a customized loss function that computes the distance between points. We use the haversine distance formula, as indicated in Equation 3.4 which uses an approximation of the earth's radius to compute the distance between points in a spherical coordinate system (De Luca 2022).

Let R represent the radius of the earth in meters. The second coordinate, in our case

the predicted coordinate, is represented by $(Lat_2, Long_2)$ and the first coordinate, the actual coordinate, is represented by $(Lat_1, Long_1)$.

$$2R \cdot arcsin\left(\sqrt{\sin^2\left(\frac{Lat_2 - Lat_1}{2}\right) + \cos(Lat_1) \cdot \cos(Lat_2) \cdot \sin^2\left(\frac{Long_2 - Long_1}{2}\right)}\right). \quad (3.4)$$

Although there are more accurate ways to represent distance, as the earth is not a perfect sphere, the haversine distance strikes a balance between accuracy and computational efficiency.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Model Results

We first discuss our method of training models and then present results and analysis. Our analysis includes the results of RNN models built using westbound vessel sub-tracks from the Strait of Hormuz into the Persian Gulf. We assess model accuracy by the mean, median, and standard deviation of miss distance, computed by the haversine distance between predicted and actual vessel locations, at future time intervals of 45 minutes and 120 minutes.

4.1 Model Training

Our models predicting 45 minutes in the future use a data set that includes 638 sub-tracks, while our models predicting 120 minutes into the future use a data set that includes 611 sub-tracks. Both data sets are divided into approximately 70 percent for training, 15 percent for validation, and 15 percent for testing. We use the training set to modify model weights after each epoch, and use the validation set to confirm that the recently updated weights improve model performance. We implement a stop condition while training that halts training if the validation loss does not improve over five consecutive epochs. This measure resets the model weights to the model associated with the lowest validation loss over the last five sets, allowing us to decrease the learning rate and prevent overtraining. We set aside the test set of sub-tracks for use after each model is complete, helping us test our model's accuracy on data it has not seen before.

Due to the varying number of reports in our sub-track data, our RNN input requires an additional data reshaping step. Keras sequential models train on one batch at a time, before updating layer weights, and continuing training on the next batch. Each batch includes a specified number of tracks, and all tracks must have the same length. To deal with this challenge, we implement a basic algorithm that first orders sub-tracks from shortest to longest. A batch of n sub-tracks is taken and all sub-tracks are truncated to the length of the smallest track in the batch. The algorithm then skips j sub-tracks forward and continues this process until all sub-tracks are reshaped. We experiment with different values of n and

j and find best results with smaller values of both. Low n and j values ensure that we do not over-truncate long tracks and lose valuable training data.

4.2 Hyperparameter Selection

The performance of machine learning models largely depends on hyperparameter selection (Zhang et al. 2021). Proper configuration of hyperparameters ensures that optimization solvers converge to a local minimum but do not overfit training data. Some hyperparameters may be explicitly defined, including number of layers, units in each layer, type of activation function in each layer, loss function, and optimizer. However, no generalized method exists to optimally select hyperparameters as users often choose default values. Instead, hyperparameter optimization aims to automate the hyperparameter selection process by first finding small sets of well performing configurations. Once this set is determined, then a user may implement hyperparameter configurations in search of an optimum model (Zhang et al. 2021). Depending on the size of data, this process may take days or even weeks.

For the purposes of this thesis, we leverage past research to narrow down hyperparameters into reasonable ranges, as depicted in Table 4.1. We then manually implement combinations of hyperparameters within these sets in search of models that minimize loss.

Hyperparameter	Value Range	Optimal
Coordinates	Yes/No	Yes
Add'l Dynamic Data (Bearing, Vel, AvgVel)	Yes/No	Yes
Cluster Membership	Yes/No	No
Cluster Probability	Yes/No	Yes
Static Data (MMSI, ShipSize, ShipType)	Yes/No	No
Window Size	[5,20]	10
LSTM Layers	[1,5]	1
Dense Layers	[1,5]	1
Dropout Layers	[1,10]	2
LSTM Layer Units	[10,1000]	250
Dense Layer Units	[10,1000]	150
Optimizer	RMSProp, SGD, Adagrad, Adam	Adam
Dense Layer Acivation Function	Rectified Linear, None	Rectified Linear
Kernel Regularizer	L1, L2	L2
Dropout Rate	[0,0.1]	0.01
Kernel Regularizer Penalty Rate	[0,0.1]	10^{-4}
Loss Function	Distance, MAE, MSE,	Distance

Table 4.1. Hyperparameter ranges and optimal values.

We find that a single LSTM layer and dense layer provides an appropriate model architecture to capture the complexity of predicting vessel trajectories. Models with multiple LSTM and dense layers performed poorly and required substantially more training time. As general practice, we build models with and without dropout layers, but only implement them after an LSTM or dense layer. We find that models without dropout layers capture noise in the training data and have very poor performance. We use several optimizers when fitting our models and find that Adam converges faster and builds more accurate models.

RMSProp and Adagrad produce some viable models but do not converge in a timely manner. We find best results using a rectified linear activation function in the dense layer, and using L2 kernel regularization in LSTM, dense, and output layers. We find best results using a simple haversine distance loss function compared to mean absolute error or mean squared error.

4.3 Results

We compute the miss distance between each predicted and actual value and use mean, median, and standard deviation of miss distance as metrics for model performance. Better performing models tend to share similar characteristics regardless of predicting locations 45 minutes or 120 minutes in the future.

4.3.1 Predicting Future Locations at 45 Minutes

We first build models to predict locations 45 minutes in the future. Table 4.2 describes the results of our best models.

#	Inputs	Window Size	LSTM layers	Dense Layers	Median Miss Distance [meters]	Mean Miss Distance [meters]	Standard Deviation [meters]
1	Coordinates	10	1x250	1x150	419	1741	4003
2	All Dynamic	10	1x250	1x150	279	865	2092
3	All Dynamic & ClustProb	10	1x250	1x150	264	837	2089
4	All Dynamic & Cluster	10	1x250	1x150	291	1195	3887
5	All Dynamic & Static	10	1x250	1x150	770	1509	3657
6	All Dynamic	10	1x400	1x250	352	1102	2855
7	Coordinates	20	1x250	1x150	838	1943	4197
8	All Dynamic	20	1x250	1x150	665	1642	3809
9	All Dynamic & ClustProb	20	1x250	1x150	613	1554	3744

Table 4.2. Test set results for 45 minute predictions of different RNN model configurations.

Our most accurate models in terms of median and mean miss distance incorporate all dynamic data. Models including cluster probability have slightly decreased median miss distance, mean miss distance, and standard deviation, performing better than models that include cluster membership. We find that models using only coordinates, like model 1, perform far worse than all other models.

The miss distance histograms depicted in Figure 4.1 and Figure 4.2 reveals generally lower miss distances for model 2. However, both models have several outliers in the range of 4400 to 6000 meters. These outliers increase the mean miss distances of both models despite most miss distance falling below 400 meters.

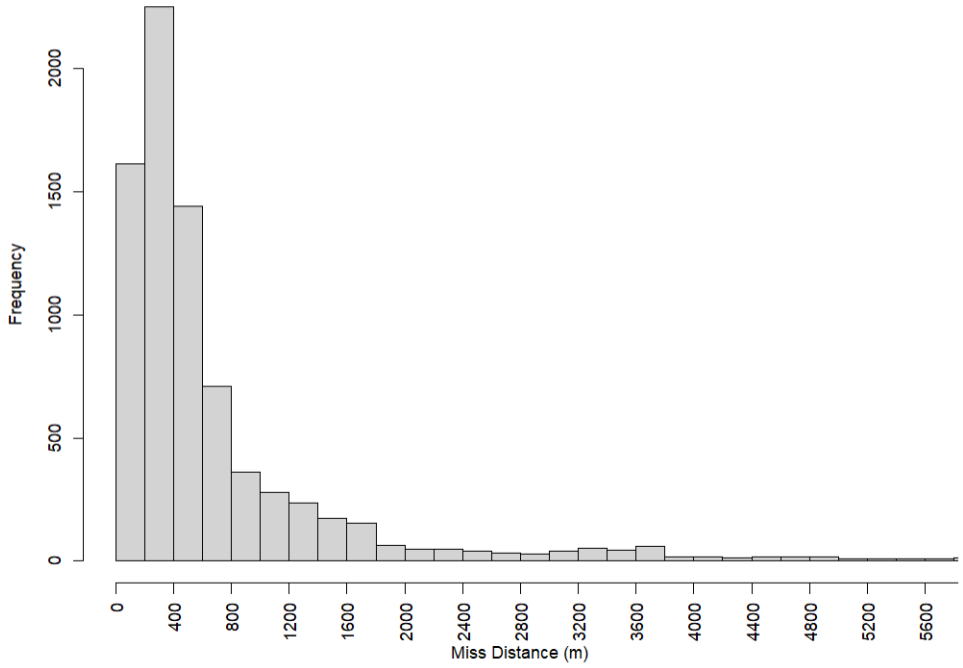

Figure 4.1. Histogram of 45 minute miss distances for model 1.

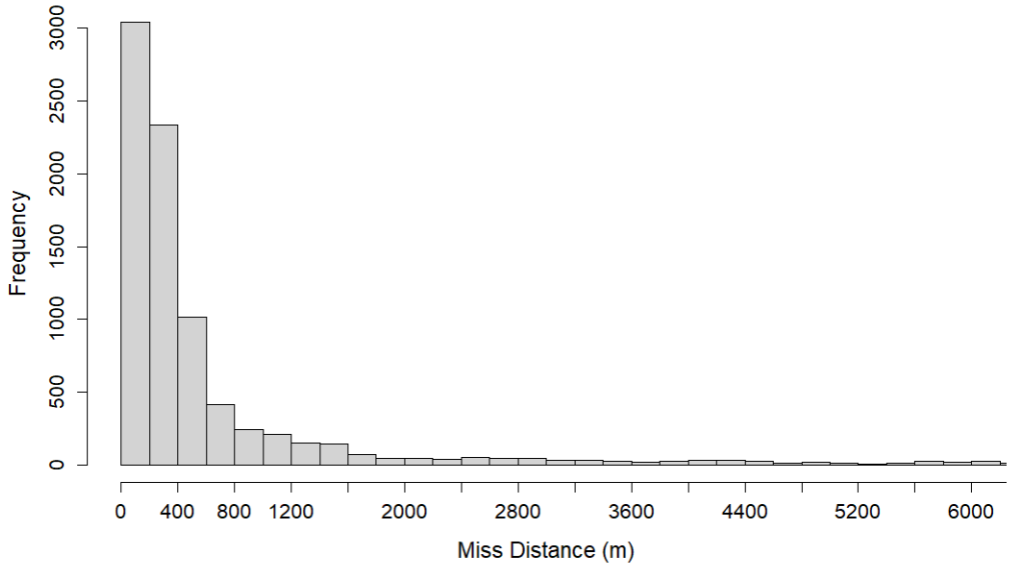

Figure 4.2. Histogram of 45 minute miss distances for model 2.

Better performing models share similar hyperparameter characteristics. We find that a window size of 10, which uses 10 coordinate, velocity, and bearing reports over the last 50 minutes of a vessel trajectory to predict a single future location, performs better than smaller and larger window sizes. Models with larger window sizes tend to underestimate speed and bearing changes, while models with smaller window sizes tend to overestimate speed and bearing changes. We find that window size has a greater effect on model performance compared to other hyperparameters and hypothesize that the optimal window size will change as vessels sail closer to ports or anchoring locations.

Additionally, models with 250 LSTM layer units and 150 dense layer units perform better than models with differing units, although we generally find that at least 50 and 25 units for LSTM and dense layers respectively are needed to build a workable model. Models with between 200 and 400 LSTM units and with 100 to 250 dense units perform similarly.

We find that including embedded static data in our models decreases performance and attribute this to three possible reasons. First, our data only includes AIS reports from a six month period. Out of the over 600 sub-tracks used to train, validate, and test our models, only 9 were traveled by the same vessel. Thus, the information provided by MMSI to the model is limited at best and detrimental at worst. Using data over a multi-year time period would likely result in vessels contributing multiple sub-tracks and allow for the model to identify vessels that travel to the same ports during each trip into the Persian Gulf. Additionally, due to the low number of sub-tracks in our data set, we limited sub-tracks to a length of 300 kilometers in order to use the maximum amount possible. Most sub-tracks were not long enough to reach the vessel's desired destination, likely a port or anchoring location. We originally anticipated the size and type of vessel to limit its ability to use certain ports, but sub-track length did not allow for this information to be relevant. Although we examine different embedding layer sizes within reasonable range of past work, we do not produce a model with better results. We hypothesize that the range of embedding values used may not appropriately represent the static data.

4.3.2 Predicting Future Locations at 120 Minutes

Table 4.3 displays the results of our best performing models that predict locations 120 minutes in the future.

#	Inputs	Window Size	LSTM layers	Dense Layers	Median Miss Distance [meters]	Mean Miss Distance [meters]	Standard Deviation [meters]
1	Coordinates	10	1x250	1x150	2179	4131	6834
2	All Dynamic	10	1x250	1x150	796	1997	4993
3	All Dynamic & ClustProb	10	1x250	1x150	749	1686	4555
4	All Dynamic & Cluster	10	1x250	1x150	778	1923	4952
5	All Dynamic & Static	10	1x250	1x150	1885	3408	6347
6	All Dynamic	10	1x400	1x250	991	2139	5028
7	Coordinates	20	1x250	1x150	2871	4778	6597
8	All Dynamic	20	1x250	1x150	1395	2719	5834
9	All Dynamic & ClustProb	20	1x250	1x150	1301	2685	5911

Table 4.3. Test set results for 120 minute predictions of different RNN model configurations.

Our best models include all dynamic data and cluster probability, similar to results from our 45 minute prediction models. Our coordinate only model has much higher median and mean miss distance than models incorporating dynamic data, as seen in Figure 4.3 and Figure 4.4. This is consistent with results from our 45 minute prediction models, indicating that the additional information provided by speed, bearing, and average sub-track speed is unable to be drawn out from coordinates alone in our models. The importance of dynamic information is exacerbated with 120 minute prediction models, as the median miss distance nearly triples.

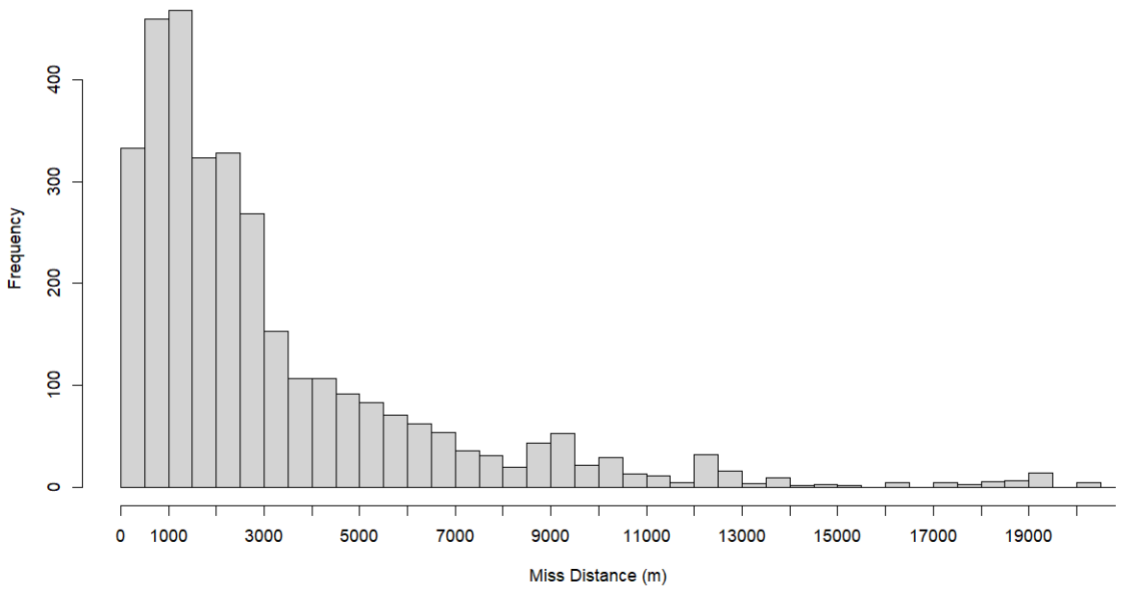

Figure 4.3. Histogram of 120 minute miss distances for model 1.

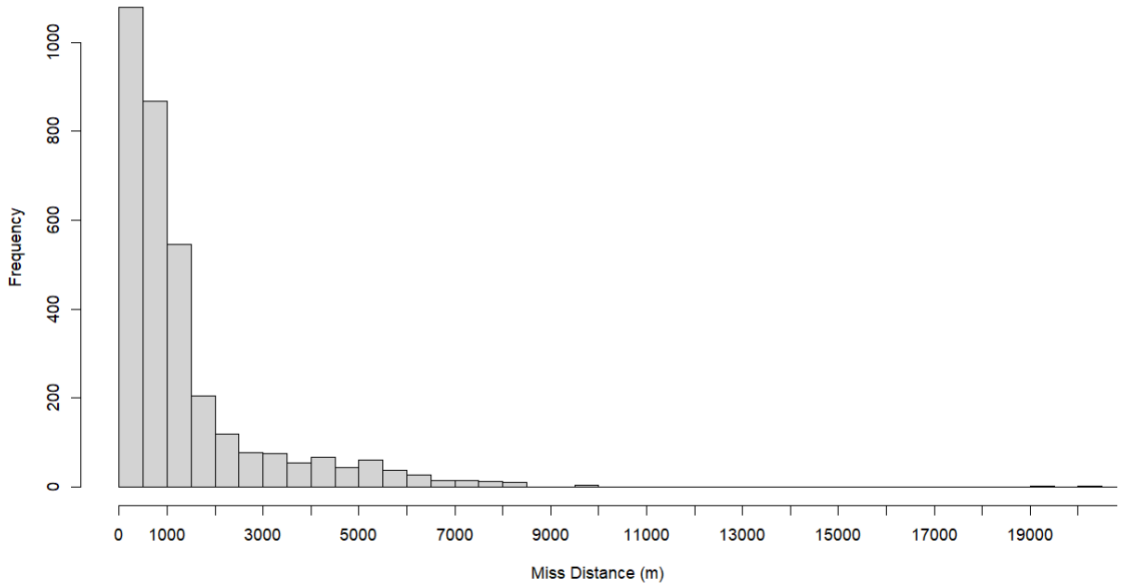

Figure 4.4. Histogram of 120 minute miss distances for model 2.

Furthermore, we see slightly better results in models that use dynamic data and cluster membership compared to models that only use dynamic information. We conclude that for longer prediction horizons, cluster membership adds additional information not captured solely by dynamic data.

For predicting 45 minute and 120 minutes in the future, we find that our optimal models use very similar hyperparameters and identical model design. Given that in most sub-tracks vessels travel over 50 kilometers in a 120 minute period, we find a median miss distance of less than a kilometer to be very accurate. We hypothesize that part of the high model accuracy stems from the similarity of tanker and cargo ship routes when entering the Persian Gulf. Accordingly, most of our miss distance outliers come from predictions near ports, where vessels often make abrupt turns.

Figure 4.5 depicts the actual trajectory (blue) of a vessel nearing Abu Dhabi, UAE and completing a near 180 degree turn compared to the associated 45 minute predicted trajectory (red) made by model 3. This plot depicts one of the largest miss distances of our test set, and is consistent with other outlier miss distances in our sub-track predictions. Because our RNN models use a window of dynamic information in making predictions, our models tend to lag in predicting turns, especially abrupt turns. We discuss potential methods of improving model performance near ports in the next chapter.

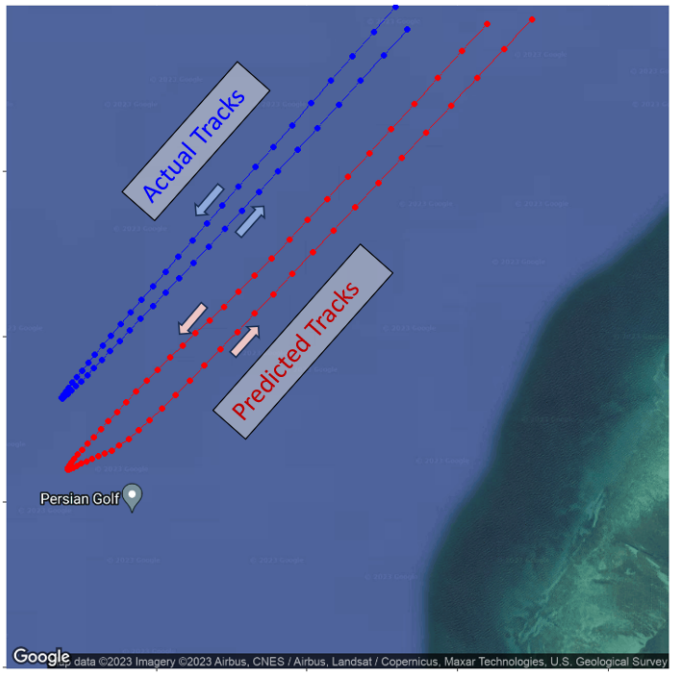

Figure 4.5. Example of abrupt vessel turn in 45 minute predictions (red line/points) and actual locations (blue line/points). This plot was rendered using the "ggmap" package in R. Source: Kahle and Wickham (2013).

As all sub-tracks in our training, testing, and validation sets originate from the Strait of Hormuz and proceed into the Persian Gulf towards ports, we hypothesized that miss distance would increase as the vessel progressed on its trajectory. Figure 4.6 displays the miss distance by the scaled sub-track coordinate number, with earlier coordinates on a sub-track displayed on the left side of the plot. A fitted linear regression (in red) indicates that the miss distance does tend to increase as the vessel continues motion, confirming our hypothesis.

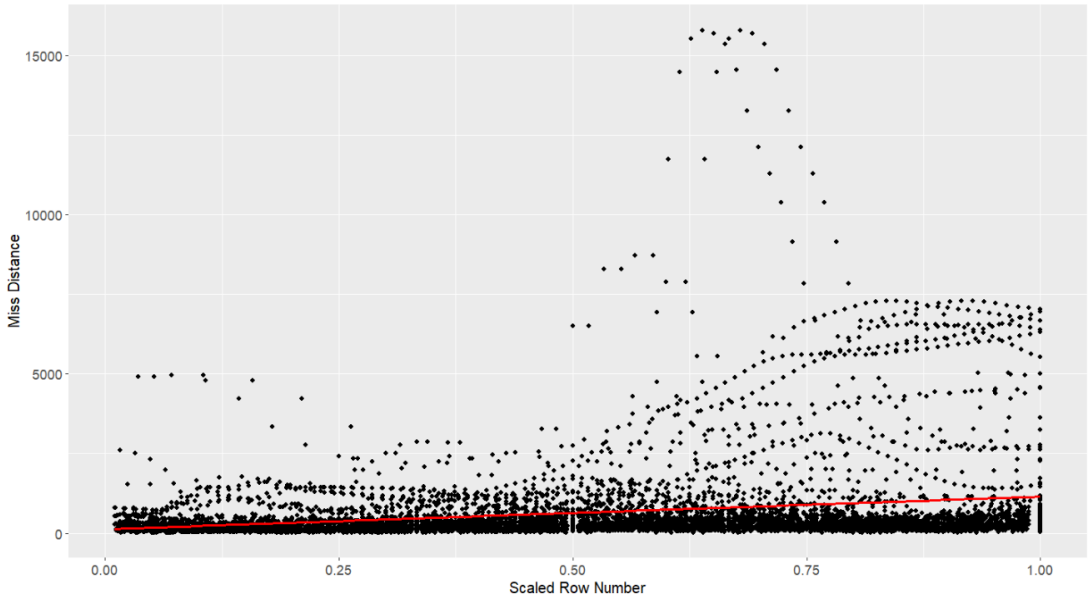

Figure 4.6. Plot of miss distance vs scaled sub-track coordinate number. Each track coordinate miss distance is denoted by a small black circle. The fitted linear regression is depicted by a solid, red line.

CHAPTER 5:
Summary and Recommendations

In this chapter, we summarize our model development efforts and propose future work that may enhance vessel motion prediction.

5.1 Summary

We propose several recurrent neural network models, relying on LSTM and dense layers, to predict vessel trajectories at 45 and 120 minutes in the future in the Persian Gulf. Models trained on a window of coordinates and dynamic information, including speed, bearing, and average speed, make the most accurate predictions across both time prediction points. Additionally, we find probability of cluster membership to aid model accuracy and outperform models that have binary cluster membership as a feature. We find that these models predict future location on our test set with a median miss distance under 400 meters when predicting ahead 45 minutes. At a speed of 15 knots, this is roughly equivalent to less than 3 percent error compared to distance traveled. We find similar success in predicting ahead 120 minutes.

Despite relative accuracy in predicting trajectories in well-defined channels, our models struggle to predict abrupt movement that is especially prevalent near major ports. We find most miss distance outliers at the end of vessel sub-tracks, as miss distances tend to increase as a vessel progresses on its journey to a destination port. Furthermore, the limited geographic area of our training, testing, and validation data suggest that our models may not generalize well to vessel trajectories outside of the Persian Gulf.

5.2 Recommendations and Future Work

Though our models predict vessel trajectories with reasonable accuracy within a specified geographic area of interest, we propose several data and modeling recommendations to improve model accuracy and generalization.

Future work should obtain more recent AIS data that spans multiple years. AIS

adoption and compliance has increased since 2014, and recent data is likely to be more numerous, thorough, and accurate. Obtaining multi-year data would enable the investigation of seasonal relationships between vessel trajectories and would improve the utility of static information as vessels would likely return to similar destinations over a longer time period. Furthermore, researchers should integrate port-specific information including distance to major ports, anchorage locations, and oil rig and field location. Including this information would allow RNN models to decipher the importance of features relative to their proximity to the coast, enabling more accurate models near ports. For example, in open ocean away from major ports, models may treat all coordinates in a window as equally important in predicting motion. As a vessel nears a port, it may rely more on the most recent coordinates as vessel motion is more abrupt near the coastline.

Additionally, future work may use web scraping or other techniques to verify manually entered static AIS reports, and only include verified static information when building models. Although our models did not improve with static information, using verified static information captured over a longer time period may allow for improved model accuracy. More efficient processes for automated data cleaning would enable the use of higher quality data and allow more time for model development. It is likely that we trained, validated, and tested our models on outlier sub-tracks and inadvertently removed sub-tracks that may have been useful.

Future work may develop hybrid models that combine network representation of the geographic area of interest with the RNN approach of our research. These models would better account for known turning points, channels, shoal water, or other geographic features that impact ship movement. Researchers may use a dual destination and trajectory prediction approach, similar to the one proposed by Wang et al. (2022), or use a dual anomaly detection and trajectory prediction approach. The latter would enable better outlier detection prior to model training and cue maritime analysts to vessel behavior worthy of further investigation. In terms of hyperparameter selection, an automated approach would enable the exploration of larger hyperparameter ranges and potentially discover hyperparameter combinations not otherwise explored. Additional features could also be explored including weather, recent key maritime events or economic indicators, which may improve model performance.

List of References

Bay SM (2017) Evaluation of factors on the patterns of ship movement and predictability of future ship location in the Gulf of Mexico. Master's thesis, Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/53021.

Bengio Y, Frasconi P, Simard P (1993) The problem of learning long-term dependencies in recurrent networks. *IEEE International Conference on Neural Networks*, volume 3, 1183–1188, https://doi.org/10.1109/ICNN.1993.298725.

Cheng J, Karambelkar B, Xie Y (2022) *Leaflet: Create Interactive Web Maps with the JavaScript 'Leaflet' Library*, https://rstudio.github.io/leaflet/.

Cutlip K (2017) AIS for safety and tracking: A brief history. Global Fishing Watch. Accessed July 28, 2023, https://globalfishingwatch.org/data/ais-brief-history/.

de Brébisson A, Simon E, Auvolat A, Vincent P, Bengio Y (2015) Artificial neural networks applied to taxi destination problem. arXiv, Ithaca, NY, https://arxiv.org/pdf/1508.00021.

De Luca G (2022) Haversine formula. Baeldung CS. Accessed August 5, 2023, https://www.baeldung.com/cs/haversine-formula.

Decis H, Breton CL (2021) A new phase in the gulf shipping threat? International Institute for Strategic Studies. Accessed July 30, 2023, https://www.iiss.org/online-analysis//military-balance/2021/09/a-new-phase-in-the-gulf-shipping-threat.

Eriksen T, Greidanus H, Delaney C (2018) Terrorism and deterrence policy with transnational support. *Marine Policy* 93 (July), https://doi.org/10.1016/j.marpol.2018.03.028.

Goodfellow I, Bengio Y, Courville A (2016) *Deep Learning* (MIT Press), http://www.deeplearningbook.org.

Hijmans RJ (2022) *Geosphere: Spherical Trigonometry*, https://cran.r-project.org/package=geosphere, r package version 1.5-18.

International Maritime Organization (2023a) Introduction to IMO. Accessed July 10, 2023, https://www.imo.org/en/About/Pages/Default.aspx.

International Maritime Organization (2023b) Ships' routeing. Accessed July 10, 2023, https://www.imo.org/en/OurWork/Safety/Pages/ShipsRouteing.aspx.

Kahle D, Wickham H (2013) ggmap: Spatial visualization with ggplot2. *The R Journal* 5(1):144–161, https://journal.r-project.org/archive/2013-1/kahle-wickham.pdf.

Kaufman L, Rousseeuw PJ (1990) *Finding Groups in Data: An Introduction to Cluster Analysis* (John Wiley & Sons, Inc., New York, NY, USA).

Kingma DP, Ba J (2017) Adam: A method for stochastic optimization. arXiv, Ithaca, NY, https://doi.org/10.48550/arXiv.1412.6980.

Koyak R (2017) Statistical tools for the analysis of automated identification system data. Unpublished manuscript, Naval Postgraduate School, Monterey, CA.

LaGrone S (2023) U.S. sending Marines, more warships to Middle East over Iranian threats. United States Naval Institute. Accessed July 30, 2023, https://news.usni.org/2023/07/20/u-s-sending-marines-more-warships-to-middle-east-over-iranian-threats.

Li W, Zhang C, Ma J, Jia C (2019) Long-term vessel motion predication by modeling trajectory pattern with AIS data. *2019 5th International Conference on Transportation Information and Safety (ICTIS)*, https://ieeexplore.ieee.org/document/8883596.

Liraz SP (2018) Ships' trajectories prediction using recurrent neural networks based on AIS data. Master's thesis, Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/60431.

Marine Regions (2017) Marine gazeteer placedetails: Persian Gulf. Accessed July 16, 2023, https://www.marineregions.org/gazetteer.php?p=details&id=4266.

MarineTraffic (2023) Predictive destinations. MarineTraffic. Accessed July 23, 2023, https://www.marinetraffic.com/en/ais-api-services/detail/vi04/predictive-destinations.

Mascaro S, Nicholson AE, Korb K (2014) Anomaly detection in vessel tracks using Bayesian networks. *International Journal of Approximate Reasoning* 55(1) (January), https://doi.org/10.1016/j.ijar.2013.03.012.

National Geospatial Agency Maritime Safety Information (2019) 2019 Edition World Port Index Shapefile. Accessed April 20, 2023, https://msi.nga.mil/Publications/WPI.

Nieto-Barajas LE, Sinha T (2014) Bayesian interpolation of unequally spaced time series. *Stochastic Environmental Research and Risk Assessment* 29(2) (February), https://doi.org/10.1007/s00477-014-0894-3.

Omholt-Jensen K (2021) AIS and the main categories of AIS challenges. ShipIntel by Maritime Optima. Accessed July 26, 2023, https://www.maritimeoptima.com/blog/ais-and-the-main-categories-of-ais-challenges.

Pallotta G, Horn S, Braca P, Bryan K (2014) Context-enhanced vessel prediction based on Ornstein-Uhlenbeck processes using historical AIS traffic patterns: Real-world experimental results. *17th International Conference on Information Fusion (FUSION)*, https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6916016.

Pallotta G, Vespe M, Bryan K (2013) Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction. *Entropy* 15(6) (June), https://doi.org/10.3390/e15062218.

Pebesma E (2018) Simple Features for R: Standardized Support for Spatial Vector Data. *The R Journal* 10(1):439–446, https://doi.org/10.32614/RJ-2018-009.

R Core Team (2021) *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, https://www.R-project.org/.

Spire Maritime Documentation and Knowledge Base (2023) Messages API. Spire Maritime Documentation and Knowledge Base. Accessed July 27, 2023, https://documentation.spire.com/messages-api/.

Tester KA (2013) A spatiotemporal clustering approach to maritime domain awareness. Master's thesis, Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/37731.

United Nations Statistics Division (2020) Overview of AIS dataset. Accessed July 17, 2023, https://unstats.un.org/wiki/display/AIS/Overview+of+AIS+dataset.

United States Coast Guard (2023a) Automatic identification system overview. United States Coast Guard Navigation Center. Accessed July 24, 2023, https://www.navcen.uscg.gov/automatic-identification-system-overview.

United States Coast Guard (2023b) Types of automatic identification systems (per itu-r m.1371 and iec standards). United States Coast Guard Navigation Center. Accessed July 24, 2023, https://www.navcen.uscg.gov/types-of-ais.

United States Navy (2022) *Chief of Naval Operations Navigation Plan 2022*. Defense Visual Information Distribution Service. Washington, DC, https://www.dvidshub.net/publication/issues/64582.

von Eiff JS (2018) Characterizing ship navigation patterns using automatic identification system data in the Baltic Sea. Master's thesis, Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/58272.

Wang W, Bin J, Zaji A, Halldearn R, Guillame F, Li E, Liu Z (2022) A multi-task learning-based framework for global maritime trajectory and destination prediction

with AIS data. *Maritime Transport Research* 3(100072), https://doi.org/10.1016/j.martra.2022.100072.

White House (2022) *National Security Strategy*. Washington, DC, https://www.whitehouse.gov/wp-content/uploads/2022/11/8-November-Combined-PDF-for-Upload.pdf.

Wolsing K, Roepert L, Bauer J, Wehrle K (2022) Anomaly detection in maritime AIS tracks: A review of recent approaches. *Journal of Marine Science and Engineering* 10(1) (January), https://doi.org/10.3390/jmse10010112.

Yan Z, Song X, Zhong H, Yang L, Wang Y (2022) Ship classification and anomaly detection based on spaceborne AIS data considering behavior characteristics. *Sensors (Basel)* 22(20) (October), https://doi.org/10.3390/s22207713.

Young B (2017) Predicting vessel trajectories from AIS data using r. Master's thesis, Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/55564.

Zhang A, Lipton ZC, Li M, Smola AJ (2021) Dive into deep learning. *arXiv:2106.11342*.

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California