CHAT: Maritime Route Prediction via Conditional Historical AIS Trajectory Data [Scalable Data Science]

Siya Qiu
The Hong Kong University of Science and Technology (Guangzhou)
sqiual@connect.ust.hk

Yihong Luo
The Hong Kong University of Science and Technology (Guangzhou)
yluocg@connect.ust.hk

Jing Tang
The Hong Kong University of Science and Technology (Guangzhou)
jingtang@ust.hk

Qiong Luo
The Hong Kong University of Science and Technology (Guangzhou)
luo@cse.ust.hk

Mauricio Collado
University of California, Santa Barbara
nestorcollado@ucsb.edu

ABSTRACT

The safety of vessel navigation has long been a key concern in the maritime industry, and route prediction is an effective solution to this problem by providing guidance to navigation and identifying route deviation. However, most existing route prediction systems are designed for land transportation, relying on pre-defined road networks and social interactions. In contrast, the ocean is an expansive open space for navigation, where vessels are sparsely distributed and rarely interact with each other. To bridge this gap, we introduce CHAT, a maritime route prediction model leveraging conditional historical trajectory data from the on-vessel Automatic Identification System (AIS). CHAT extracts spatial and temporal information solely from historical and target trajectories to work without pre-defined networks and social interactions. Furthermore, we propose a Bi-directional Temporal Convolutional Network (B-TCN) for feature extraction in CHAT, which convolves in both temporal and reverse temporal order to consider contextual information in both directions. Also, we eliminate inter-element dependency within each layer in CHAT to mitigate prediction error accumulation commonly observed in recurrent neural network models. Finally, as vessel trajectory datasets are rarely available in the public domain, we construct such a dataset consisting of representative navigational scenarios. Extensive experiments on this dataset demonstrate that our model outperforms the state-of-the-art methods by 38.4% in 1-hour Average Displacement Error (ADE) and 34.5% in 1-hour Final Displacement Error (FDE).

PVLDB Reference Format:

Siya Qiu, Yihong Luo, Jing Tang, Qiong Luo, and Mauricio Collado. CHAT: Maritime Route Prediction via Conditional Historical AIS Trajectory Data [Scalable Data Science]. PVLDB, 14(1): XXX-XXX, 2020.
doi:XX.XX/XXX.XX

PVLDB Artifact Availability:

The source code, data, and/or other artifacts have been made available at URL_TO_YOUR_ARTIFACTS.

Proceedings of the VLDB Endowment, Vol. 14, No. 1 ISSN 2150-8097.
doi:XX.XX/XXX.XX

1 INTRODUCTION

The rapid growth of positioning technologies results in abundant location data for moving objects, significantly contributing to the advancement of trajectory analysis [22, 24, 43, 52]. Route prediction is an important application of trajectory analysis, which assists in traffic management and sustainable mobility [18, 39]. In particular, trajectory prediction for maritime transportation can help improve vessel safety and efficiency in the sea, as it allows for early detection of potential collisions, aids in route optimization, and facilitates effective decision-making by ship operators and maritime authorities. However, existing studies primarily focus on vehicle and pedestrian trajectory prediction, and maritime trajectory prediction receives much less attention. Therefore, we aim to bridge this gap and propose effective maritime trajectory prediction methods.

Previous work on maritime trajectory prediction is largely limited to short prediction durations, ranging from minutes to half an hour [15, 38]. This limitation is because the cumulative prediction error often experiences a significant rise as the prediction length increases. As such, generating predictions of longer duration typically required pre-determined locations to reduce the error [32]. However, long-term vessel trajectory prediction for arbitrary locations plays a pivotal role in enhancing maritime situational awareness and facilitating efficient ship allocation in response to global freight demand [23]. Therefore, we propose a generalizable model for long-term maritime trajectory prediction that is applicable across various distinct regions and capable of forecasting for a few hours, while mitigating the accumulation of prediction errors.

Trajectory prediction is extensively studied for vehicle and pedestrian. Traditional vehicle trajectory prediction emphasizes on predicting the driver's driving maneuvers [8, 13, 31, 50], such as going straight or turning left, based on the road network and user preferences. Other vehicle trajectory prediction methods generate long predictions by turning the pre-defined road network into graph-structured data [19, 34, 59]. However, unlike land-based scenarios, maritime navigation is not constrained by a pre-defined road network, which predicting driving maneuvers and creating graph-like network inapplicable.

Vessels traversing the open sea share similarities to pedestrians walking in unobstructed areas. However, vessel traversing time spans are much longer, and their accessing areas are much larger

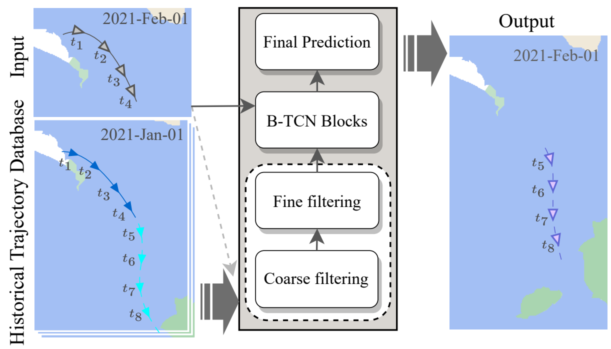

Figure 1: Trajectory prediction with CHAT.

and more sparse. Pedestrian trajectory prediction utilizes social interactions, often represented by inter-personal distances, to predict interactions and how individuals mutually influence route selection for the next few seconds [2, 29, 37, 41, 45]. For instance, the ETH/UCY universal pedestrian trajectory dataset [2, 30] collects data through camera recordings in small areas such as schools and shops, where trajectories typically last for less than a minute. In contrast, maritime vessel trajectories can extend for hours, days, or even longer as they travel between ports or cross oceans. Furthermore, the social interactions used for pedestrian trajectory prediction are usually irrelevant or infeasible for maritime navigation, with vessels sparsely distributed at sea. Considering these differences, we propose to develop a novel trajectory prediction system tailored explicitly for maritime navigation.

To overcome the lack of road networks and social interactions in maritime trajectory prediction, we propose a novel model, namely CHAT, leveraging Conditional Historical AIS (automatic identification system) Trajectories. The core concept of CHAT lies in utilizing the *historical future* in the historical trajectory database to predict the *current future* in the target trajectory. Specifically, searching the historical trajectory database, CHAT identifies the topmost similar historical trajectories to the target trajectory under current observation. These similar trajectories not only reveal the common routing preferences of past travelers [35], but also implicitly provide insight into external environmental factors, such as directions of currents and wind, that may influence routing choices. Consequently, the selected historical trajectories contain environmental context information, the features from the currently observed trajectory, and the features of historical future routes. These selected trajectories will serve as a basis for predicting the current future trajectory of the target vessel. Figure 1 illustrates the workflow of our CHAT model.

Recurrent neural networks (RNNs) are extensively utilized in maritime trajectory prediction. However, RNNs suffer from two primary limitations. First, RNN-based models rely solely on past information to make predictions, which are often impractical in real-world scenarios where decisions are influenced by both the past movement pattern and the dynamic external factors that lie ahead. Second, the sequential nature of RNN predictions renders them susceptible to error accumulation, as each predicted item relies on the previous one. These characteristics can lead to a compounding effect of errors over time.

In order to address these two limitations, we design the Bi-directional Temporal Convolutional Network (B-TCN) as a feature extraction module within our CHAT model. B-TCN is derived from the Temporal Convolutional Network (TCN) [6], which essentially consists of one-dimensional convolutional layers that exclusively zero-pad the sequences at the beginning to prevent any information leakage from the future to the past. In contrast, our B-TCN incorporates both TCN and reverse TCN with zero-padding at the end of sequences. By employing convolution in both directions, B-TCN ensures that each output item is convolved using elements from both the past and future in the historical database. Furthermore, all convolutional output items within the same layer are generated in parallel, effectively mitigating the accumulation of errors.

To evaluate the effectiveness of the CHAT model, we construct a novel vessel trajectory dataset utilizing publicly available data sources provided by the Danish Maritime Authority [4] and the Office for Coastal Management [14]. The two chosen data sources are highly reputable providers of AIS data, which is analogous to GPS data for vehicles. Different from GPS data, AIS samples ship trajectory information at possibly irregular time intervals, so interpolation is necessary when using AIS data. Our AIS dataset encompasses trajectories from both the Atlantic and Pacific Oceans, spanning three distinct types of navigational areas: areas with clear waterways, areas without clear waterways, and general areas. The purpose of identifying these areas is to capture different navigational characteristics. The areas with clear waterways are designed to mimic the conditions on land, where vehicles adhere to a pre-determined road network. In contrast, the areas without clear waterways represent open seas where vessels navigate freely without waterway constraints. Finally, the general areas contain a mix of the first two types, exhibiting characteristics of both areas—with or without clear waterways. Extensive experimental evaluations on this dataset demonstrate that our CHAT model outperforms existing maritime trajectory prediction methods, especially on long-term predictions' robustness.

Our main contributions are summarized as follows.

- We propose a novel maritime route prediction model named CHAT, which predicts trajectory based on similar historical trajectories and the current trajectory under observation. Unlike land trajectory predictions, CHAT does not depend on a fixed road network or the calculation of neighbor distance matrices.
- We introduce B-TCN, which incorporates TCN and reverse TCN layers. This novel approach enables convolutions to be conducted utilizing past and future elements in the historical database, thereby enhancing the extracted features from both directions.
- We create a vessel trajectory dataset by gathering and processing data from two publicly accessible sources. This comprehensive dataset incorporates trajectory data from two oceans, containing three distinct types of area in each ocean.
- We evaluate our proposed method experimentally on the newly created dataset. The results demonstrate that our model surpasses the performance of baseline models regarding trajectory prediction accuracy, and that our method accumulates prediction errors notably slower than baselines on long-term prediction tasks.

2 RELATED WORK

We review trajectory prediction in three primary domains: vehicle, pedestrian, and vessel.

2.1 Vehicle Trajectory Prediction

Vehicle trajectory prediction involves destination prediction and driving maneuver prediction. Destination prediction typically relies on a pre-determined road network or a routable graph. Wei et al. [49] propose using Bayesian inference and a grid representation of the road network's data space to predict destinations. Building on Wei et al.'s work [49], Xue et al. [51] decompose the original trajectory into sub-trajectories comprising two neighboring locations to address data sparsity. Khezerlou et al. [26] introduce a spatio-temporal hybrid model for continuous destination prediction of trajectories. Yang et al. [53] propose using Markov transition matrix to obtain the transition probability between locations and then predicting future locations through Bayesian reasoning.

Driving maneuver prediction refers to predicting the next driving actions of the driver, such as going straight or turning left. Among different models, Long-Short Term Memory (LSTM), a variant of recurrent neural network (RNN), stands out for its ability to effectively handle sequential data, making it one of the most commonly used models for predicting driving maneuvers. Zyner et al. [60] use a three-layer LSTM to predict driving maneuvers. Ding et al. [13] employ LSTM to anticipate maneuvers, and then combine the anticipated maneuvers with relevant contextual information for prediction. Another research [40] suggests adopting an LSTM encoder-decoder structure together with beam search to generate multiple possible future trajectories. However, LSTM alone cannot simultaneously deal with spatio-temporal information.

Despite RNN's proficiency in handling sequential data, some studies suggest combining the strengths of RNN and convolutional neural network (CNN) to better capture spatio-temporal features. Xie et al.'s research [50] uses CNN to extract social interaction features, which are then fed into LSTM for prediction. Another study [12] utilizes LSTM to encode past trajectories, including the ego-vehicle and surrounding vehicles, constructing a social-tension layer to capture interaction features.

Vehicle trajectory prediction typically involves predicting discrete maneuvers or relying on pre-determined road networks. However, maritime trajectory prediction poses a unique challenge as navigational maneuvers cannot be easily categorized discretely, and there is no explicit road network in the open sea. Vessel trajectories need to be predicted with freedom of direction, without the constraints of a pre-defined road network.

2.2 Pedestrian Trajectory Prediction

Compared to land transportation, pedestrian walking in an open space resembles more to vessels traveling at sea but is susceptible to social interactions. Social-LSTM [2] is one of the earliest works on pedestrian trajectory prediction. It constructs distance matrices for pedestrians appearing at the same frame and models the trajectory at each time step using RNN, then computes the social interaction with pedestrians within certain distances from the hidden states. SGAN [20] follows the work of Social-LSTM but proposes a new pooling layer to learn social interactions. STGCNN [37] constructs a weighted spatio-temporal graph for all trajectories under the same time window, then performs spatio and temporal graph convolution neural network (GCN) [28] on the graph for feature extraction, which is further put forward to another temporal GCN for trajectory prediction. SGCN [41] also involves GCN for interaction learning. Unlike STGCNN, SGCN creates sparse spatio-temporal graphs by applying a self-attention mechanism and filtering out weak interactions under certain thresholds.

As we can see, social interaction plays an essential role in pedestrian trajectory prediction, and the interactions are learned starting from the distance matrices. However, computing the distance matrices becomes infeasible when it comes to predicting maritime trajectories due to the much larger traveling area and longer time span involved.

2.3 Vessel Trajectory Prediction

In the domain of vessel trajectory prediction, Forti et al. [16] make significant contributions by introducing a single-layer LSTM encoder-decoder architecture for trajectory prediction, marking a pioneering effort in utilizing deep learning techniques. Building upon this work, Gao et al. [17] introduce a Bi-LSTM model, which improves the relevance between historical and future data, thus enhancing prediction accuracy. Furthermore, Wang et al. [47] propose to augment the Bi-LSTM with an attention mechanism, resulting in an attention-based Bi-LSTM with superior performance compared to vanilla LSTM and Bi-LSTM. Capobianco ei al. [10] replace the general attention mechanism with the Bahdanau Attention Mechanism [5].

The aforementioned works leverage past movement patterns to make predictions. However, in the case of trajectories that lack strong periodic patterns like vessel trajectories, the past movement alone may not provide sufficient information. In such cases, historical traffic maps can be valuable as they offer potential insights into future movements. Nonetheless, models based solely on fixed traffic maps lack the necessary flexibility to generalize across different areas. To address this limitation, we propose CHAT, a trajectory prediction model that integrates historical information to guide future predictions in a generalizable manner.

3 METHODOLOGY

In this section, we elaborate on the design of our CHAT model for vessel trajectory prediction, leveraging conditional historical AIS trajectory data, in which a Bi-directional Temporal Convolutional Network (B-TCN) is used for feature extraction. For ease of reference, Table 1 summarizes the frequently used notations.

3.1 Notations and Problem Formulation

Denote by R the trajectory which is a sequence of timestamped locations $\{(x_t, y_t)\}$ over time $t \in \{1, 2, \ldots, T\}$ in a 2D-Cartesian plane, i.e., $R = \{(x_t, y_t)\}_{t=1}^{T}$, where x represents the latitude and y represents the longitude. Denote by τ the current timestamp of a vessel and by $R_\tau = \{(x_t, y_t)\}_{t=1}^{\tau}$ the observed sub-sequence of R until time τ. Let $\mathcal{R} = \{R^n\}_{n=1}^{N}$ be a historical trajectory database with a collection of N trajectories, where R^n is the n-th trajectory in the database. Corresponding to R_τ, let $R_\tau^n = \{(x_t^n, y_t^n)\}_{t=1}^{\tau}$ be a sub-sequence of R^n until time τ, where (x_t^n, y_t^n) is the location at time t

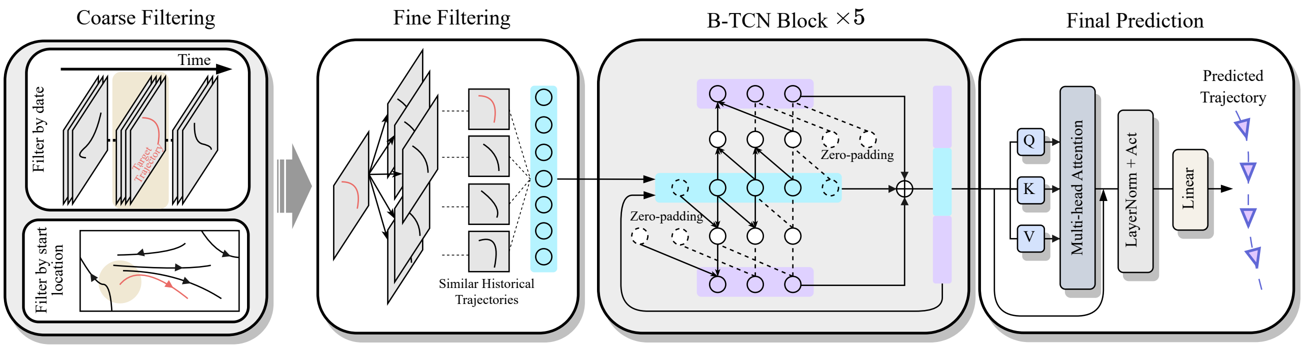

Figure 2: Architecture of CHAT.

Table 1: Frequently used notations.

Notation	Description
R	The (full) target trajectory
R_τ	The observed trajectory until current timestamp τ
R_{pred}	The future trajectory (ground truth) to be predicted
$\hat{R}_{\text{pred}}$	The predicted trajectory generated by the model
$\mathcal{R}$	Historical trajectory database with N trajectories
R^n	The n-th trajectory in the database $\mathcal{R}$
$\mathcal{R}_{\text{coarse}}$	A subset of $\mathcal{R}$ via coarse filtering
$\mathcal{R}_{\text{fine}}$	A subset of $\mathcal{R}$ via fine filtering

in trajectory R^n. For the target trajectory with an observation R_τ, we aim to predict the subsequent trajectory R_{pred} within a future time $t \in \{\tau+1, \tau+2, \ldots, T\}$, leveraging the historical trajectory database $\mathcal{R}$.

3.2 Model Description

The basic idea of CHAT is to predict the target trajectory based on similar historical trajectories. Therefore, the CHAT model's architecture consists of two main components: the determination of similar historical trajectories and the prediction of the future trajectory of the target vessel.

Inspired by the filtering-refine framework [55], which defines a trajectory signature to prune dissimilar trajectories and verifies the remaining candidates in the refining step, we propose a two-step coarse-to-fine approach to determine similar historical trajectories. This approach involves identifying the top-k similar trajectories from a historical trajectory database, with respect to the given target trajectory. Subsequently, these similar historical trajectories and the target trajectory, are passed through a B-TCN for feature extraction. The extracted features serve as contextual inputs for a self-attention layer, enabling the projection of the target vessel's future trajectory. Figure 2 gives the general architecture of CHAT.

3.2.1 Determination of Similar Historical Trajectories. Algorithm 1 presents the pseudocode of the coarse-to-fine approach for identifying the top-k similar trajectories from the historical trajectory database $\mathcal{R}$. We employ the Dynamic Time Warping (DTW) distance [7] to measure the similarity between two trajectories, which is widely used in time series distance calculations [3, 9, 44, 48, 54, 57]. DTW distance is effective for identifying trajectories that share similar shapes and movement patterns while accommodating variations in their moving radius and real geographical locations [42]. Ideally, we want to select the top-k similar trajectories with the smallest DTW distance. However, calculating pair-wise DTW distance for all trajectories is time-consuming. To accelerate the process, we propose a coarse-to-fine approach that utilizes some simple yet effective features, including the travel date and the starting location of the trajectory, as a coarse filtering. Intuitively, if two trajectories have close travel dates and starting locations, considering the external environmental factors (e.g., the current and wind directions), they are likely to be similar. Motivated by this remark, for an observed trajectory R_τ, we only consider those trajectories in $\mathcal{R}$ such that their travel dates are within a time span (i.e., m days) of R_τ and they fall within a specific range γ from the starting location of R_τ (Lines 1–8). Through such a coarse filtering, we get a subset $\mathcal{R}_{\text{coarse}}$ of $\mathcal{R}$ with a much smaller size while containing (almost) all the top-k similar trajectories to R_τ. Upon obtaining $\mathcal{R}_{\text{coarse}}$, we use a fine filtering to generate the top-k similar trajectories $\mathcal{R}_{\text{fine}}$. Specifically, we compute the DTW distance between R_τ and R^n_τ

Algorithm 1: Top-k Similar Trajectories

Input: an observed target trajectory R_τ, historical trajectory database $\mathcal{R} = \{R^n\}_{n=1}^N$, threshold m on difference of date, threshold γ on difference of starting location, integer k on the number of trajectories to be selected
Output: top-k similar trajectories $\mathcal{R}_{\text{fine}}$ selected from $\mathcal{R}$ for the observed target trajectory R_τ

`// coarse filtering`
1 $\mathcal{R}_{\text{coarse}} \leftarrow \emptyset$;
2 date$(R_\tau) \leftarrow$ date of R_τ;
3 loc$(R_\tau) \leftarrow$ starting location of R_τ;
4 **for** $n \in \{1, \ldots, N\}$ **do**
5   date$(R^n) \leftarrow$ date of R^n;
6   loc$(R^n) \leftarrow$ starting location of R^n;
7   **if** $\text{date}(R_\tau) - \text{date}(R^n) \le m \wedge |\text{loc}(R_\tau) - \text{loc}(R^n)| \le \gamma$ **then**
8   update $\mathcal{R}_{\text{coarse}} \leftarrow \mathcal{R}_{\text{coarse}} \cup \{R^n\}$;

`// fine filtering`
9 compute $d(R^n) \leftarrow \text{DTWdistance}(R_\tau, R^n_\tau)$ for $R^n \in \mathcal{R}_{\text{coarse}}$;
10 sort $\mathcal{R}_{\text{coarse}}$ with respect to $d(R^n)$ in ascending order;
11 $\mathcal{R}_{\text{fine}} \leftarrow$ first k trajectories in $\mathcal{R}_{\text{coarse}}$;
12 **return** $\mathcal{R}_{\text{fine}}$;

for each trajectory R^n in $\mathcal{R}_{\text{coarse}}$, and select k trajectories with the smallest DTW distance as $\mathcal{R}_{\text{fine}}$ (Lines 9–11).

3.2.2 Prediction of Future Trajectory. All trajectories participating in the prediction process will go over the data normalization step. The absolute latitude and longitude values, represented by $(x_0, y_0), \ldots, (x_\tau, y_\tau)$, are transformed into displacement values relative to the previous position, resulting in $(0, 0), \ldots, (x_l - x_{l-1}, y_l - y_{l-1})$. These displacement values were then normalized to fall within $[-1, 1]$ range. Finally, the normalized trajectories of equal length were generated, thereby preparing them for training by the model.

For a trajectory $R^n \in \mathcal{R}_{\text{fine}}$, since R_τ is similar to R^n_τ, we expect a small value of DTW distance between the full trajectories R and R^n, indicating that R_{pred} and R^n_{pred} share similar patterns and hence the features extracted from R^n can be used to predict R_{pred}. Together with the observed target trajectory R_τ, we develop a B-TCN for enriching feature extraction, and apply a self-attention layer for the ultimate trajectory prediction.

B-TCN is based on the conventional TCN [6]. TCN incorporates causal convolutional layers, which can be simplified as "1D convolutional layers". Unlike standard convolutional layers that apply padding on both ends of the sequence, causal convolutional layers exclusively pad the sequence at the beginning with a padding length of (kernel size − 1) to ensure convolutions solely operate on elements preceding timestamp t. The original TCN employs causal convolutional layers to prevent the contamination of future information into the past. However, given that the CHAT model provides historical sequences similar to the target sequence, it becomes crucial to incorporate the "historical future information". Consequently, B-TCN encompasses both TCN and reverse TCN, facilitating access to both past and future contextual information when making predictions at any sequence point. The reverse TCN layers exclusively rely on "future" information occurring at and after timestamp t by padding the sequence at the end, thereby ensuring that current decisions entirely rely on the anticipation of the future. By incorporating both TCN and reverse TCN layers, B-TCN expands the capabilities of the conventional TCN and enables the utilization of past and future contextual information.

B-TCN is particularly well-suited for scenarios where the current determination of an object is influenced by both the past pattern and the external factors from the future. Similarly, in the context of vessel navigation, a vessel's course of action is influenced by the historical movement and the oceanographic conditions ahead. By incorporating past and future information, B-TCN enriches the extracted features when making trajectory predictions. Each B-TCN block contains two TCN and reverse TCN layers respectively. An Exponential Linear Unit (ELU) activation function will be added to the end of each convolutional layer. For each block, the output of the TCN and reverse TCN layers will be concatenated together, then connected with residuals to prevent gradient vanish. The convolutional layers will be dilated with a dilation size of 2^d, where d is the number of B-TCN blocks. Therefore, by stacking multiple B-TCNs, each output will incorporate information from progressively further away from the current time step. The feature extracted from B-TCN is subsequently passed to a self-attention layer for the ultimate trajectory prediction.

B-TCN, incorporating both past and future contextual information, provides an effective framework for enriching feature extraction in trajectory prediction tasks, resulting in accurate predictions considering the interplay between historical patterns and external factors.

4 DATASET CONSTRUCTION

To evaluate the performance of the CHAT model, we construct a novel maritime trajectory dataset using AIS data. We select AIS data for two primary considerations. First, AIS data offers finer granularity than other ship trajectory recording data, enabling more precise tracking of vessel trajectories. Second, it can be readily obtained from reliable official sources. For constructing our dataset, we have selected two government-provided data sources [4, 14]. The first source, hereinafter referred to as DMA, was derived from the open vessel trajectory source made available by the Danish Maritime Authority [4]. The second source, hereinafter referred to as USA, was obtained from the Office for Coastal Management [14]. Both sources provide the raw signals transmitted from the onboard AIS equipment. The dataset construction process involves fundamental steps of trajectory data mining, including data preprocessing such as noise filtering and segmentation [58] and area partitioning.

4.1 Data Pre-processing

AIS continuously transmits both static and dynamic information about a vessel, such as latitude, longitude, ship type, among others. The broadcasting frequency of AIS data varies depending on the vessel's speed and course, typically ranging from a few seconds to up to three minutes [21]. Consequently, the dataset exhibits a non-fixed time interval and contains a notable amount of redundant information and noise. To address these issues, a series of pre-processing steps are executed on the dataset prior to the construction of the dataset.

Selection of relevant information. We focus on 5 features: MMSI (Maritime Mobile Service Identity, i.e., the unique ID number of vessels), time, latitude, longitude, and navigational status. These features provide sufficient spatio-temporal information about trajectories, setting the stage for the following data preprocessing steps. Any data points with MMSI of less than 9 digits are eliminated as they are non-vessel-related signals.

Handling trajectory disruptions. We organize data points based on the MMSI and arrange them chronologically to deduce possible vessel trajectories. For AIS signals emitted from the same vessel, we break them into shorter segments when the time interval between two consecutive data points exceeds 30 minutes. This segmentation is necessary because vessel operations may temporarily pause, the AIS system may be manually deactivated during prolonged navigations, or signals may go missing, resulting in gaps in spatial and temporal data. It is hard to restore trajectories accurately within these gap periods. Therefore, we establish a trajectory separation threshold of 30 minutes, which is ten times the maximum signal interval of 3 minutes, minimizing the impact of trajectory disruptions on data quality.

Table 2: Statistics of our vessel trajectory dataset.

Area name	Latitude (degrees)	Longitude (degrees)	Time period	Number of vessels	Number of trajectories
Area 1	55 to 58.5	10.3 to 13	2019-Jan-01 to 2019-Mar-31	1295	16155
Area 2	54 to 55.5	12 to 14.5	2019-Jan-01 to 2019-Mar-31	1246	8960
Area 3	54 to 56	5 to 8	2019-Jan-01 to 2019-Mar-31	622	1651
Area 4	22 to 26	-85 to -80	2023-Jan-01 to 2023-Feb-28	2628	17943
Area 5	41 to 46	-85 to -75	2023-Jan-01 to 2023-Feb-28	292	14421
Area 6	17 to 20	-67 to -64	2023-Jan-01 to 2023-Feb-28	1298	19184

Exclusion of static trajectories. Trajectories labeled as "Moored" or "Anchored" in the navigational status are discarded. Additionally, trajectories exhibiting zero standard deviation in both latitude and longitude are eliminated. This characteristic suggests that either the vessel is inactive or the positional signal is inaccurate.

Data interpolation. The raw trajectory data exhibits a non-fixed time interval, resulting in potential variations in the number of sample points between trajectories traveling the same routes. This variability can directly influence the length of predicted trajectories, making it challenging to control the prediction's length if the model generates a fixed number of data points. Predicting the temporal dimension is also a crucial aspect of trajectory prediction, as it involves forecasting when the target will move to a specific location. To gain control over prediction length, we need to employ data interpolation. We apply cubic spline interpolation [36] to the original data to address the non-fixed signal broadcasting frequency of the AIS, which does not align with the input requirements of our designed model. Cubic spline interpolation is widely used in assisting path planning and trajectory interpolation [11, 25, 33, 56], and it shows to produce better trajectories for AIS data [56]. Following this, we re-sample the interpolated data at a fixed rate of 30 seconds. This re-sampling step ensures that the data adheres to the desired fixed time intervals, enabling compatibility with our model's input structure.

Unqualified speed removal. This process involves the removal of trajectories that contain any data points displaying impractical speeds, specifically when the speed exceeds 100 knots (i.e., 185 km/h) [1]. The speed is calculated for every two consecutive points using their latitudes and longitudes, employing the haversine distance formula 1. These trajectories exhibit extreme spatial variation, significantly disrupting the quality of the dataset, and thus should be removed.

4.2 Area Selection

To ensure data diversity, we identify three types of navigational areas—the general area, area with clear waterways, and area without clear waterways—from each of the DMA and USA datasets, resulting in a total of six areas. The DMA dataset is examined within a specific time-frame (January 2019 to March 2019) and a well-defined geographic region (latitude: 53 to 60 degrees, longitude: 2 to 19 degrees). Three areas are further selected within this region: Area 1 (latitude: 55 to 58.5 degrees, longitude: 10.3 to 13 degrees) represents the general area, Area 2 (latitude: 54 to 55.5 degrees, longitude: 12 to 14.5 degrees) indicates an area with clear waterways, and Area 3 (latitude: 54 to 56 degrees, longitude: 5 to 8 degrees) represents an area without clear waterways.

Likewise, the USA dataset undergoes investigation within a specific timeframe (January 2023 to February 2023) and a designated geographic region (latitude: 15 to 50 degrees, longitude: -100 to -60 degrees). Three areas are also selected with this region: Area 4 (latitude: 22 to 26 degrees, longitude: -85 to -80 degrees) represents the general area, Area 5 (latitude: 41 to 46 degrees, longitude: -85 to -75 degrees) denotes an area with clear waterways, and Area 6 (latitude: 17 to 20 degrees, longitude: -67 to -64 degrees) indicates an area without clear waterways.

The selection of areas aims to encompass diverse navigational scenarios. The area with clear waterways simulates land-based conditions, resembling the adherence to pre-determined road networks, while the area without clear waterways represents open seas with unrestricted vessel navigation. The general area combines characteristics from both areas with and without clear waterways. The resultant dataset comprises the processed trajectories extracted from the six areas. Table 2 gives the statistics of our dataset. Figure 3 visually depicts the geographic distribution of trajectories in each selected area.

5 EXPERIMENTS

We evaluate our model using the newly constructed dataset, which comprises trajectories in six distinct areas. Trajectories are divided into equal-length segments using a sliding window approach. The window size is set at 4 hours, with a step length of 1 hour. Following this operation, all trajectories are segmented into fixed lengths of 4 hours with sampling rate of 30 seconds, each containing 480 data points that represent the trajectory.

Our experiment focuses on observing the first hour of the target trajectory and predicting the subsequent three hours of future trajectory. We chose these parameters based on vessel collision case studies, which suggest that early recognition before a collision can occur as soon as 34 minutes before the collision [27]. Additionally, it typically takes 1.6 to 2.4 hours to detect trajectory anomalies [46]. Therefore, whether for collision avoidance, risk assessment, or trajectory anomaly detection, it is prudent to predict trajectories for a minimum of 1 to 3 hours in advance.

5.1 Model Training

We use Area 1 (General area in DMA) for model training to ensure a balanced representation of trajectories that encompasses properties observed in areas with and without clear waterways. Specifically, for every trajectory spanning a duration of 4 hours, we designate

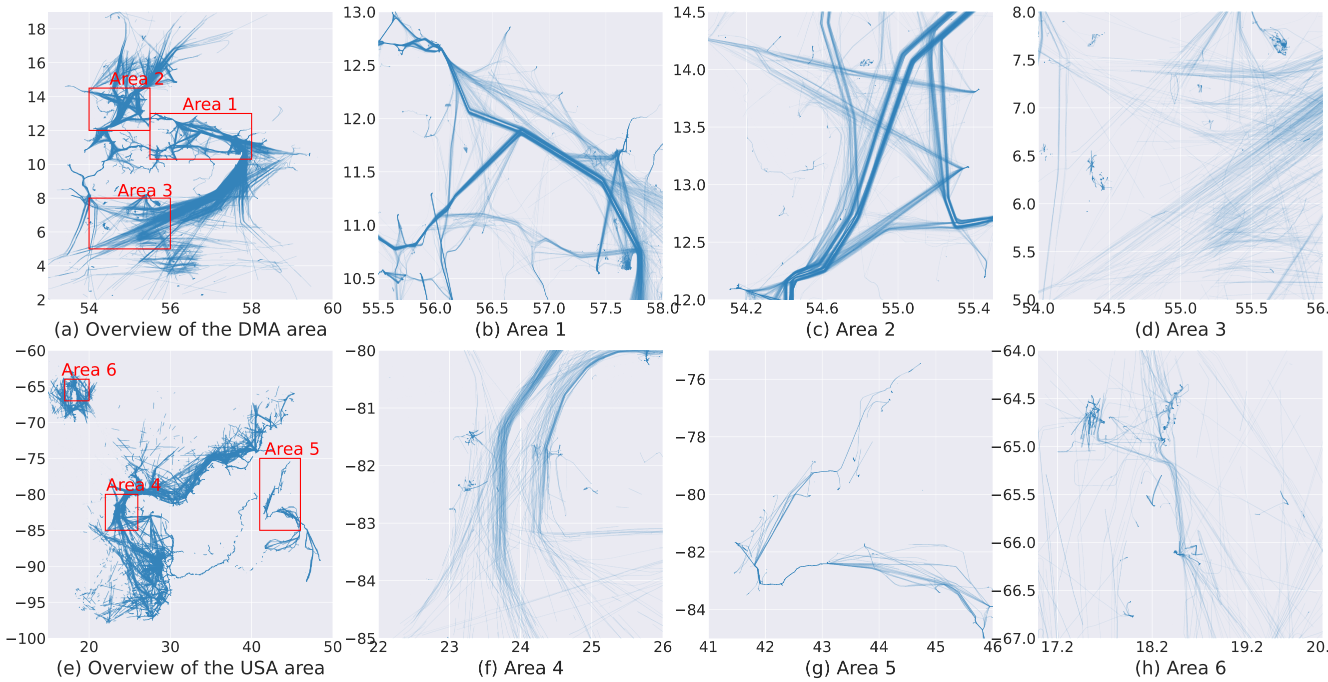

Figure 3: Geographic distribution of vessel trajectories of the six distinct areas in our dataset. (a) Overview of trajectory distribution in DMA areas. (b) Trajectory distribution in the area selected for training. This is the general area from DMA. (c) Trajectories are compact and neatly arranged, indicating that there may be a well-defined waterway framework in this area. This is the area with clear waterway from DMA. (d) Trajectories are scattered, indicating that this area is an open sea. This is the area without clear waterway from DMA. (e) Overview of trajectory distribution in the USA areas. (f) The general area from USA. (g) The area with clear waterway from USA. (h) The area without clear waterway from USA.

the initial 1 hour (120 data points) as R_τ (observation section) and the subsequent 3 hours (360 data points) as R_{pred} (prediction section). In order to construct the coarse set for each target trajectory, we consider the historical trajectories from the preceding 30 days from the happening date of the target trajectory. That is, we select trajectories where the starting locations are within a range of 0.005 degrees of latitude and longitude compared to the starting location of the target trajectory. Subsequently, we extract the top-7 trajectories from the coarse set to create the fine set based on their minimal DTW distance to the target trajectory.

The CHAT model comprises 5 B-TCN blocks, each comprising two layers of TCN and reverse TCN layers, respectively, and connecting through residual connections. With each subsequent block added, the convolutions will be dilated by a factor of 2^n, where n is the number of blocks. The hidden size of each block is specified as $[2, 2, 2, 2, 2]$. The training process spans 50 epochs, while the initial learning rate is set to 0.0001.

5.2 Evaluation Metrics

We evaluate our model using the commonly utilized Average Displacement Error (ADE) and Final Displacement Error (FDE). To calculate these metrics, both the predicted trajectory $\hat{R}_{\text{pred}}$ and the ground truth trajectory R_{pred} are de-normalized back to their original space, after which the ADE and FDE are computed by comparing the trajectories, i.e.,

$$\text{ADE} = \frac{\sum_{t=\tau+1}^{T} \|(\hat{x}_t, \hat{y}_t) - (x_t, y_t)\|^2}{T - \tau},$$
$$\text{FDE} = \|(\hat{x}_T, \hat{y}_T) - (x_T, y_T)\|^2,$$

where τ is the current timestamp and T is the final timestamp, $(\hat{x}_t, \hat{y}_t)$ is the predicted location at the t-th timestamp and (x_t, y_t) is the corresponding ground truth. The computed ADE and FDE will be converted to kilometers using the haversine distance formula, which calculates the distance between two points on the surface of a sphere.

$$d = 2r \times \arcsin\left(\sqrt{\sin^2(\bar{\theta}) + \cos(\theta_1)\cos(\theta_2)\sin^2(\bar{\lambda})}\right), \qquad (1)$$

where r is the radius of the earth in kilometers, $\bar{\theta} = \frac{(\theta_2 - \theta_1)}{2}$, $\bar{\lambda} = \frac{(\lambda_2 - \lambda_1)}{2}$, θ_1 and θ_2 denote the latitudes while λ_1 and λ_2 denotes the longitudes of the predicted and the ground truth trajectories, respectively.

5.3 Comparison with Baselines

The performance of the CHAT model is compared to five baseline methods: LSTM [16], BiLSTM+ATT [47], Bahdanau Attention [10], TCN [6] and CHAT-TCN. LSTM [16], BiLSTM+ATT [47], and Bahdanau Attention [10] are specifically designed for vessel trajectory prediction. LSTM [16], which adopts the LSTM encoder-decoder

Table 3: The ADE/FDE metrics for five baseline methods in comparison with CHAT (lower values indicate better performance).

Model	Time step	ADE/FDE Area 1	Area 2	Area 3	Area 4	Area 5	Area 6
LSTM [16]	1-h	5.89/11.95	7.58/15.10	4.98/9.91	2.66/5.28	0.53/1.01	2.32/4.68
	2-h	12.05/24.27	15.52/31.81	10.13/20.88	5.34/10.75	1.00/1.94	4.70/9.51
	3-h	18.17/36.67	23.81/48.89	15.64/32.29	8.07/16.35	1.46/2.83	7.18/14.72
BiLSTM+ATT [47]	1-h	21.22/41.20	29.54/57.66	22.16/42.94	4.59/8.95	1.78/3.47	3.24/6.32
	2-h	42.17/85.54	59.40/121.61	44.32/90.92	9.18/18.74	3.60/7.32	6.48/13.21
	3-h	64.02/127.15	90.79/182.17	67.93/136.66	14.03/28.36	5.47/10.90	9.87/19.71
Bahdanau Attention [10]	1-h	3.32/7.20	4.29/9.53	2.38/5.28	0.82/1.72	0.27/0.47	0.58/1.27
	2-h	7.71/17.15	10.82/25.95	6.15/15.47	1.85/4.14	0.48/0.94	1.43/3.51
	3-h	12.82/29.26	19.34/47.20	11.58/29.74	3.12/7.38	0.75/1.63	2.63/6.59
TCN [6]	1-h	10.48/21.00	15.26/30.91	8.96/18.06	2.26/4.53	1.26/2.53	2.23/4.47
	2-h	20.94/41.53	31.34/63.70	18.55/38.24	4.59/9.32	2.57/5.14	4.49/9.01
	3-h	31.25/61.97	47.73/97.10	28.52/58.66	7.01/14.49	3.87/7.70	6.77/13.62
CHAT-TCN	1-h	2.60/5.93	3.19/7.19	1.82/4.18	0.43/0.84	0.25/0.49	0.50/1.10
	2-h	6.21/13.53	7.52/16.76	4.35/9.90	0.98/2.38	0.47/0.95	1.25/3.10
	3-h	10.03/22.11	12.48/**27.90**	7.32/16.70	1.85/5.01	0.74/1.69	2.34/5.96
CHAT	1-h	**2.48/5.68**	**2.86/6.56**	**1.06/2.63**	**0.30/0.73**	**0.17/0.29**	**0.32/0.77**
	2-h	**6.01/13.38**	**7.16/16.70**	**3.01/7.69**	**0.81/1.98**	**0.30/0.60**	**0.90/2.36**
	3-h	**9.81/21.81**	**12.30**/28.37	**5.60/13.95**	**1.52/4.01**	**0.47/1.03**	**1.75/4.55**

structure, is one of the most classical models for vessel trajectory prediction. BiLSTM+ATT [47], which incorporates a bi-directional LSTM with a self-attention layer, is a popular architecture for vessel trajectory prediction. Bahdanau Attention [10] improves upon the bi-directional LSTM and self-attention structure by using Bahdanau Attention. TCN [6], on the other hand, is a distinct type of time series prediction model that differs from LSTM and serves as the foundation for the CHAT model. In the CHAT model, we use B-TCN for feature extraction, which is based on the top 7 similar historical trajectories. CHAT-TCN, while retaining the architecture of CHAT, serves to evaluate the effectiveness of B-TCN. CHAT-TCN also considers the top 7 similar historical trajectories but employs the original TCN layer for feature extraction instead of B-TCN.

We compare the long-term prediction performance of the six models in terms of ADE and FDE, up to a three-hour horizon, across the six distinct areas of our dataset. Table 3 presents the model performances for the ADE/FDE, which shows that the CHAT model consistently outperforms the other five baselines across all areas. Specifically, except for predicting the 3-hour trajectory in Area 2, where CHAT has a slightly higher FDE, CHAT achieves the best ADE/FDE among all the tested models in the six areas over the entire 3-hour time span. Furthermore, the overall consistently higher performance of CHAT over CHAT-TCN demonstrates the superiority of B-TCN.

Specifically, for the average 1-hour ADE metric, the CHAT model outperforms previous methods, including LSTM [16], BiLSTM+ATT [47], Bahdanau Attention [10], TCN [6], and CHAT-TCN, with percentage improvements of 70.0%, 91.3%, 38.4%, 82.2%, and 18.3%, respectively. Similarly, for the average 1-hour FDE metric, the CHAT model also surpasses other methods by a significant margin, achieving percentage improvements of 65.2%, 89.6%, 34.5%, 79.6%, and 15.6% for LSTM, BiLSTM+ATT, Bahdanau Attention, TCN, and CHAT-TCN, respectively. These results demonstrate the superiority of the CHAT model in maritime trajectory prediction. The robust performance of the CHAT model in different areas further supports its effectiveness in various navigational contexts.

In addition to evaluating model performance, we investigate the rate of error accumulation across the 3-hour time span to assess the models' robustness over time. The detailed ADE and FDE results for all models, ranging from 15 minutes to 3 hours, are presented in Figure 4 and Figure 5. Both figures demonstrate that the CHAT model exhibits notably smoother slopes for ADE and FDE throughout the entire 3-hour duration. This characteristic highlights the CHAT model's capability to reduce error accumulation, a crucial aspect for accurate long-term predictions, and solidifies its superiority in trajectory prediction and robustness over time.

5.4 Ablation Study

The objective of this section is two-fold: first, to assess the influence of historical information on the accuracy of trajectory prediction (i.e., the size of the coarse set), and second, to determine the optimal amount of historical information utilized for prediction (i.e., the size of the fine set). For this objective, we examine the performance of the CHAT model with various sizes of the coarse set. The coarse set

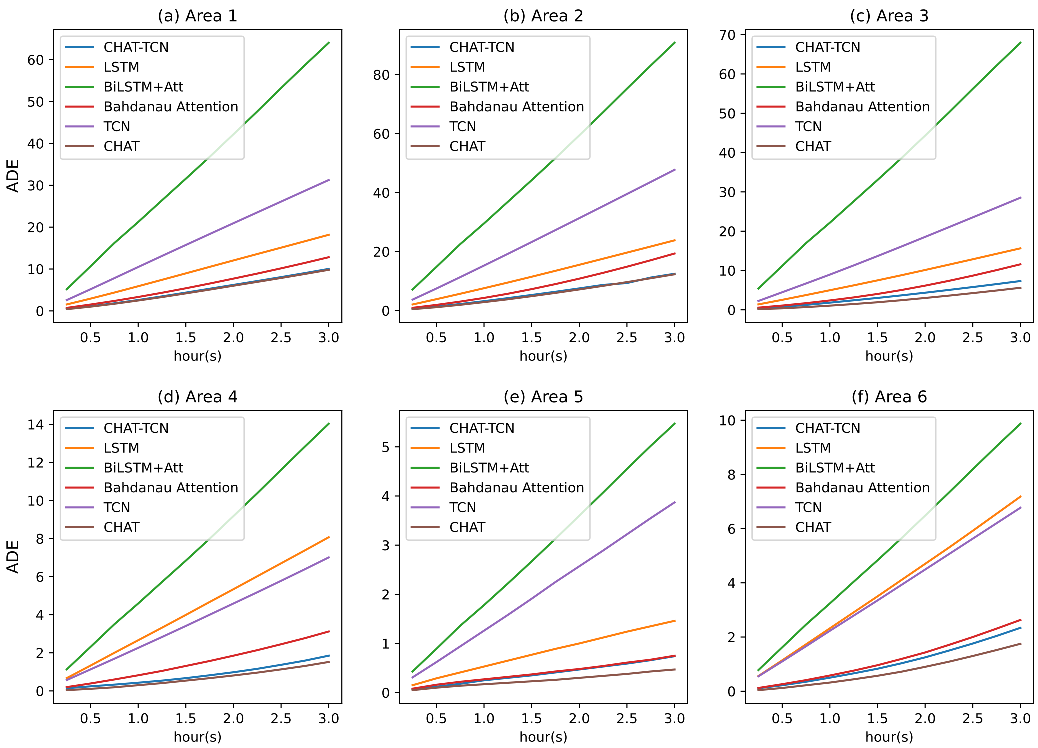

Figure 4: The ADE metrics for the five baseline methods in comparison with CHAT (Lower values indicate better performance).

comprises trajectories m days prior to the target trajectory, where $m \in [1, 15, 30, 45, 60]$. We also examine the model performance with different sizes of the fine set R_{fine}. The fine set contains top-k similar historical trajectories concerning the target trajectory, where $k \in [1, 3, 5, 7, 9]$. The ablation study uses the same area as the model training, corresponding to Area 1: the general area (DMA). The results of our ablation study are presented in Figure 6.

Based on the results, the best performance is achieved when m, the size of the coarse set, is set to 30. Setting the size of the coarse set either too small or too large negatively impacts the model's performance. To the best of our knowledge, the underlying reason for this is twofold. On one hand, the database needs to accumulate a sufficient amount of data to cover a wide range of situations that queries may encounter. On the other hand, sea navigation is influenced by monsoon circulation and ocean currents, which vary by season. The further the collected trajectory data is from the testing date, the more significant the trajectory deviation becomes.

The ideal scenario is to possess a sufficiently large database that includes trajectories spanning multiple years and seasons with similar conditions. Nonetheless, there exists a trade-off between the size of the coarse set and the inference time. Table 4 presents the inference time in seconds for one trajectory with respect to different sizes of the coarse set. The inference time is computed by averaging the inference time of all test trajectories from Area 1. As the size of the coarse set increases, the inference time also increases.

Table 4: Inference time (in seconds) of the CHAT model with the coarse set containing different numbers of days, m, of historical trajectories having occurred prior to the target trajectory.

Size of m	Inference time
1 day	0.044
15 days	0.32
30 days	0.66
45 days	0.79
60 days	0.98

Therefore, considering the model performance and computational efficiency, we take $m = 30$ as the default setting of the coarse set.

As for the size of the fine set, setting k to 7 consistently delivers the best performance in terms of ADE and FDE throughout the 3-hour time span, under the condition of $m = 30$. However, when k equals 9, the results show the worst performance in terms of ADE and FDE regardless of the size of the coarse set. This indicates that historical trajectory information is crucial to consider for better predictions. These findings support the notion that employing an appropriate amount of historical information significantly improves in trajectory prediction accuracy. Considering both computational

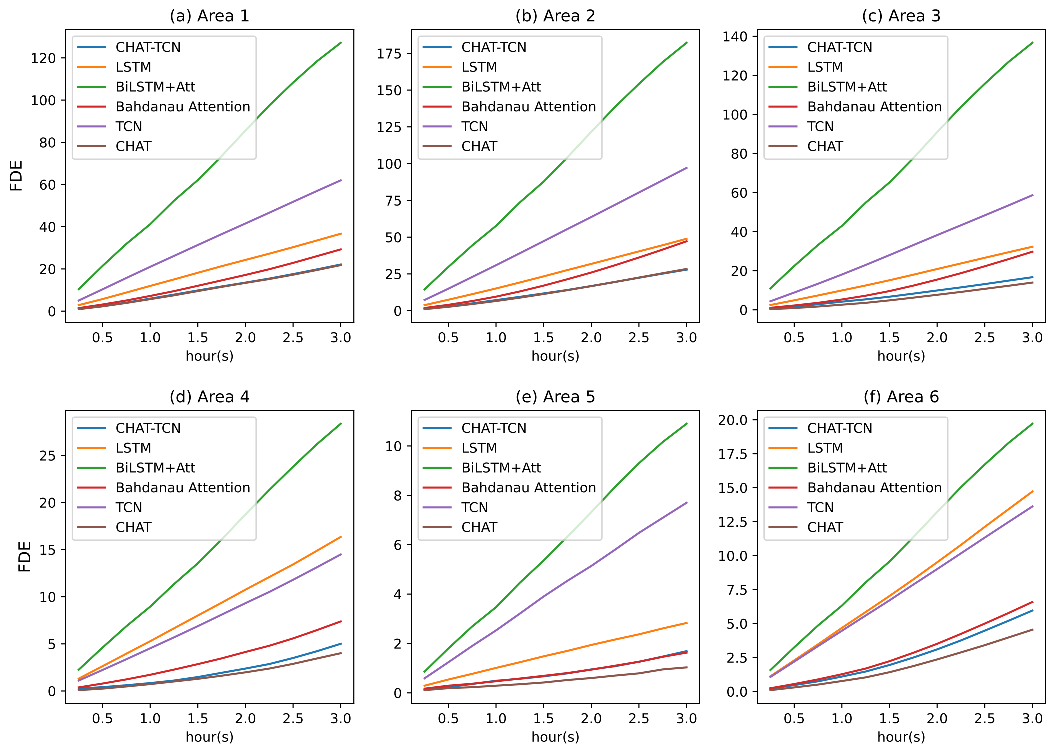

Figure 5: The FDE metrics for the five baseline methods in comparison with CHAT (Lower values indicate better performance).

efficiency and model performance, our study establishes $m = 30$ and $k = 7$ as the recommended configuration for the CHAT model.

6 CONCLUSION

In this paper, we address the challenge of maritime route prediction by introducing the CHAT model, which leverages similar historical trajectories and the target trajectory to make accurate predictions without relying on pre-determined networks or social interactions. The proposed B-TCN further enhances the prediction performance by incorporating both TCN and reverse TCN layers, enabling information extraction from both past and future elements. Additionally, we construct a vessel trajectory database comprising six distinct navigational areas to ensure data diversity. Through evaluation on this dataset, we demonstrate the CHAT model's effectiveness in trajectory prediction and robustness over long-term predictions. This work contributes to the advancement of trajectory analysis in maritime navigation, paving the way for enhanced safety and efficiency in vessel operations.

For future work, an important direction would be to explore more effective similarity search methods for identifying similar historical vessel trajectories. While our CHAT model leverages similar trajectories, further optimization of the efficiency and accuracy of the similarity search process could potentially improve trajectory prediction performance.

REFERENCES

[1] Roar Adland, Gro Fonnes, Haiying Jia, Ove D Lampe, and Siri P Strandenes. 2017. The impact of regional environmental regulations on empirical vessel speeds. *Transportation Research Part D: Transport and Environment* 53 (2017), 37–49.

[2] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. 2016. Social lstm: Human trajectory prediction in crowded spaces. *In Proceedings of the IEEE conference on computer vision and pattern recognition,* (2016), 961–971.

[3] Ira Assent, Marc Wichterich, Ralph Krieger, Hardy Kremer, and Thomas Seidl. 2009. Anticipatory DTW for efficient similarity search in time series databases. *Proceedings of the VLDB Endowment* 2, 1 (2009), 826–837.

[4] Danish Maritime Authority. 2023. Historical AIS Data. http://web.ais.dk/aisdata/.

[5] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[6] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* (2018).

[7] Richard Bellman and Robert Kalaba. 1959. On adaptive control processes. *IRE Transactions on Automatic Control* 4, 2 (1959), 1–9.

[8] Holger Berndt, Jorg Emmert, and Klaus Dietmayer. 2008. Continuous driver intention recognition with hidden markov models. In *2008 11th International IEEE Conference on Intelligent Transportation Systems.* IEEE, 1189–1194.

[9] K Selçuk Candan, Rosaria Rossini, Maria Luisa Sapino, and Xiaolan Wang. 2012. sDTW: computing DTW distances using locally relevant constraints based on salient feature alignments. *arXiv preprint arXiv:1208.0275* (2012).

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

[11] Jing M Chen, Feng Deng, and Mingzhen Chen. 2006. Locally adjusted cubic-spline capping for reconstructing seasonal trajectories of a satellite-derived surface parameter. *IEEE Transactions on Geoscience and Remote Sensing* 44, 8 (2006), 2230–2238.

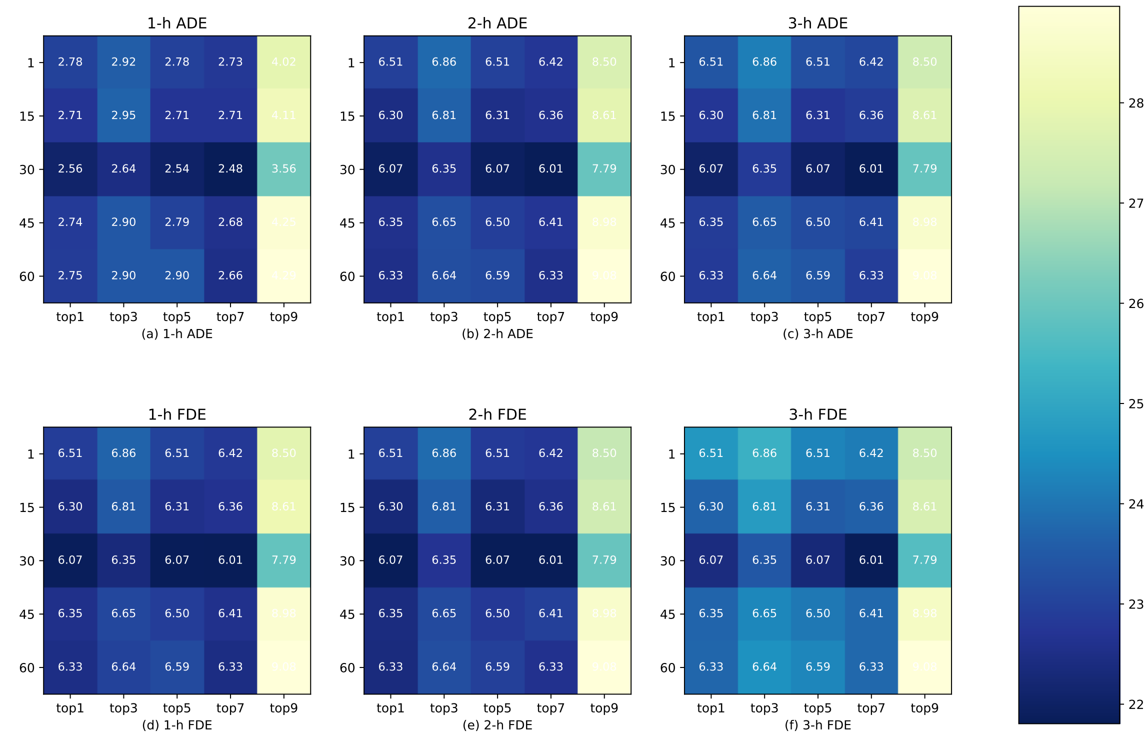

Figure 6: Abolation study of the CHAT model conducted in Area 1, the general area in DMA. The top row represents the Average Displacement Error (ADE) of the CHAT model, with the size of the coarse set varied from containing 1 day to 60 days of trajectories prior to the target trajectory, with each group of experiment setting spaced 15 days apart. Each subplot illustrates the ADE of the CHAT model with various sizes of the fine set, ranging from 1 trajectory (i.e., $|\mathcal{R}_{\text{fine}}| = 1$) to 9 trajectories (top-9, i.e., $|\mathcal{R}_{\text{fine}}| = 9$), corresponding to the size of the coarse set. Similarly, the bottom row displays the Final Displacement Error (FDE) of CHAT, with different sizes of coarse and fine sets.

[12] Nachiket Deo and Mohan M Trivedi. 2018. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 1468–1476.

[13] Wenchao Ding and Shaojie Shen. 2019. Online vehicle trajectory prediction using policy anticipation network and optimization-based context reasoning. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 9610–9616.

[14] Office for Coastal Management. 2023. Nationwide Automatic Identification System 2023. https://www.fisheries.noaa.gov/inport/item/70162.

[15] Nicola Forti, Leonardo M Millefiori, Paolo Braca, and Peter Willett. 2020. Prediction oof vessel trajectories from AIS data via sequence-to-sequence recurrent neural networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 8936–8940.

[16] Nicola Forti, Leonardo M Millefiori, Paolo Braca, and Peter Willett. 2020. Prediction oof vessel trajectories from AIS data via sequence-to-sequence recurrent neural networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 8936–8940.

[17] Miao Gao, Guoyou Shi, and Shuang Li. 2018. Online prediction of ship behavior with automatic identification system sensor data using bidirectional long short-term memory recurrent neural network. *Sensors* 18, 12 (2018), 4211.

[18] Fosca Giannotti, Mirco Nanni, Fabio Pinelli, and Dino Pedreschi. 2007. Trajectory pattern mining. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. 330–339.

[19] Chenjuan Guo, Bin Yang, Jilin Hu, and Christian Jensen. 2018. Learning to route with sparse trajectory sets. In *2018 IEEE 34th International Conference on Data Engineering (ICDE)*. IEEE, 1073–1084.

[20] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. 2018. Social gan: Socially acceptable trajectories with generative adversarial networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recogni- tion* (2018), 2255–2264.

[21] Abbas Harati-Mokhtari, Alan Wall, Philip Brooks, and Jin Wang. 2007. Automatic Identification System (AIS): data reliability and human error implications. *The Journal of Navigation* 60 (2007), 373–389.

[22] Chih-Chieh Hung, Wen-Chih Peng, and Wang-Chien Lee. 2015. Clustering and aggregating clues of trajectories for mining trajectory patterns and routes. *The VLDB Journal* 24 (2015), 169–192.

[23] Ibadurrahman, Kunihiro Hamada, Yujiro Wada, Jota Nanao, Daisuke Watanabe, and Takahiro Majima. 2021. Long-Term Ship Position Prediction Using Automatic Identification System (AIS) Data and End-to-End Deep Learning. *Sensors* 21, 21 (2021), 7169.

[24] Fengmei Jin, Wen Hua, Jiajie Xu, and Xiaofang Zhou. 2019. Moving object linking based on historical trace. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 1058–1069.

[25] Wonseok Kang, Soohwan Yu, Seungyong Ko, and Joonki Paik. 2015. Multisensor super resolution using directionally-adaptive regularization for UAV images. *Sensors* 15, 5 (2015), 12053–12079.

[26] Amin Vahedian Khezerlou, Xun Zhou, Ling Tong, Yanhua Li, and Jun Luo. 2019. Forecasting gathering events through trajectory destination prediction: A dynamic hybrid model. *IEEE transactions on knowledge and data engineering* 33, 3

(2019), 991–1004.
[27] Jong-Kwan Kim and Deuk-Jin Park. 2023. Determining the Proper Times and Sufficient Actions for the Collision Avoidance of Navigator-Centered Ships in the Open Sea Using Artificial Neural Networks. *Journal of Marine Science and Engineering* 11, 7 (2023), 1384.
[28] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *In ICLR* (2016).
[29] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, Hamid Rezatofighi, and Silvio Savarese. 2019. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *In NeurIPS* (2019), 137–146.
[30] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. 2007. Crowds by example. *In Computer graphics forum* 26, 3 (2007), 655–664.
[31] Junxiang Li, Bin Dai, Xiaohui Li, Xin Xu, and Daxue Liu. 2019. A dynamic Bayesian network for vehicle maneuver prediction in highway driving scenarios: Framework and verification. *Electronics* 8, 1 (2019), 40.
[32] Wenkai Li, Chunwei Zhang, Jie Ma, and Chengfeng Jia. 2019. Long-term vessel motion predication by modeling trajectory patterns with AIS data. In *2019 5th International Conference on Transportation Information and Safety (ICTIS)*. IEEE, 1389–1394.
[33] Jianfang Lian, Wentao Yu, Kui Xiao, and Weirong Liu. 2020. Cubic spline interpolation-based robot path planning using a chaotic adaptive particle swarm optimization algorithm. *Mathematical problems in engineering* 2020 (2020), 1–20.
[34] Qin Liu, Panlin Hou, Guojun Wang, Tao Peng, and Shaobo Zhang. 2019. G*-tree: An efficient spatial index on road networks. *In 2019 IEEE 35th International Conference on Data Engineering (ICDE)* (2019), 268–279.
[35] Wuman Luo, Haoyu Tan, Lei Chen, and Lionel M Ni. 2013. Finding time period-based most frequent path in big trajectory data. In *Proceedings of the 2013 ACM SIGMOD international conference on management of data*. 713–724.
[36] Sky McKinley and Megan Levine. 1998. Cubic spline interpolation. *College of the Redwoods* 45, 1 (1998), 1049–1060.
[37] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. 2020. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. *in Conference on Computer Vision and Pattern Recognition* (2020).
[38] Brian Murray and Lokukaluge Prasad Perera. 2021. An AIS-based deep learning framework for regional ship behavior prediction. *Reliability Engineering & System Safety* 215 (2021), 107819.
[39] Giuliana Pallotta, Michele Vespe, and Karna Bryan. 2013. Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction. *Entropy* 15, 6 (2013), 2218–2245.
[40] Seong Hyeon Park, ByeongDo Kim, Chang Mook Kang, Chung Choo Chung, and Jun Won Choi. 2018. Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture. In *2018 IEEE intelligent vehicles symposium (IV)*. IEEE, 1672–1678.
[41] Liushuai Shi, Le Wang, Chengjiang Long, Sanping Zhou, Mo Zhou, Zhenxing Niu, and Gang Hua. 2021. SGCN: Sparse graph convolution network for pedestrian trajectory prediction. *InProceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 8994–9003.
[42] Han Su, Shuncheng Liu, Bolong Zheng, Xiaofang Zhou, and Kai Zheng. 2020. A survey of trajectory distance measures and performance evaluation. *The VLDB Journal* 29 (2020), 3–32.
[43] Han Su, Kai Zheng, Kai Zeng, Jiamin Huang, and Xiaofang Zhou. 2014. STMaker: a system to make sense of trajectory data. *Proceedings of the VLDB Endowment* 7, 13 (2014), 1701–1704.
[44] Zhifu Tao, Qinghua Xu, Xi Liu, and Jinpei Liu. 2023. An integrated approach implementing sliding window and DTW distance for time series forecasting tasks. *Applied Intelligence* (2023), 1–12.
[45] Anirudh Vemula, Katharina Muelling, and Jean Oh. 2018. Social attention: Modeling attention in human crowds. *In: Proc. IEEE Conf. on Robotics and Automation* (2018).
[46] Julius Venskus, Povilas Treigys, and Jurgita Markevičiūtė. 2021. Unsupervised marine vessel trajectory prediction using LSTM network and wild bootstrapping techniques. *Nonlinear analysis: modelling and control* 26, 4 (2021), 718–737.
[47] Chao Wang and Yuhui Fu. 2020. Ship trajectory prediction based on attention in bidirectional recurrent neural networks. In *2020 5th International Conference on Information Science, Computer Technology and Transportation (ISCTT)*. IEEE, 529–533.
[48] Jingyuan Wang, Chen Yang, Xiaohan Jiang, and Junjie Wu. 2023. WHEN: A Wavelet-DTW Hybrid Attention Network for Heterogeneous Time Series Analysis. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2361–2373.
[49] Ling-Yin Wei, Yu Zheng, and Wen-Chih Peng. 2012. Constructing popular routes from uncertain trajectories. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 195–203.
[50] Guo Xie, Anqi Shangguan, Rong Fei, Wenjiang Ji, Weigang Ma, and Xinhong Hei. 2020. Motion trajectory prediction based on a CNN-LSTM sequential model. *Science China Information Sciences* 63 (2020), 1–21.
[51] Andy Yuan Xue, Rui Zhang, Yu Zheng, Xing Xie, Jin Huang, and Zhenghua Xu. 2013. Destination prediction by sub-trajectory synthesis and privacy protection against such prediction. In *2013 IEEE 29th international conference on data engineering (ICDE)*. IEEE, 254–265.
[52] Peilun Yang, Hanchen Wang, Defu Lian, Ying Zhang, Lu Qin, and Wenjie Zhang. 2022. TMN: Trajectory Matching Networks for Predicting Similarity. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 1700–1713.
[53] Zhou Yang, Heli Sun, Jianbin Huang, Zhongbin Sun, Hui Xiong, Shaojie Qiao, Ziyu Guan, and Xiaolin Jia. 2018. An efficient destination prediction approach based on future trajectory prediction and transition matrix optimization. *IEEE transactions on knowledge and data engineering* 32, 2 (2018), 203–217.
[54] Chin-Chia Michael Yeh, Yan Zhu, Hoang Anh Dau, Amirali Darvishzadeh, Mikhail Noskov, and Eamonn Keogh. 2019. Online amnestic dtw to allow real-time golden batch monitoring. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2604–2612.
[55] Haitao Yuan and Guoliang Li. 2019. Distributed in-memory trajectory similarity search and join on road network. In *2019 IEEE 35th international conference on data engineering (ICDE)*. IEEE, 1262–1273.
[56] Daiyong Zhang, Jia Li, Qing Wu, Xinglong Liu, Xiumin Chu, and Wei He. 2017. Enhance the AIS data availability by screening and interpolation. In *2017 4th International Conference on Transportation Information and Safety (ICTIS)*. IEEE, 981–986.
[57] Mingyang Zhang, Pentti Kujala, Mashrura Musharraf, Jinfen Zhang, and Spyros Hirdaris. 2023. A machine learning method for the prediction of ship motion trajectories in real operational conditions. *Ocean Engineering* 283 (2023), 114905.
[58] Yu Zheng. 2015. Trajectory data mining: an overview. *ACM Transactions on Intelligent Systems and Technology (TIST)* 6, 3 (2015), 1–41.
[59] Ruicheng Zhong, Guoliang Li, Kian-Lee Tan, and Lizhu Zhou. 2013. G-tree: An efficient index for knn search on road networks. *In Proceedings of the 22nd ACM international conference on Information & Knowledge Management* (2013), 39–48.
[60] Alex Zyner, Stewart Worrall, and Eduardo Nebot. 2018. A recurrent neural network solution for predicting driver intention at unsignalized intersections. *IEEE Robotics and Automation Letters* 3, 3 (2018), 1759–1764.