Multimodal Transportation 4 (2025) 100191

Contents lists available at ScienceDirect

Multimodal Transportation

journal homepage: www.elsevier.com/locate/multra

Full Length Article

Maritime vessel movement prediction: A temporal convolutional network model with optimal look-back window size determination

Farshad Farahnakian*, Paavo Nevalainen, Fahimeh Farahnakian, Tanja Vähämäki, Jukka Heikkonen

Department of Computing, University of Turku, Turku, 20500, Finland

ARTICLE INFO

Keywords:
Intelligent maritime transportation
Ship trajectory prediction
AIS data
Deep learning
Clustering algorithm

ABSTRACT

Ship movement prediction models are crucial for improving safety and situational awareness in complex maritime shipping networks. Current prediction models that utilize Automatic Identification System (AIS) data to forecast ship movements typically rely on a fixed look-back window size. This approach does not effectively consider the necessary amount of data required to train the models properly. This paper presents a framework that dynamically determines the optimal look-back window size for AIS data, tailored to user-defined prediction intervals. Initially, a DB-SCAN clustering method, along with various pre-processing techniques, has been employed to efficiently eliminate non-essential data points and address noise in the raw AIS data. Following this, Temporal Convolutional Networks (TCNs) have been trained using the dynamic characteristics of ship movements based on one month of AIS data (April 2023) collected from the Baltic Sea, evaluating various look-back window sizes to identify the optimal size required for predictions. Subsequently, the framework has been tested using an additional AIS dataset in two scenarios: 1-hour and 5-hour predictions. The experimental results indicate that the proposed framework can effectively select the necessary AIS samples for forecasting a ship's future movements. This framework has the potential to optimize prediction services by identifying the ideal look-back window size, thereby providing maritime agents with high-quality and accurate predictions to enhance their decision-making processes.

1. Introduction

The shipping industry plays a vital role in international trade as it is the primary means of transporting goods between countries. Shipping facilitates global trade by connecting markets, prompting economic development, and ensuring the efficient delivery of products. According to the Organization for Economic Co-operation and Development (OECD), and the United Nations Conference on Trade and Development (UNCTAD), over 80% of goods (approximately 11 billion tons annually) are shipped across oceans by various types of ships (United Nations Conference on Trade and Development, 2022). The maritime transportation network encompasses a wide range of ships, including container ships, oil tankers, cruise ships, passenger ferries, and smaller boats like fishing ships. As of 2023, more than 106700 trading ships were navigating the world's seas (Handbook of Statistics, 2023). The statistical data underscores the critical role of the shipping industry in enabling global trade. Therefore, it is imperative to continuously take measures and establish standards to maintain the efficiency, security, and safety of global commerce and transportation networks.

Sailing at sea, particularly in areas with high traffic density, obstructions, and a complex navigation environment, raises the danger of maritime accidents. Therefore, to ensure smooth maritime traffic, prevent collisions, and monitor ships' movements, the 2002 IMO

* Corresponding author.
E-mail address: farfar@utu.fi (F. Farahnakian).

https://doi.org/10.1016/j.multra.2025.100191
Received 7 May 2024; Received in revised form 26 July 2024; Accepted 27 August 2024

Table 1
Categorization of AIS Data.

Category	Description	Examples
Static	Data that remains constant over time per voyage	Ship's identity, type, and dimensions
Dynamic	Data that frequently changes during a voyage	Position, course, speed
Voyage-related	Data related to a voyage or trip	Destination, draught

SOLAS Agreement (Joseph and Dalaklis, 2021) requires most ships over 300 gross tons on international voyages and all passenger ships, regardless of size, to be equipped with an Automatic Identification System (AIS) (Farahnakian et al., 2023). AIS is an automatic tracking system that transmits messages on two Very High Frequency (VHF) channels (161.975 and 162.025 MHz), which are then received by offshore AIS stations and nearby ships (Ribeiro et al., 2023). AIS messages provide detailed, real-time information about a ship's identity, position, course, and speed. The information obtained from AIS data can be classified into three categories: static, dynamic, and voyage-related. Detailed AIS features based on their types are listed in Table 1.

AIS data, containing a rich record of ship movement characteristics over time, offers a valuable time series dataset for maritime applications. Over the past decades, data scientists have actively focused and worked on AIS data, and used the power of novel Machine Learning (ML) and Deep Learning (DL) methods to develop and propose different ship movement prediction systems to enhance the efficiency, security, and safety level of global commerce and transportation networks (Liang et al., 2022). By estimating future movements of ships, the stakeholders would have an opportunity to get a variety of benefits from the prediction system, such as optimizing shipping schedules, reducing transit times, and cutting operating costs for marine trade enterprises (United Nations Conference on Trade and Development, 2022). Additionally, precise movement prediction enhances maritime domain awareness, aids in preventing illegal activities such as smuggling and piracy, and facilitates efficient responses to environmental catastrophes or maritime incidents (Joseph and Dalaklis, 2021). Consequently, reliable ship movement prediction systems are indispensable for maritime authorities, port operators, logistics companies, and other stakeholders to streamline global trade and ensure the safety and sustainability of marine operations.

However, leveraging AIS data effectively for training a ship movement prediction model presents significant challenges. First, ships transmit numerous AIS messages daily during their voyages, creating a high volume of data that prediction systems must handle. These models require to perform pre-processing functions to train the model. To illustrate the extent of AIS data transmitted by ships, Fig. 1 has been generated, and it shows the extensive ships' trajectories (spatiotemporal sequences of AIS messages) sailed in the Baltic Sea on 1st April 2023. Additionally, it is worth mentioning the ship movement characteristics may range from size, hull design, propulsion system, and loading capacity to mission-specific factors such as routes, cargo, and destination ports (Farahnakian et al., 2022; International Convention for the Safety of Life at Sea SOLAS, 2002). For example, a fully loaded bulk carrier would sail differently compared to an empty one, and a fishing boat would exhibit irregular patterns due to fishing operations. Consequently, accurate prediction demands advanced models capable of handling large-scale, multi-dimensional data, extracting meaningful patterns, and adapting to the ever-evolving maritime ecosystem. To demonstrate the diversity of ship types that sailed in the Baltic Sea on 1st April 2023, the bar chart in Fig. 2 depicts the number of ships based on their types.

AIS data of ships is updated frequently, with positions potentially transmitted every two to three minutes. This is unlike many other time series datasets (e.g., finance (Sezer et al., 2020), energy (Bourdeau et al., 2019), and healthcare (Kumar and Susan, 2020)) that are collected at daily or hourly intervals. The high frequency of data collection provides a rich historical dataset for training predictive models. However, a key challenge in marine monitoring systems, especially in a prediction service that responds to various user requests, is to determine the ideal look-back window size for delivering accurate and effective prediction results. This involves taking into account the variability in future steps based on user needs. For instance, a Coast Guard agent may require detailed and high-frequency prediction results for a specific ship to make well-informed, immediate decisions. Conversely, a port authority agent might need a broader overview of ship movements for port management purposes, which does not necessitate as detailed a prediction. Consequently, a longer look-back window is crucial for detailed, short-term predictions, whereas a shorter look-back window can suffice for general, long-term overviews, thereby conserving energy and reducing computational costs.

Additionally, some types of ships, such as passenger and cargo ships, typically follow consistent patterns and routes, thus requiring only a shorter look-back window to predict their movements accurately. An efficient prediction system must intelligently determine the optimal look-back window size based on the specific requirements of the user, adapting dynamically to different scenarios. Using a fixed look-back size may limit the model's access to essential historical data or introduce unnecessary complexity and the risk of overfitting. Thus, optimizing the look-back window size is crucial for enhancing the precision, efficiency, and adaptability of ship movement prediction systems.

In this study, a comprehensive framework is proposed to determine the optimal look-back window size for accurate maritime ship movement prediction using Time Series cross-validation (TSCV) (Bergmeir and Benitez, 2012) and Temporal Convolutional Network (TCN)-based models (Bai et al., 2018a). The process begins with a four-week AIS data, which is stored in a centralized database. This data undergoes several pre-processing steps to ensure its quality and suitability for model training, including missing data handling, outlier detection, normalization, and formatting into a supervised learning problem. The framework then defines a range of potential look-back window sizes (1-day, 2-day, 3-day, and 4-day) and employs TSCV to train and validate the TCN model for each window size to investigate how much AIS data is needed for two prediction time intervals: 1-hour and 5-hour. The window size that yields the lowest average prediction error is identified as optimal. This optimal look-back window size is

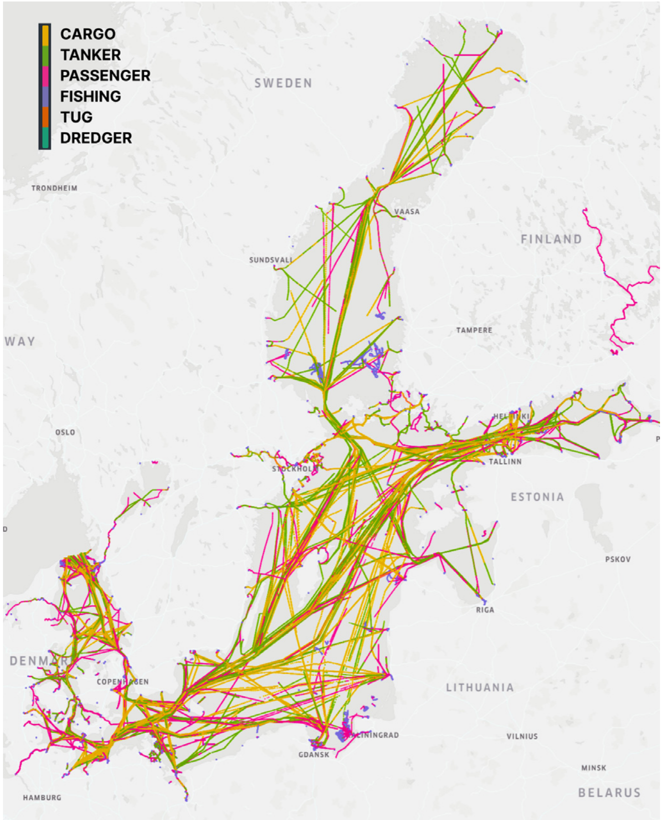

Fig. 1. Maritime ship trajectories in the Baltic Sea on 1st April 2023.

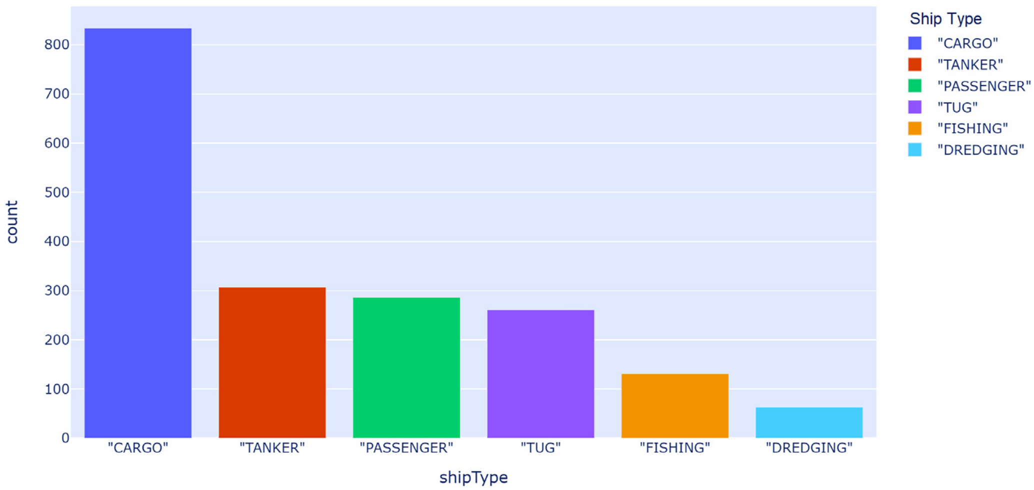

Fig. 2. Number of unique ships based on their types sailed across the Baltic Sea on 1st April 2023.

subsequently used to process real-time prediction requests from maritime agents. Upon receiving a request through an Application Programming Interface (API) (Ofoeda et al., 2019), the AIS data of the target ship is filtered, pre-processed, and formatted before the pre-trained TCN model predicts the ship's future movements. The predicted results are then communicated back to the maritime agent, providing timely and accurate ship movement predictions to enhance maritime situational awareness and operational efficiency. To validate the framework, a Finnish cargo ship is tested by sending its request through API, and a maritime agent is asked to have both the next 1-hour and 5-hour future movement of the cargo ship. Here are the contributions of this study compared to existing studies:

1. Unlike previous studies, this work systematically investigates and determines the optimal look-back window size for specific prediction intervals. This addresses the crucial question of how much AIS data is needed for a specific future step ahead, enhancing the accuracy and reliability of ship movement predictions.
2. This study employs a clustering algorithm to detect loitering ship behaviors and remove AIS samples generated while a ship is loitering. In addition, a robust pre-processing pipeline that handles missing data, outlier detection, and normalization, ensures the high quality and consistency of the AIS data used for training the predictive models has been used.
3. The framework is designed to be user-centric, allowing maritime agents to specify prediction intervals and receive customized predictions based on the optimal look-back window size. This flexibility is crucial for practical applications in diverse maritime scenarios, yet it is not commonly found in existing studies.
4. The study demonstrates the framework's effectiveness in specific scenarios, such as predicting the movements of ships over 1-hour and 5-hour intervals.

The paper is structured as follows. Section 2 reviews previous studies on ship movement prediction. Section 3 defines time series data and TSCV, including the dataset used, the pre-processing phase, and evaluation metrics. Section 4 introduces the proposed framework for determining sufficient look-back window sizes, outlining theoretical foundations and methodologies. Section 5 presents the results of applying the framework to ship movement prediction. Finally, Section 6 summarizes the findings, highlighting the limitations and effectiveness of the proposed framework.

2. Related work

The prediction of ship movements using AIS data employs two overarching methodologies: data-driven and model-based approaches (Alizadeh et al., 2021). Data-driven methods primarily focus on extracting insights directly from the data without imposing any predefined assumptions about the underlying system (Spiliopoulos et al., 2018). They include techniques like ML and clustering, where algorithms learn from the data's inherent structure and patterns to make predictions. ML models, such as random forest (Zhang et al., 2020), neural networks (Zhou et al., 2019), or support vector machines (Gang et al., 2016), excel in handling high-dimensional AIS data and capturing complex, non-linear relationships. Clustering, on the other hand, partitions ships into groups based on similar movement patterns or other shared characteristics, enabling tailored prediction strategies for different clusters (Virjonen et al., 2018). Conversely, model-based methods rely on explicit mathematical or computational models to describe ship movements. These models can range from deterministic or probabilistic models (Wei et al., 2023), which use physical laws of motion and maritime regulations to estimate ship movement, to statistical models like regression analysis (Rong et al., 2022) which draws on historical AIS data to identify trends and patterns. Fig. 3 summarises different approaches used for ship movement prediction.

Recently, advancements in DL algorithms, a subset of data-driven methods, have significantly improved the accuracy and efficiency of ship trajectory prediction and maritime surveillance (Capobianco et al., 2021). These improvements are largely due to the extensive availability of AIS data, which provides a rich source of real-time information on ship movements, allowing DL models to learn and make precise predictions. DL-based methods offer several distinct advantages over other methods when it comes to predicting future ships' movements. Firstly, DL methods are adept at learning from the inherent structure of the data itself, identifying hidden patterns and nonlinear relationships that may be missed by traditional models (Lara-Benitez et al., 2021). This capability enables them to handle the high dimensionality of AIS data and to adapt to new data, improving their predictive accuracy over time. Secondly, DL-based models, excel at dealing with the curse of dimensionality, a common problem in maritime ship movement prediction due to the vast number of influencing factors (Han et al., 2019). These models can effectively navigate the large input space to make accurate predictions, which might not be feasible with model-based methods. Lastly, while model-based methods can struggle with sudden or unpredictable changes in ship movement patterns, DL-based methods, given their inherent ability to learn from the data, can often respond more quickly and accurately to these changes (Han et al., 2019).

In Kim et al. (2020), they built a neural network with three Long Short-Term Memory (LSTM) layers that can monitor the first 10 min of the ship's status and forecast its position after 20 min. In another work (Gao et al., 2018), the forecasting of maritime ship actions is tackled, where bidirectional LSTM and Recurrent Neural Networks (RNNs) are directly employed for real-time prediction. Meanwhile, in Yu et al. (2020), two different methods have been proposed for data association. They first trained an RNN-based model for ship trajectory perdition and combined the trained model with a binary classifier for tracking messages. Then, they used an auto-encoder for ship movement reconstruction and did data association with the error of the reconstruction method.

In Alam et al. (2024), historical AIS data from the US West Coast was used to cluster shipping routes for different ship types, such as cargo and tanker ships, to improve prediction accuracy. This method allows ML models to focus on similar ship types. Experiments showed that the Random Forest algorithm achieved the lowest distance errors of 1.2 km for short-term (30 min) ship movement predictions. Another study (Ren et al., 2019) proposed an enhanced LSTM navigation prediction model based on the attention mechanism. The model preprocesses AIS data to address missing values and combines dynamic ship navigation characteristics with time series data (longitude, latitude, heading, speed, ship heading, and time increment) to establish a dynamic forecasting model using LSTM. Evaluated on NOAA public AIS data from 2017, the system demonstrated effective prediction for timestamps ranging from 10 to 30 min. In Farahnakian et al. (2023), two approaches for short and long-term ship movement prediction were proposed and tested using two months of real-world AIS data. Results showed high accuracy for short-term predictions and improved performance metrics for long-term predictions in terms of computational speed and memory use.

The study (Zhao et al., 2023a) introduced a Graph Attention Network (GAT) combined with LSTM to predict ship trajectories, emphasizing accuracy and reliability by exploiting spatial and temporal correlations. GAT extracts spatial features of ship trajectory

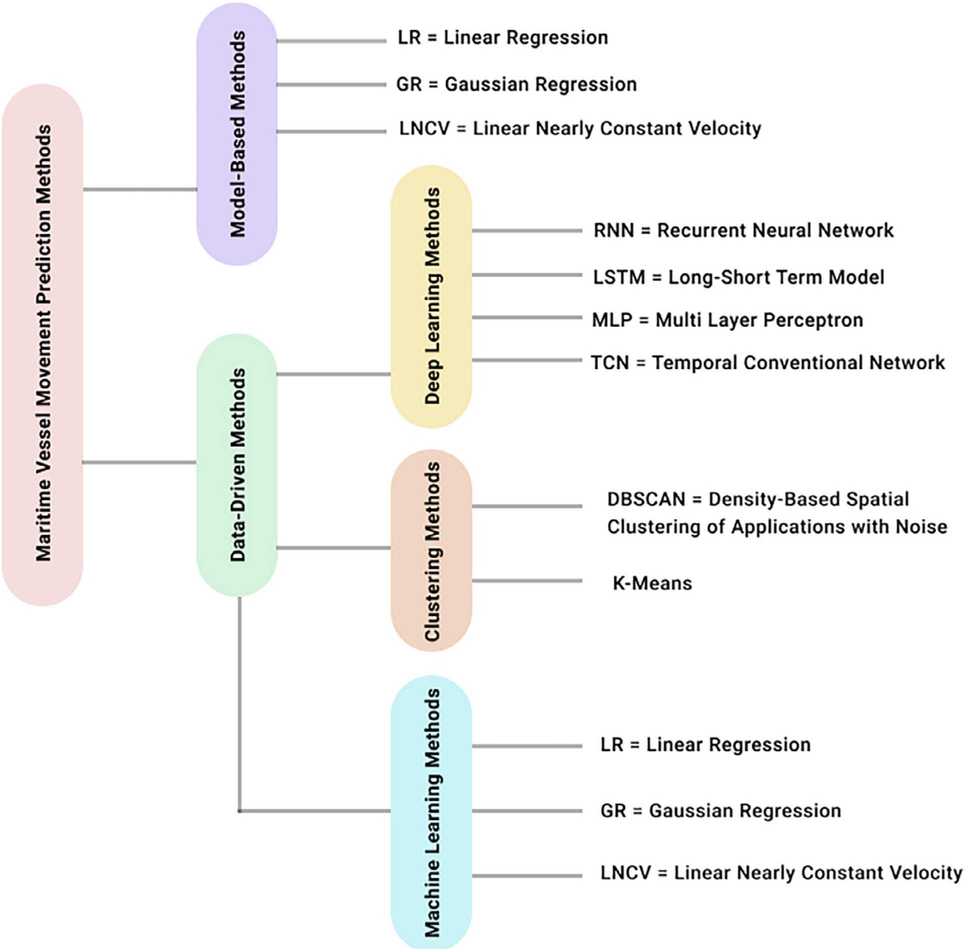

Fig. 3. Techniques utilized for the ship movement prediction task.

data, while LSTM learns temporal features, resulting in precise predictions. The approach was tested on three ships in China with prediction intervals of 10s, 30s, and 1 min. Another article (Liu et al., 2023) presented a multi-task DL model for both ship movement prediction and collision risk prediction. The model uses an encoder-decoder architecture with LSTM layers. The encoder processes historical data to obtain navigation characteristics, and the decoder uses this information to predict ship trajectories and collision risks. The model was validated using 28,869 AIS sequences from Marinecadastr.gov, demonstrating its effectiveness. The paper (Tian and Suo, 2023) suggested a solution for a ship trajectory prediction model based on a different long short-term memory neural network (D-LSTM). A new D-LSTM unit is produced by combining different variables with an existing LSTM unit to form a new D-LSTM network unit. The neural network could perform single-step prediction by forecasting the eleventh trajectory point based on the first ten. To evaluate the suggested approach, AIS data from ships in the Beijing-Hangzhou Canal from September to November 2021 were used.

Furthermore, the study (Nguyen et al., 2018) introduces a variational RNN structure, referred to as GeoTrackNet, designed to glean insights into the concealed behavior of ships from AIS data streams. The principal applications of this model are for reconstructing navigation tracks and identifying any irregularities. In Yang et al. (2022), the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) method which is one of the popular clustering techniques has been used first for ship trajectories, and then the LSTM-based model has been employed to train and predict ship movements. In summary, Table 2 presents other existing bodies of work, categorizing them based on the prediction methods employed for ship movement forecasting, the future steps they're capable of predicting, and the geographical area of study.

It has been observed that a significant amount of research in the maritime ship movement prediction field relies heavily on a predetermined look-back window. However, there is a lack of understanding regarding the optimal volume and time range of AIS data required for accurate forecasting. This knowledge gap could result in suboptimal or inaccurate forecasts. In some cases, insufficient data may fail to capture the full picture of ship movement, leading to the omission of critical patterns or trends. On the other hand, feeding too much data into the model may increase computational complexity without necessarily improving prediction accuracy, resulting in the waste of resources. Therefore, investigating the optimal volume and range of AIS data required for accurate forecasting

Table 2
Model parameters and time.

Ref	Future step	Method	Research Area
Zhao et al. (2023a)	50 min	CNN	Yeosu, Korea
Liu et al. (2023)	5 min	CNN, LSTM	Singapore
Tian and Suo (2023)	15 min	MLP	Poland
Nguyen et al. (2018)	4 hrs	MLP	Aegean, Greece
Yang et al. (2022)	1 min	LSTM	Yangtze, China
Kim and Lee (2018)	20 min	MLP	Nanpu, China

is vital. This could enhance the model's efficiency and usability in real-world scenarios, bridging the gap between theoretical research and practical application in the maritime industry.

3. Preliminary study

Maritime ship movements exhibit distinct patterns and trends over time. To precisely predict future movements, it is essential to analyze data points collected at both regular and irregular intervals. This section provides an overview of time series data, setting the stage for the specific methodologies discussed in the next sections. Additionally, the concept of TSCV is explored, which plays a crucial role in determining the optimal window size for analyzing past ship movements to achieve accurate predictions.

3.1. Time series data

A time series is a sequence of repeated measurements of a given set of K variables over a specific time (Agrawal and Adhikari, 2013). This concept finds application in diverse scenarios such as fluctuating stock prices, changing precipitation levels, traffic volumes in communication or transportation networks, and, pertinent to this study, ship movements during a voyage. Mathematically, a time series can be represented as $[x_1, x_2, \ldots, x_n]$. Here, x_t, observed at time t, symbolizes a vector of K random variables. The variable K specifically denotes the number of different attributes or dimensions being measured at each time point (Cochrane, 1997). In the context of maritime ship movement, K could include variables such as latitude (Lat), longitude (Lon), speed over ground (SOG), course over ground (COG), and draught (D). A ship movement or a ship trajectory ($V(t)$), which consists of a series of vectors, can be defined based on the time series definition as follows:

$$V(t) = \left[\left(Lat_t, Lon_t, SOG_t, COG_t, D_t\right)\right] \tag{1}$$

where $V(t)$ represents the state vector of the ship at time t capturing latitude, longitude, speed, course, and draught.

An AIS dataset collected from a specific geographical region over a certain period contains numerous data points representing these variables. Therefore, AIS data for a ship with a specific International Maritime Organization (IMO) ID can be defined as follows:

$$AIS_{IMO} = [V(1), V(2), ..., V(n-1), V(n)] \tag{2}$$

where each $V(t)$ represents the comprehensive state of the ship at time t, and n is the number of AIS reports transmitted by a ship during its journey.

3.2. Time series cross validation

Cross-validation is a statistical analysis and machine-learning approach used to evaluate the robustness and generalizability of a model's performance and prevent overfitting (Diana and Tommasi, 2002). Time series data should be validated using a special type of cross-validation known as TSCV (Bergmeir and Benitez, 2012), such as AIS data for ship movement prediction. This is because time series data are sequential and the temporality and order of the data points are important. In the TSCV method, the data is first divided into a minimal ”training set” and the remaining data as the ”testing set.” Following this initial training set, the model is trained and then verified using the subsequent data points in the testing set. The process is then repeated after expanding the training set to incorporate this testing data point. Fig. 4 shows how the TSCV method splits a time series data for the evaluation phase. This ”rolling” or ”look-forward expanding” window strategy guarantees that the temporal structure of the data is always preserved and that the model is always evaluated on unseen data.

4. Methodology

DL models employ historical data to recognize a functional correlation between input characteristics and future instances of the desired variable (Zeng et al., 2022). This established model then becomes capable of forecasting the desired variable at subsequent moments in time. When presented with a time series, $[x_1, x_2, \ldots, x_n]$, where each x_t represents a vector containing m observed input features at time t, the time series can be conceptualized as a two-dimensional structure with dimensions $n \times m$. Here, n denotes the total number of time steps, and m represents the number of observed features at each time step. The primary challenge in formulating such a model lies in accurately predicting a target variable y_{t+k} for a future moment, $t + k$, using the historical data up until time $t - 1$, represented as $[..., x_{t-2}, x_{t-1}]$. This task involves several complex factors:

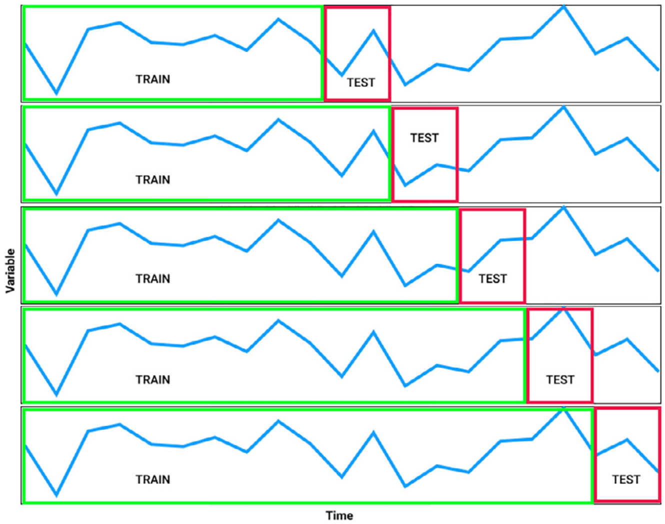

Fig. 4. Five-fold splitting by using time series cross-validation.

1. The complexity of time series data, especially in maritime contexts, which often exhibits non-linearity and strong temporal dependencies (Zhao et al., 2023b).
2. The need to choose an appropriate look-back window size (w) that is sufficiently large to capture relevant historical patterns without causing overfitting (Keelawat et al., 2021).
3. Adapting the model to different user-specified forecasting requirements, represented by $t+k$ (Zhao et al., 2023b).

In DL models, a fixed-length sliding time window of size w is initially used to maintain a consistent and uniform input data structure across all training and prediction iterations. This standard approach in time series analysis is crucial for effectively managing the temporal dependencies in AIS data. Although the use of a fixed-length window is common practice, its application to the specific dynamics and complexities of maritime data is essential for the model's effectiveness.

In this paper, the framework introduces an innovative aspect by adjusting the look-back window size during development. This adjustment aims to find the optimal window size that minimizes the average error rate for predictions based on user-defined future time intervals. While this process might seem to contrast with the concept of a fixed-length window, it is a strategic component of model tuning and optimization. It enhances the model's adaptability and precision in predicting maritime ship movements without compromising operational consistency. The framework's innovation lies in its tailored application to AIS data for maritime surveillance, adapting to varying user requirements, which is a unique challenge in this domain. This approach combines the reliability of a fixed-length window with the flexibility of window size adjustment, grounded in a data-driven methodology that complements the expertise of maritime agents.

This method strikes a balance between capturing sufficient historical context and maintaining computational efficiency. Fig. 5 illustrates the various look-back window sizes (w) experimented with for a given step ahead ($t+k$). The functional relationship used by the DL models for ship movement prediction can be mathematically described as follows:

$$\hat{y}_{t+k} = f_k\left(x_{t-w}, \ldots, x_{t-1}, y_{t-w}, \ldots, y_{t-1}\right) \tag{3}$$

where $\hat{y}_{t+k}$ forecasts the target variable for time $t+k$; k signifies how far into the future the desired variable needs to be projected; $y_{t-w}, \ldots, y_{t-1}$ includes the target values observed from time $t-w$ to $t-1$; $x_{t-w}, \ldots, x_{t-1}$ comprises the vector of m input features noticed from time $t-w$ to $t-1$; f_k symbolizes the function grasped by DL models; m represents the count of input features; and w defines the dimension of the window utilized for input.

In this paper, the features used as an input and the expected output are defined as follows:

$$Input_t = \left[Lat_t, Lon_t, SOG_t, COG_t, D_t\right] \tag{4}$$

$$Output_{t+k} = \left[Lat_{t+k}, Lon_{t+k}\right] \tag{5}$$

where Lat_t is latitude, Lon_t is longitude, SOG_t is speed over ground, COG_t is course over ground, and D_t is the vertical distance between the waterline and the keel (bottom of the hull) of the targeted ship at time t. $Output_{t+k}$ represented the predicted position of the ship at a future time $t+k$.

4.1. Model architecture

The TCN algorithm, presented in 2018, represents a specialized form of the convolutional neural network (CNN), innovatively appropriate to address time series challenges (Bai et al., 2018b). Building upon the foundational architecture of the traditional

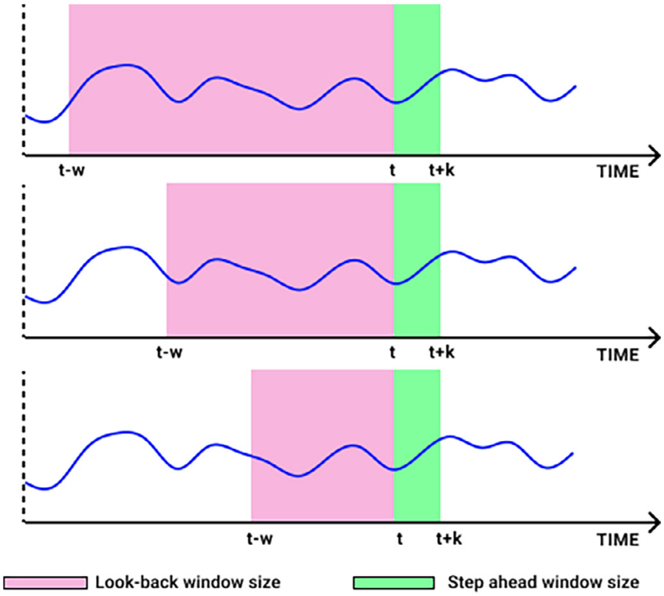

Fig. 5. Employing a look-back window with size w to create the training dataset for predicting future with size k (future step window).

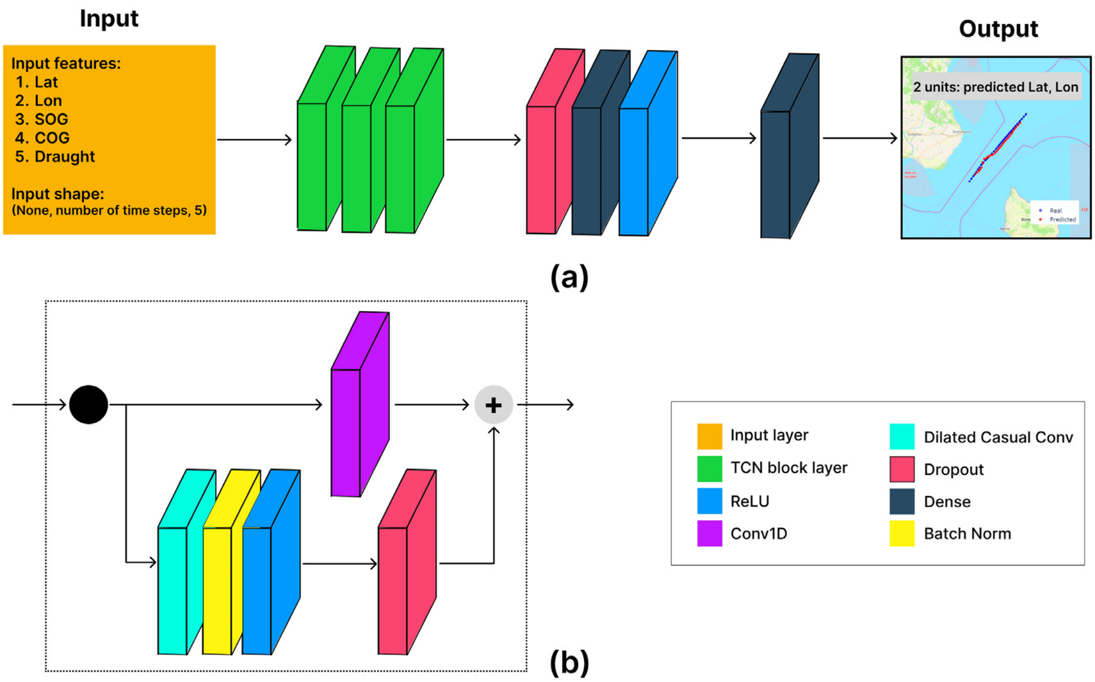

Fig. 6. (a) Architecture of prediction model. (b) A TCN block layer.

CNN, the TCN introduces novel components like causal convolution, dilated convolution, and residual connections to craft a unique network structure. The TCN block architecture is shown in Fig. 6(b). While CNNs are praised for their powerful feature extraction capabilities, they often falter in preserving the temporal dynamics inherent to time series data. TCN, stemming from a one-dimensional convolutional network, alleviates this limitation by augmenting the existing feature extraction strength of CNNs with adaptations that make it adept at handling time series nuances (Luo et al., 2021). This marks a departure from commonly utilized recurrent neural network algorithms like LSTM for processing ship motion attitude time series. By offering an alternative pathway for the hierarchical extraction of ship motion attitude characteristics, the TCN algorithm constitutes a significant advancement in the field, opening up new avenues for exploration and application.

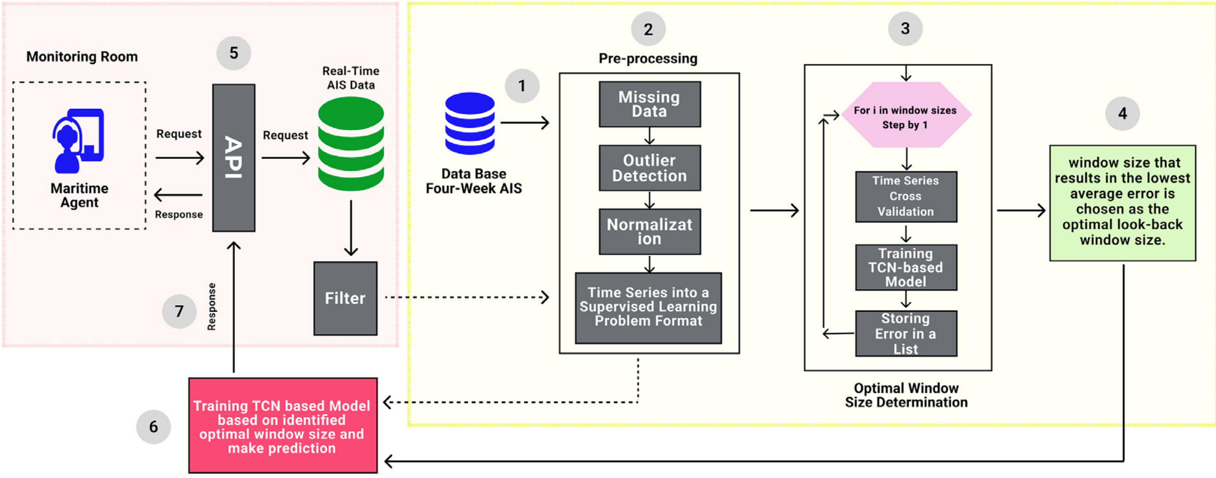

Fig. 7. Maritime ship movement prediction framework using TCN-based model and optimal window size determination.

The architecture of the model, depicted in Fig. 6(a), employs a Sequential approach, designed to process time series data for maritime ship movement prediction. The model comprises several layers, each with a specific function in the predictive process:

1. The first three layers are TCN layers, each designed to capture temporal dependencies at various scales. These layers process input sequences shaped by the look-back window size, covering five key features: SOG, COG, Draught, Latitude, and Longitude.
2. Following the TCN layers, a Dropout layer (Srivastava et al., 2014) with a rate of 0.2 is applied to mitigate overfitting by randomly omitting a fraction of the inputs during training, thus enhancing the model's generalization capability.
3. The model concludes with a fully connected dense layer of 128 units with ReLU activation for forming preliminary predictions, followed by another dense layer that outputs the final predictions for latitude and longitude.

Furthermore, the Adam optimizer (Zhang, 2018) is employed for its advantages in handling sparse gradients and adapting the learning rate based on the algorithm's internal state.

4.2. Hyperparameter selection

The selection of appropriate hyper-parameters is crucial for the performance of ML models (Hu and Xiong, 2023). Hyper-parameters such as the dropout rate and learning rate significantly impact the model's ability to generalize and converge. The dropout rate helps mitigate overfitting by randomly omitting a fraction of the inputs during training, enhancing the model's generalization capability. The learning rate controls how much the weights of the network are adjusted with respect to the loss gradient, balancing convergence speed and training stability.

To determine the optimal values for these hyper-parameters, a Grid Search method was employed. The grid search evaluated various combinations of dropout rates [0.1, 0.2, 0.3, 0.4] and learning rates [0.001, 0.01, 0.1] to identify the configuration that provided the best performance. This process is essential as it systematically explores the hyper-parameter space to find the best settings for the model, ensuring it performs optimally on the task at hand.

4.3. Proposed framework for optimal look-back window size determination

The framework aims to identify the optimal look-back window size for ship movement prediction, which is determined based on the step ahead or future step value specified by the user. Once the optimal window size is determined, it is employed to process real-time prediction requests from maritime agents. Fig. 7 illustrates the flowchart of the proposed framework and summarizes the steps for discovering the optimal window size for the ship movement prediction task. The detailed descriptions of the steps in the proposed framework are as follows:

1. The process begins with a four-week AIS data. This historical data is stored in a centralized database.
2. **Pre-processing:** The collected AIS data undergoes several pre-processing steps to ensure its quality and suitability for model training:
   (a) **Missing Data Handling:** Any missing data points in the AIS dataset are identified and imputed or removed to maintain data integrity.
   (b) **Outlier Detection:** Outliers in the dataset are detected and managed to prevent them from skewing the model's learning process.
   (c) **Normalization:** The data is normalized to a standard scale, which helps improve the model's performance by ensuring consistent data ranges.

   (d) **Time Series Formatting:** The data is then formatted into a supervised learning problem, converting the time series data into input-output pairs suitable for model training.
3. **Optimal Window Size Determination**
   (a) A range of look-back window sizes (1-day, 2-day, 3-day, 4-day) is defined. The TCN-based model is trained and validated using TSCV for each window size for two prediction intervals: 1-hour and 5-hour.
   (b) For each look-back window size, TSCV is performed, splitting the data into training and validation sets across different folds.
   (c) The TCN-based model is trained on the training sets and validated on the validation sets for each fold.
   (d) The prediction error for each fold is calculated and stored.
   (e) The average prediction error for each look-back window size is computed by averaging the errors across all folds.
4. **Optimal Look-back Window Size Selection:** The window size that results in the lowest average error is chosen as the optimal look-back window size.
5. **Real-time Prediction Request Handling:**
   (a) A maritime agent sends a prediction request through an API. This request includes the AIS data of the target ship.
   (b) The AIS data of the target ship is filtered and pre-processed (missing data handling, outlier detection, normalization).
   (c) The pre-processed AIS data is formatted into the supervised learning problem format.
6. **Prediction Using TCN Model:** The TCN-based model is trained with the identified optimal look-back window size and processes the AIS data to predict the ship's future movements.
7. **Response to Maritime Agent:** The predicted results are sent back to the maritime agent through the API, providing accurate and timely ship movement predictions.

The framework is designed with a user-centric approach, allowing maritime agents and analysts to customize the predictive model according to their specific operational contexts. Features such as customizable look-back window sizes and user-defined future prediction intervals make the model adaptable to a wide range of maritime scenarios. By integrating real-time AIS data through an API, the framework ensures the provision of current and relevant information, which is crucial for timely and informed decision-making. This user-centric approach not only enhances the practical applicability of the framework but also ensures that the predictive analysis is directly aligned with the users' specific requirements.

4.4. Data

A large amount of AIS data, comprising both static and dynamic information, was collected from two publicly available APIs. The Finnish transport infrastructure agency's "digitraffic.fi" website was the source of information related to marine activities such as marine warnings, harbor schedules, and ship location AIS messages for ships in the Baltic Sea. The data collection period for this study was from April 1st to April 30th, 2023, resulting in a database size of approximately 11 GB. Separate APIs were used to extract static and dynamic ship attributes which were then merged to obtain a more comprehensive understanding of the data. The integration of the data followed standards set by the IMO ID. Timestamps, latitude, longitude, SOG, and COG were among the variable components encapsulated by the AIS data, while MMSI, ship name or call sign, ship type, IMO number, draught, width, and length were among the unchanging elements.

4.5. Pre-processing

Data pre-processing is a crucial initial step in data analysis and mining, especially with real-world AIS data that often contains inconsistencies, errors, and incomplete information. Several pre-processing techniques were employed to refine the raw data for use in a TCN-based model to discern patterns. The AIS dataset, collected over one month, initially contained 3,143,734 observations and 13 attributes. After the initial division of the data, the MMSI numbers were closely inspected, and rows with invalid nine-digit MMSI codes were discarded. Rows with more than five missing values (approximately 50%) were also excluded. The study then focused on specific ship types (such as cargo, tanker, passenger, fishing, tug, and dredging), and the dataset was filtered accordingly. Further precautions were taken to eliminate samples where Longitude values exceeded 180 or were less than -180, and where Latitude values were beyond 90 or below -90. Additionally, the COG value, representing the ship's directional movement, was scrutinized, and rows with invalid COG (greater than 360 or below 0) were removed. These measures were essential in maintaining the integrity and relevance of the data for the subsequent modeling process.

After the data pre-processing phase, it was identified that specific data points associated with certain ship types, notably cargo and tankers, were not contributing constructively to the TCN-based prediction model. These ships exhibited a distinctive pattern of behavior, termed "ship loitering," within defined spatial confines. Fig. 8 illustrates the loitering behaviors of these two ship types. Such behavior is predominantly observed around ports and can be attributed to various operational factors, including entry permissions, border control procedures, peak maritime traffic hours, and passport checks. Furthermore, during periods of ship loading and unloading, a ship may remain docked for extended durations, during which its AIS records data at frequent intervals (every 2-3 min), generating a substantial volume of duplicative AIS messages.

To remove the impact of these non-essential data points and thereby enhance the accuracy and response time of predictive modeling, the DBSCAN clustering algorithm (Deng, 2020) was implemented. The effectiveness of this algorithm depends on choosing two parameters optimally: ϵ and $min\_samples$. For the ϵ value, the median of the distances between consecutive AIS location points was calculated, establishing a standard threshold distance representative of typical ship movement per AIS data transmission. The

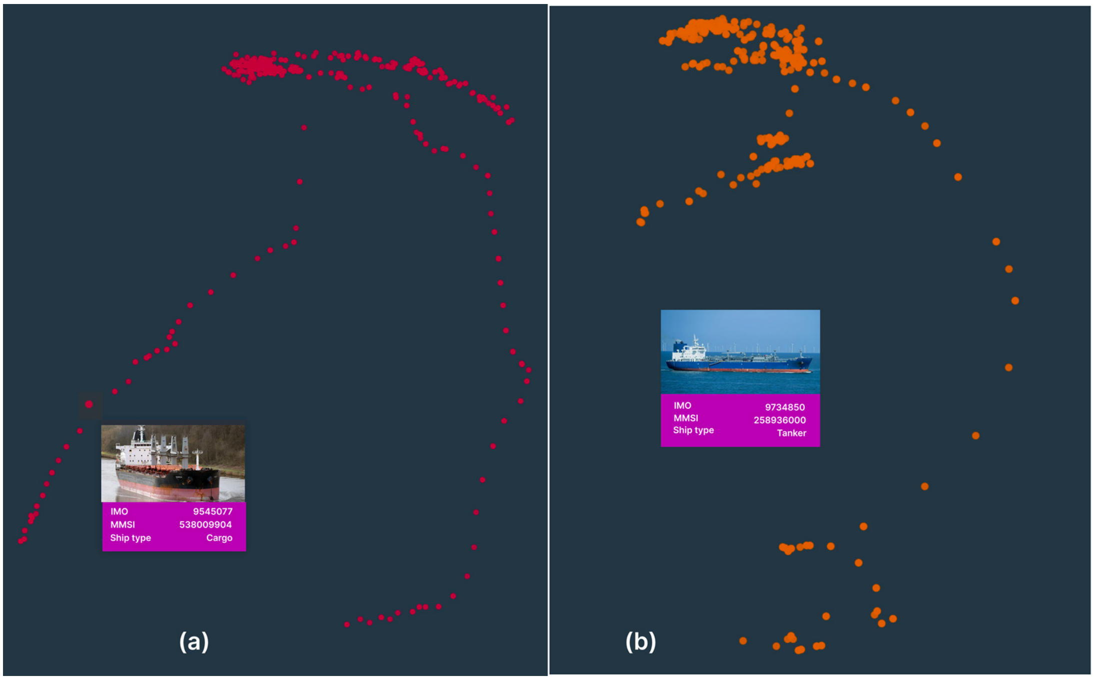

Fig. 8. Two types of ships: (a) cargo located near Reposaari port, and (b) tanker located near Uusikaupunki port in Finland, which are engaged in loitering activities.

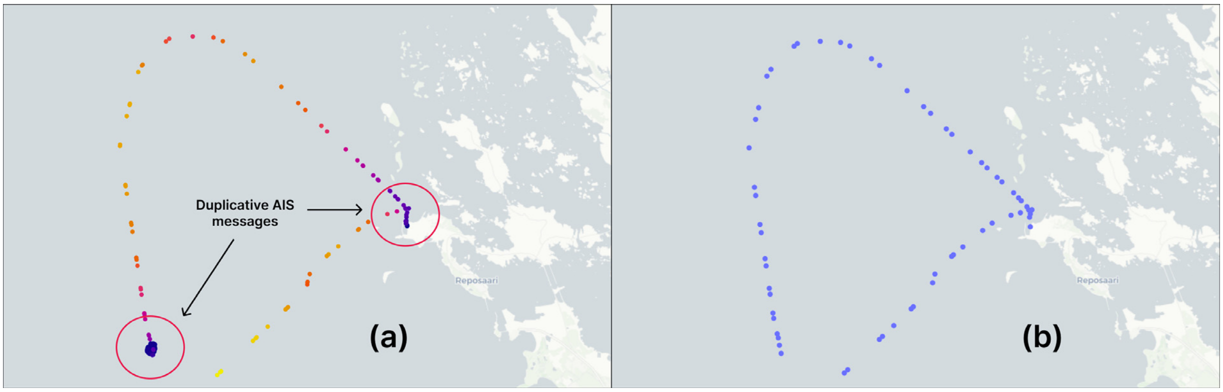

Fig. 9. (a) original AIS data before pre-processing (b) AIS data of the ship after applying the DBSCAN clustering method.

min_samples parameter was set to 1, ensuring that each data point was either integrated into a cluster or formed an independent cluster, thus avoiding the classification of any data points as noise. Fig. 9 shows the trajectory of a ship identified by its IMO number 9545077, both before and after applying the DBSCAN algorithm.

Furthermore, the spline interpolation (Zaman et al., 2023) method was used to generate the dataset regularly, resampling the information to create an AIS dataset reported every 3 min. This step ensured temporal consistency and aided in handling missing or irregularly spaced data. In the final step of the pre-processing stage, the MinMax scaler function (Bisong, 2019) was applied to normalize the values between 0 and 1, ensuring that different features have comparable scales and contribute evenly to the model's learning process. It should be noted that the MinMax scaler function fits the scaler only in the training set to ensure that information from the test set does not leak into the training process. The refined dataset contained 3,098,712 observations with 13 attributes, presenting a well-structured and standardized dataset ready for training the TCN-based model.

4.6. Evaluation metric

To evaluate the maritime ship prediction model, three distinct yet complementary metrics were utilized to assess the performance and accuracy of the predictions.

- **Mean Square Error (MSE):** It is a widely used metric that computes the average squared differences between the predicted and actual values. The MSE is mathematically described as:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{6}$$

- **Root Mean Square Error (RMSE):** RMSE is the square root of the MSE, providing an interpretable measure in the same units as the target variable. The RMSE formula is given by:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{7}$$

- **R-squared (R^2) score:** The R^2 score, also known as the coefficient of determination, quantifies the proportion of the variance in the dependent variable that is predicted by the independent variables. While an R^2 value of 1 indicates a perfect fit and values closer to 1 indicate a better fit, the R^2 score can range from negative infinity to 1. Negative values indicate that the model performs worse than a horizontal line representing the mean of the observed values. The equation for R^2 is:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{8}$$

5. Results

5.1. Grid search results

In this subsection, the grid search results for determining the optimal hyper-parameters (dropout rate and learning rate) for the TCN model and LSTM algorithm that is used for comparison are presented. The grid search was conducted over dropout rates [0.1, 0.2, 0.3, 0.4] and learning rates [0.001, 0.01, 0.1]. The evaluation metrics used were the mean squared error (MSE) for 1-hour and 5-hour predictions across different look-back window sizes. For the 1-hour prediction, the grid search results are summarized in Fig. 10(a). The plot indicates that the optimal dropout rate of 0.2 and learning rate of 0.01 provide the lowest MSE values across different look-back windows for the TCN model. This configuration balances the trade-off between overfitting and convergence speed, ensuring stable and accurate predictions. Similar trends are observed for the LSTM model, with a dropout rate of 0.2 and a learning rate of 0.01 yielding the most favorable results. For the 5-hour prediction, the grid search results are summarized in Fig. 10(b). The results show that a dropout rate of 0.2 and a learning rate of 0.01 consistently result in lower MSE values for the TCN model, confirming the robustness of this configuration for longer-term predictions. The optimal hyper-parameters for the LSTM model also align with those of the TCN model, further validating the choice of parameters.

5.2. Look-back window size results

In this subsection, the results obtained from training models on 4-week AIS data across different look-back window sizes are presented in this section. The prediction intervals selected for the experimental part are 1-hour and 5-hour. The look-back window size determination process involved defining a range of look-back window sizes, and training and validating the TCN-based model using TSCV for each window size. The detailed steps included splitting the data into training and validation sets across different folds, training the TCN model on the training sets, validating the model on the validation sets, calculating and storing the prediction error for each fold, computing the average prediction error for each look-back window size, and identifying the window size that yielded the lowest average prediction error. This optimal look-back window size was then selected for future predictions. It should be noted that the experiments utilized look-back window sizes of w= 1 day (480 intervals), 2 days (960 intervals), 3 days (1440 intervals), and 4 days (1920 intervals). Each interval represents a 3-minute time frame. This calculation is based on the fact that the AIS data is reported every 3 min, resulting in the number of intervals per day being determined by multiplying the number of days by 24 (the number of hours in a day), then by 60 (the number of minutes in an hour), and finally dividing by 3.

The training was performed over 150 epochs with a batch size of 64 using the Adam optimizer and a learning rate of 0.01. The model was trained on a Windows workstation with 11th Gen Intel(R) Core(TM) i7-11800H CPU processor running 2.30 GHz and with 64.0 GB of memory. Python version 3.8 has been used as the programming language.

Tables 3 and 4 present the results of experiments for predicting 1-hour and 5-hour ship movements into the future, respectively, across defined look-back window sizes. The experiments compare the performances of the TCN model, Linear Regression (LR), and LSTM networks. For 1-hour predictions, the TCN model with a two-day look-back window shows the most effective performance, achieving the lowest RMSE = 0.035, MSE = 0.0012, and R^2 score = 0.98. This finding underscores the importance of an appropriately sized look-back window to capture vital patterns for accurate short-term predictions. In the case of 5-hour predictions, a four-day window is optimal for the TCN model, resulting in RMSE = 0.118, MSE = 0.0140, and R^2 score = 0.95. These results suggest that a broader window is beneficial for longer-term predictions. The inclusion of LSTM in the analysis provides an intermediate perspective between the simplicity of LR and the complexity of TCN, further enriching the understanding of model performance in maritime ship movement prediction. While LR, due to its linear nature, may be limited in capturing complex non-linear relationships in time-series

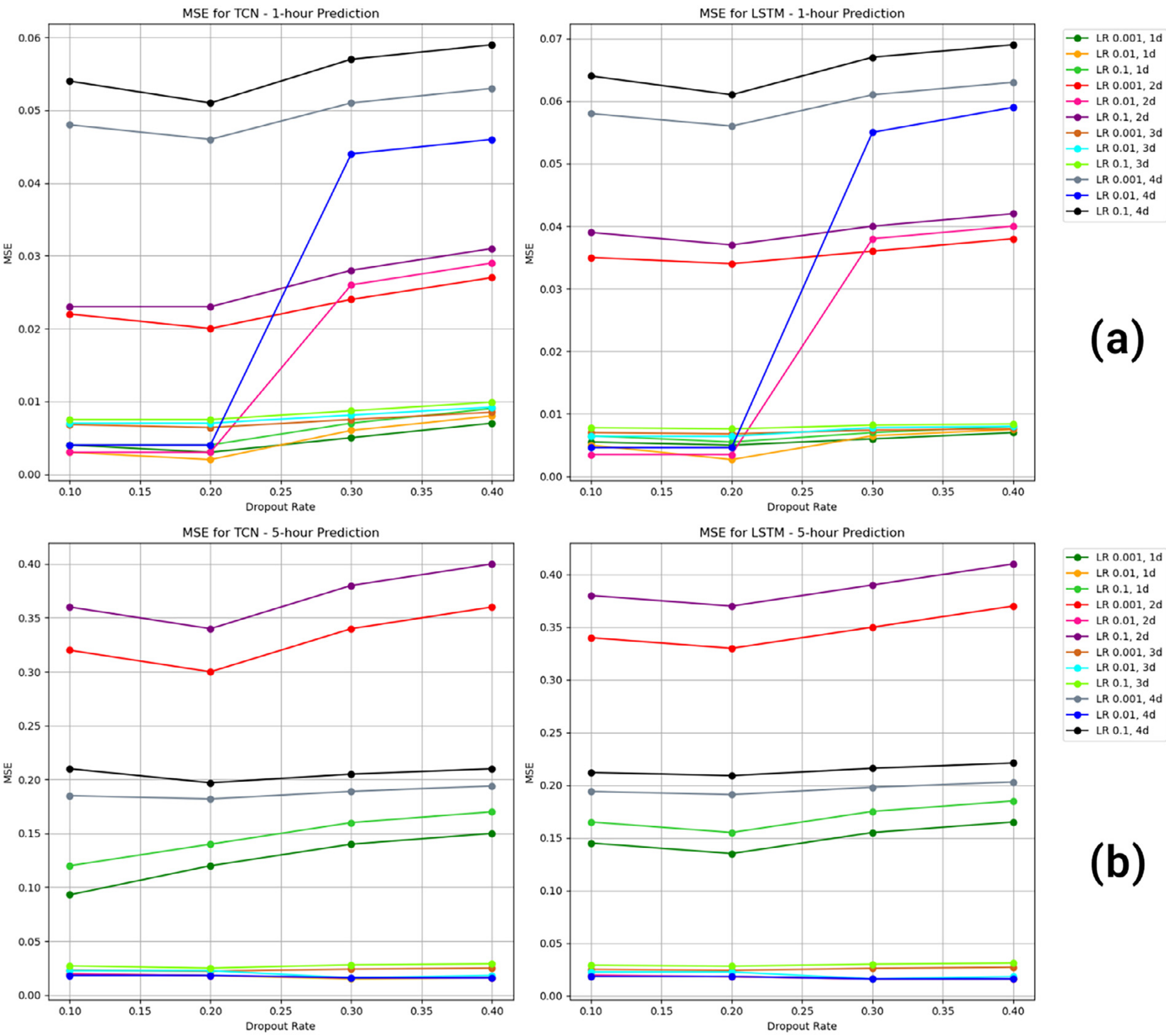

Fig. 10. Grid search results for 1-hour prediction (a), and 5-hour prediction (b) with different dropout rates and learning rates.

Table 3
Comparison of LR, LSTM, and TCN model performances for 1-hour ship movement prediction across different look-back windows.

Look-back Window	Metrics	LR	LSTM	TCN
1d	RMSE	0.075	0.052	0.045
	MSE	0.0056	0.0027	0.0020
	R^2	0.89	0.94	0.96
2d	RMSE	0.115	0.041	**0.035**
	MSE	0.0132	0.0017	**0.0012**
	R^2	0.85	0.96	**0.98**
3d	RMSE	0.675	0.058	0.052
	MSE	0.455	0.0034	0.0027
	R^2	0.32	0.95	0.97
4d	RMSE	0.405	0.049	0.046
	MSE	0.164	0.0024	0.0021
	R^2	0.60	0.96	0.97

Table 4
Comparison of LR, LSTM, and TCN model performances for 5-hour ship movement prediction across different look-back windows.

Look-back Window	Metrics	LR	LSTM	TCN
1d	RMSE	0.275	0.210	0.215
	MSE	0.0756	0.0441	0.0462
	R^2	0.70	0.91	0.92
2d	RMSE	0.510	0.130	0.125
	MSE	0.2601	0.0169	0.0156
	R^2	0.64	0.92	0.94
3d	RMSE	0.880	0.140	0.120
	MSE	0.7744	0.0196	0.0144
	R^2	0.34	0.93	0.94
4d	RMSE	0.460	0.125	**0.118**
	MSE	0.2116	0.0156	**0.0140**
	R^2	0.22	0.93	**0.95**

data, the TCN model, designed to handle sequential data with intricate temporal patterns, proves more adept. Similarly, LSTM models, known for their ability to capture long-term dependencies in sequences, also show promising results, although not always surpassing the TCN model. This comparative analysis demonstrates the significance of choosing an appropriate modeling approach based on the complexity and nature of the prediction task at hand.

Additionally, an important observation from the experiments is that the results obtained are similar across four different look-back window sizes for both future step intervals (1 h and 5 h). The analysis indicates that larger look-back window sizes do not significantly improve prediction accuracy. However, this finding emphasizes the effectiveness of smaller window sizes, which maintain sufficient accuracy while potentially reducing computational requirements. This efficiency is particularly beneficial in operational scenarios where processing speed and resource allocation are critical factors.

5.3. Prediction results

The best value of look-back window sizes for 1-hour and 5-hour prediction intervals were identified. In this section, The selected window sizes are tested based on two scenarios. In the first scenario, a maritime agent requests a prediction for a cargo ship's location over the next 1 h. This corresponds to a future step window size of 20, given that data is reported every 3 min. In the second scenario, the maritime agent seeks to predict a cargo ship's trajectory over the next 5 h, translating to a future step window size of 100 (calculated as 5 h or 300 min divided by 3 min). The cargo ship chosen for the evaluation phase is FINNSEA with IMO 9468891, sailing under the Finnish flag. Its AIS data collected for 1 month contains 10890 rows after the pre-processing stage.

Figs. 11 and 12 visualize the results for two optimal look-back window sizes: a two-day look-back window for a 1-hour prediction, and a four-day look-back window for a 5-hour prediction. In the following, a comprehensive analysis of both figures is presented:

1. Fig. 11(a) first compares the actual and predicted latitude and longitude values over the 1-hour prediction interval. The close alignment of the red (predicted) and blue (actual) lines indicates a high level of accuracy in the model's predictions, demonstrating the model's ability to capture the vessel's movement patterns effectively. Secondly, Fig. 11(b) illustrates the distribution of prediction errors for both latitude and longitude. The concentration of errors around zero suggests that the model's predictions are consistently close to the actual values. This distribution confirms the model's precision and reliability in predicting the vessel's future positions. Additionally, the models trained for 1-hour and 5-hour predictions, employing the best look-back sizes, were tested on the next 1000 steps or 50-hour prediction intervals. The sustained overlap in Fig. 11(c) between the predicted and actual values over this extended timeframe underscores the model's robustness and its ability to maintain accuracy over longer prediction horizons. Finally, Fig. 11(d) visualizes the actual (blue) and predicted (red) movements of the cargo ship along its route on a map. The close correspondence between the two trajectories further validates the model's spatial accuracy, showing that the predictions closely follow the actual navigational path of the vessel.
2. Fig. 12 presents the results for the 5-hour prediction using a four-day look-back window, revealing the model's capability to handle longer prediction intervals. The line charts in panel (a) indicate a strong correlation between the predicted and actual latitude and longitude values over the 5 h, signifying the model's proficiency in maintaining accuracy over extended prediction intervals. The histograms in panel (b) show that prediction errors are narrowly distributed around zero, confirming the model's reliability. The robustness of the model is further demonstrated in panel (c), which extends the prediction horizon to 1000 steps (50 h), where the predicted values remain closely aligned with the actual trajectory. Panel (d) offers a spatial validation, showing the predicted and actual paths on a map, highlighting the model's capability to accurately follow the ship's route. These findings illustrate that the TCN-based model, when using a four-day look-back window, provides dependable long-term predictions, making it suitable for real-time maritime applications requiring accurate and timely navigation insights.

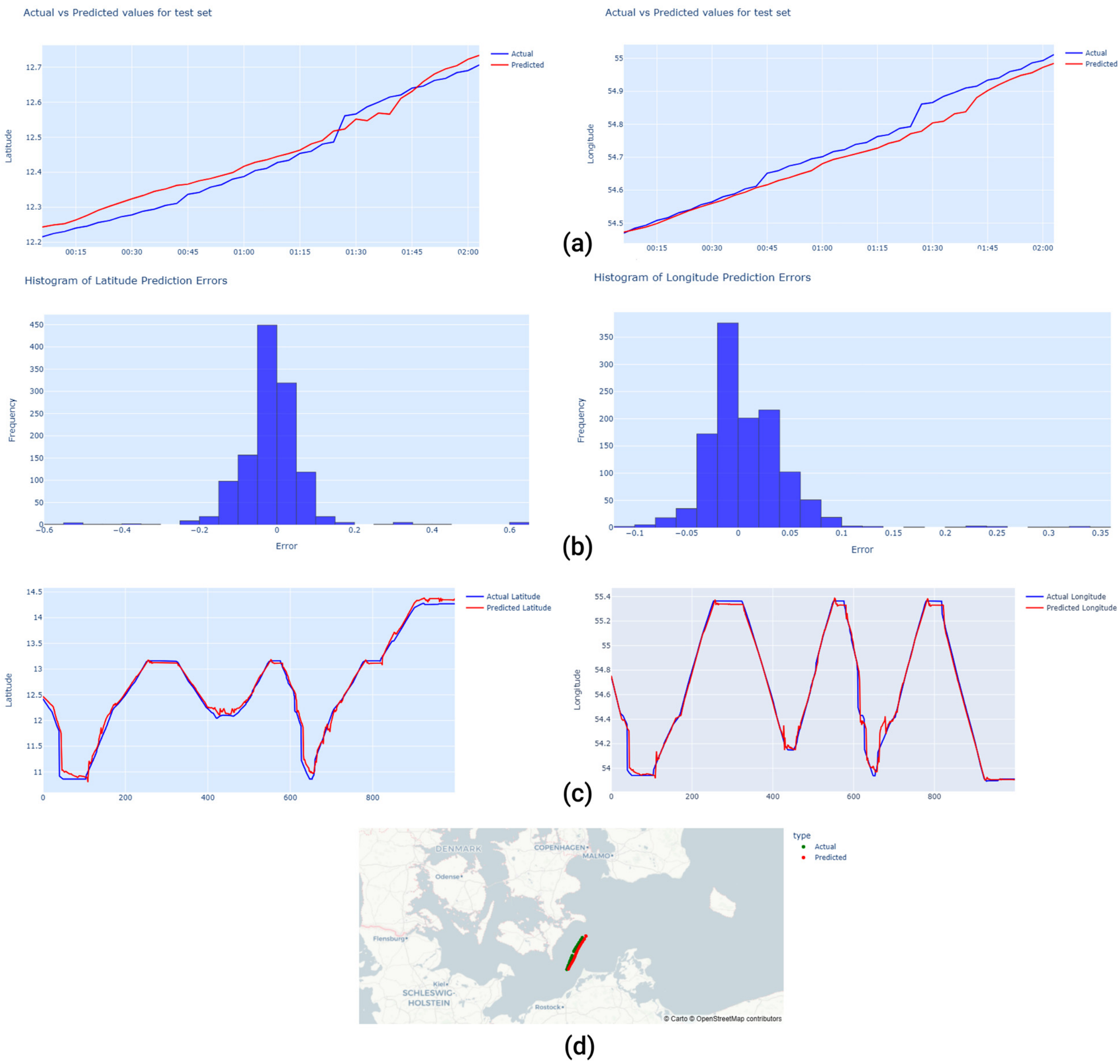

Fig. 11. Experimental results for 1-hour prediction with a two-day look-back for a cargo ship. (a) The line chart illustrates the actual and predicted latitude and longitude values; (b) The histogram chart shows the distribution of the model's error for both variables: latitude and longitude through histogram graphs; (c) The line graph presents the actual and predicted latitude and longitude values for 1000 steps; (d) The map visualizes the actual and predicted movements on a map.

5.4. Models training time

The training times for the TCN, LSTM, and LR models were evaluated for look-back window sizes of 1-day, 2-day, 3-day, and 4-day for both 1-hour and 5-hour prediction intervals, as shown in Fig. 13. The results indicate that the LR model consistently exhibited the shortest training times across all scenarios, ranging from 59 to 245 s for both intervals, due to its simplicity and linear nature. In contrast, the LSTM model, which captures long-term dependencies, had significantly longer training times, ranging from 710 to 2278 s for 1-hour predictions and 2506 to 6057 s for 5-hour predictions, highlighting its computational intensity. The TCN model demonstrated a balanced performance with training times that were consistently lower than LSTM but higher than LR, ranging from 579 to 1972 s for 1-hour predictions and 2183 to 4800 s for 5-hour predictions. This suggests that TCN can efficiently handle sequential data with less computational overhead than LSTM. Notably, TCN's training time was approximately 18.5% to 27.4% shorter than LSTM for the 1-hour interval and about 12% to 20% shorter for the 5-hour interval, indicating TCN's superior computational efficiency

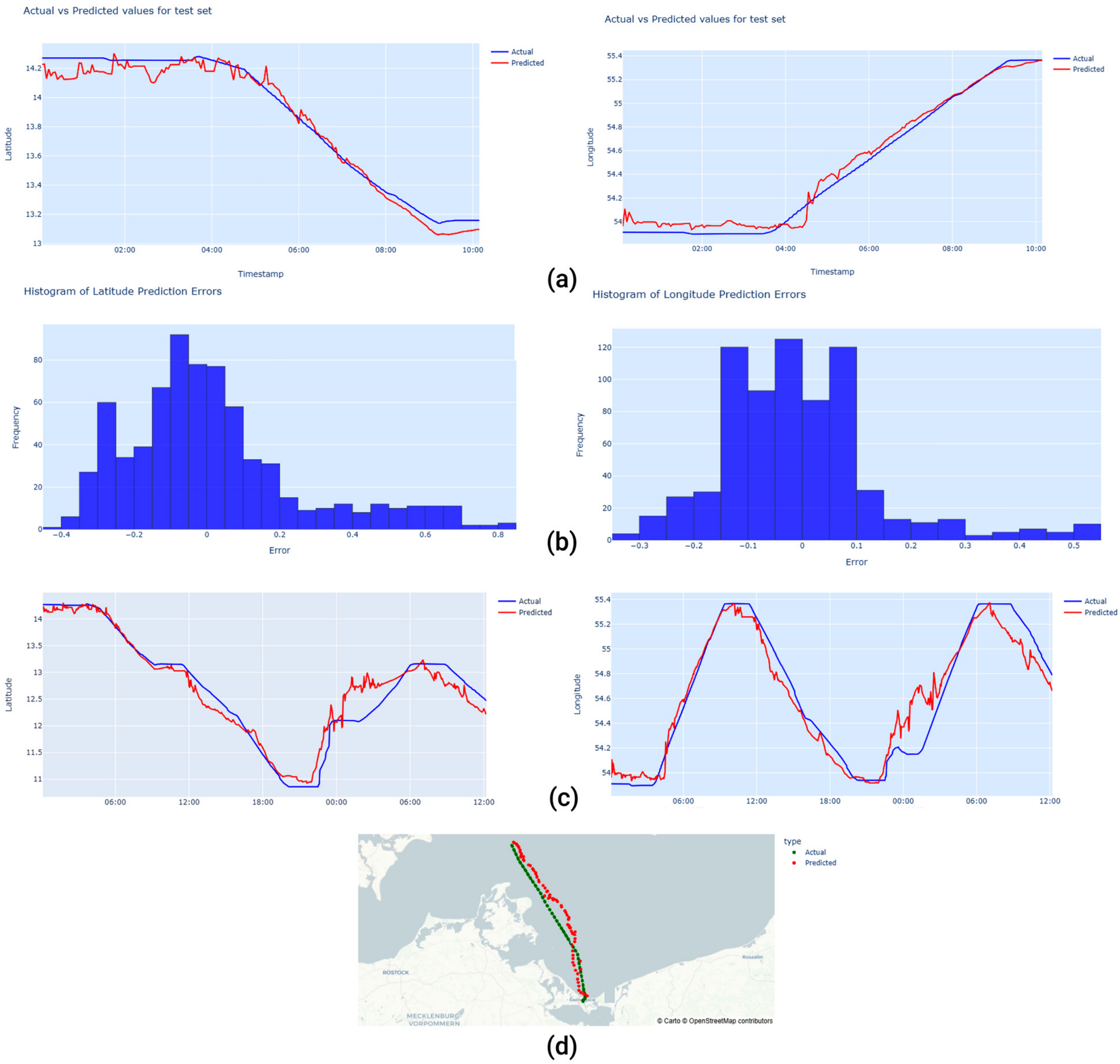

Fig. 12. Experimental results for 5-hour prediction with a four-day look-back for a cargo ship. (a) The line chart illustrates the actual and predicted latitude and longitude values; (b) The histogram chart shows the distribution of the model's error for both variables: latitude and longitude through histogram graphs; (c) The line graph of the actual and predicted latitude and longitude values for 1000 steps; (d) The map visualizes the actual and predicted movements on a map.

and scalability. Therefore, TCN provides a robust balance between prediction accuracy and training time, making it a preferred choice for ship trajectory prediction tasks since it requires fewer computational resources and is faster for real-world scenarios.

5.5. Handling missing ship data

During this study, it was identified a scenario where the target ship's data might be absent from the 1-month AIS database. To address this challenge and maintain the reliability of the framework, the following protocols have been established :

1. **Notification:** In the event of missing data for a requested ship, the maritime agent is immediately notified. This prompt notification enables quick response and decision-making.
2. **Data Retrieval Attempt:** Subsequently, the system attempts to retrieve the latest data for the ship. This involves querying alternative databases or external data sources, aiming to gather the most recent ship information.

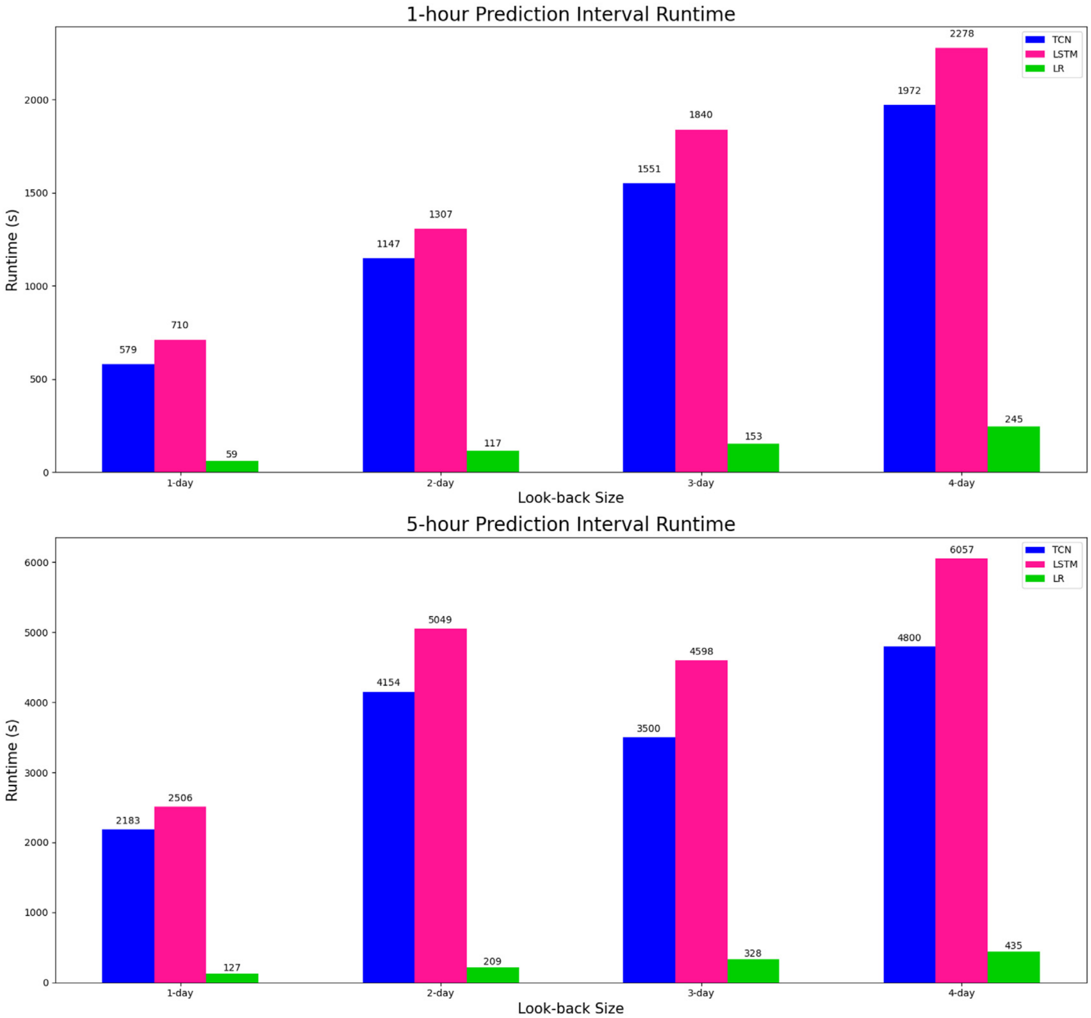

Fig. 13. Comparison of model training time for 1-hour and 5-hour prediction intervals.

3. **Error Logging and Flagging:** If the system is unable to locate any data following the retrieval attempt, an error is logged, and the request is marked for manual review. This step ensures systematic tracking and management of instances of data unavailability.
4. **Manual Intervention:** Thereafter, a maritime analyst may intervene to investigate the cause behind the absence of data. This investigation may include checking the ship's AIS transponder functionality or examining recent ship reports and logs.

This protocol enhances the resilience of the ship movement prediction framework, ensuring its effectiveness even in scenarios of incomplete data. By implementing these measures, the framework underscores its commitment to providing robust and adaptable solutions for maritime surveillance and analysis.

6. Conclusion

The study presents a framework for determining the optimal look-back window sizes needed to predict maritime ship movement, based on prediction intervals provided by maritime agents. The predictive model is designed to be user-centric, allowing maritime agents and analysts to customize the model to fit their specific operational requirements. To develop the framework, a four-week AIS dataset from the Baltic Sea is initially used to train TCN-based models using various look-back window sizes for two prediction intervals. The average prediction error for each look-back window size is then calculated, and the window sizes with the lowest average prediction error are identified. Based on the experimental results, the optimal look-back window sizes for 1-hour and 5-hour prediction intervals are found to be 2 days and 4 days, respectively. The framework's validity is further established by applying it to additional AIS data from a cargo ship in two scenarios: predicting 5 h and 1 h into the future to ensure that the window sizes are applicable to new ships. Furthermore, the TCN-based algorithm has proven to be more accurate and faster than the LSTM-based model. This user-centric approach not only enhances the practical usability of the framework but also ensures that the predictive analysis aligns with the individual needs of the users.

However, this study presents certain limitations that must be considered. Firstly, the evaluation focused on a specific scenario involving a Finnish cargo ship in the Baltic Sea. Therefore, extending the findings to other geographic areas or ship types requires further validation. This regional focus may limit the generalizability of the results, and future studies should aim to test the framework in a variety of maritime contexts. Moreover, the computational demands of training DL models could be a constraint, particularly for real-time predictions. The large dataset size and model complexity require significant computational resources, which might not always be feasible in real-time applications.

Another notable limitation is related to the input vector used in the model, which includes the Draught of the ship. It is important to acknowledge that in some AIS datasets, Draught information is not broadcasted. This omission limits the applicability of the proposed method in datasets where such information is absent. Future iterations of this research could explore the implications of excluding the Draught feature or investigate methods for handling incomplete datasets, such as data imputation techniques, to enhance the model's robustness and broaden its applicability.

Future work intends to incorporate weather data, as it may significantly impact maritime ship predictions. Additionally, exploring methods to reduce the number of data points processed by the prediction model aims to enhance the system's prediction speed. Continuous improvement and adaptation to various influencing factors will further bolster the robustness and applicability of the predictive framework in diverse maritime contexts.

Declaration of competing interest

We declare that we have no known competing financial interests or personal relationships which have or could be perceived to have, influenced the work reported in this manuscript.

Each author certifies that he or she has no commercial associations that might pose a conflict of interest in connection with the submitted manuscript, except as disclosed in a separate attachment. All funding sources supporting the work and all institutional or corporate affiliations of the authors are acknowledged in a footnote.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that all have approved the order of authors listed in the manuscript.

CRediT authorship contribution statement

Farshad Farahnakian: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Investigation, Data curation, Conceptualization. **Paavo Nevalainen:** Writing – review & editing, Writing – original draft, Validation, Methodology. **Fahimeh Farahnakian:** Writing – review & editing, Writing – original draft, Software, Conceptualization. **Tanja Vähämäki:** Writing – review & editing, Writing – original draft, Visualization, Validation. **Jukka Heikkonen:** Writing – review & editing, Writing – original draft, Validation, Supervision.

Acknowledgment

This work is part of the AI-ARC project funded by the European Union's Horizon 2020 research and innovation programme under grant 96 agreement No. 101021271.

References

Agrawal, R. K., Adhikari, R., 2013. An introductory study on time series modeling and forecasting. Nova York: CoRR.200–212.
Alam, M.M., Spadon, G., Etemad, M., Torgo, L., Milios, E., 2024. Enhancing short-term vessel trajectory prediction with clustering for heterogeneous and multi-modal movement patterns. Ocean Eng. 308, 118303.
Alizadeh, D., Alesheikh, A.A., Sharif, M., 2021. Vessel trajectory prediction using historical automatic identification system data. J. Navig. 74 (1), 156–174.
Bai, S., Kolter, J. Z., Koltun, V., 2018a. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. ArXiv preprint arXiv:1803.01271.
Bai, S., Kolter, J. Z., Koltun, V., 2018b. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. ArXiv preprint arXiv:1803.01271.
Bergmeir, C., Benitez, J.M., 2012. On the use of cross-validation for time series predictor evaluation. Inf. Sci. 191, 192–213.
Bisong, E., 2019. Building Machine Learning and Deep Learning Models on The Google Cloud Platform. CA: Apress, Berkeley.
Bourdeau, M., Qiang Zhai, X., Nefzaoui, E., Guo, X., Chatellier, P., 2019. Modeling and forecasting building energy consumption: a review of data-driven techniques. Sustain. Cities Soc. 48, 101533.
Capobianco, S., Millefiori, L.M., Forti, N., Braca, P., Willett, P., 2021. Deep learning methods for vessel trajectory prediction based on recurrent neural networks. IEEE Trans. Aerosp. Electron. Syst. 57 (6), 4329–4346.
Cochrane, J. H., 1997. Time Series for Macroeconomics and Finance. University of Chicago, Chicago.
Deng, D., 2020. DBSCAN clustering algorithm based on density. In: 2020 7th International Forum on Electrical Engineering and Automation (IFEEA). IEEE, p. 949-953.
Diana, G., Tommasi, C., 2002. Cross-validation methods in principal component analysis: a comparison. Stat. Methods Appl. 11, 71–82.
Farahnakian, F., Farahnakian, F., Sheikh, J., Nevalainen, P., Heikkonen, J., 2023. Short and long term vessel movement prediction for maritime traffic. In: International Conference on Critical Information Infrastructures Security. Cham: Springer Nature, Switzerland, pp. 62–80.
Farahnakian, F., Heikkonen, J., Nevalainen, P., 2022. Abnormal behaviour detection by using machine learning-based approaches in the marine environment: a literature survey. IEEE, pp. 1–11.
Gang, L., Wang, Y., Sun, Y., Zhou, L., Zhang, M., 2016. Estimation of vessel collision risk index based on support vector machine. Adv. Mech. Eng. 8 (11). 1687814016671250
Gao, M., Shi, G., Li, S., 2018. Online prediction of ship behavior with automatic identification system sensor data using bidirectional long short-term memory recurrent neural network. Sensors 18 (12), 4211.

Handbook of Statistics 2023. United Nations Conference on Trade and Development, 14 Dec.2023, [Online] Available: unctad.org/system/files/official-document/tdstat48en.pdf.
Han, Z., Zhao, J., Leung, H., Ma, K.F., Wang, W., 2019. A review of deep learning models for time series prediction. IEEE Sens. J. 21 (6), 7833–7848.
Hu, S., Xiong, C., 2023. High-dimensional population inflow time series forecasting via an interpretable hierarchical transformer. Transp. Res. Part C Emerg. Technol. 146, 103962.
International Convention for the Safety of Life at Sea (SOLAS), Chapter V: Safety of Navigation, Regulation 19, 12 December. 2002.
Joseph, A., Dalaklis, D., 2021. The international convention for the safety of life at sea: highlighting interrelations of measures towards effective risk mitigation. J. Int. Marit. Saf. Environ. Aff. Ship. 5 (1). 1–1
Keelawat, P., Thammasan, N., Numao, M., Kijsirikul, B., 2021. A comparative study of window size and channel arrangement on EEG-emotion recognition using deep CNN. Sens. 21 (5), 1678.
Kim, J., Jung, C., Kang, D., Lee, C.J., 2020. A new vessel path prediction method using long short-term memory. Trans. Korean Inst. Electr. Eng. 69 (7), 1131–1134.
Kim, K.I., Lee, K.M., 2018. Deep learning-based caution area traffic prediction with automatic identification system sensor data. Sensors 18 (9), 3172.
Kumar, N., Susan, S., 2020. COVID-19 pandemic prediction using time series forecasting models. In: 2020 11th International Conference on Computing, Communication, and Networking Technologies (ICCCNT). IEEE, pp. 1–7.
Lara-Benitez, P., Carranza-Garcia, M., Riquelme, J.C., 2021. An experimental review on deep learning architectures for time series forecasting. Int. J. Neural Syst. 31 (03), 2130001.
Liang, M., Liu, R.W., Zhan, Y., Li, H., Zhu, F., Wang, F.Y., 2022. Fine-grained vessel traffic flow prediction with a spatio-temporal multigraph convolutional network. IEEE Trans. Intell. Transp. Syst. 23 (12), 23694–23707.
Liu, T., Xu, X., Lei, Z., Zhang, X., Sha, M., Wang, F., 2023. A multi-task deep learning model integrating ship trajectory and collision risk prediction. Ocean Eng. 287, 115870.
Luo, X., Gan, W., Wang, L., Chen, Y., Ma, E., 2021. A deep learning prediction model for structural deformation based on temporal convolutional networks. Comput. Intell. Neurosci. 2021 (1), 8829639.
Nguyen, D., Vadaine, R., Hajduch, G., Garello, R., Fablet, R., 2018. A multi-task deep learning architecture for maritime surveillance using AIS data streams. In: 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA). IEEE, pp. 331–340.
Ofoeda, J., Boateng, R., Effah, J., 2019. Application programming interface (API) research: a review of the past to inform the future. Int. J. Enterprise Inf. Syst. (IJEIS) 15 (3), 76–95.
Ren, Y., Zhao, J., Liu, W., Wang, S., Wei, Y., 2019. Ship navigation behavior prediction based on AIS data and LSTM network. J. Shanghai Marit. Univ. 40, 32–37.
Ribeiro, C.V., Paes, A., de Oliveira, D., 2023. AIS-based maritime anomaly traffic detection: a review. Expert Syst. Appl. 231, 120561.
Rong, H., Teixeira, A.P., Soares, C.G., 2022. Maritime traffic probabilistic prediction based on ship motion pattern extraction. Reliab. Eng. Syst. Saf. 217, 108061.
Sezer, O.B., Gudelek, M.U., Ozbayoglu, A.M., 2020. Financial time series forecasting with deep learning: a systematic literature review: 2005-2019. Appl. Soft Comput. 90, 106181.
Spiliopoulos, G., Zissis, D., Chatzikokolakis, K., 2018. A big data-driven approach to extracting global trade patterns. In: Mobility Analytics for Spatio-Temporal and Social Data: First International Workshop, MATES. Springer International Publishing, pp. 109–121. Munich, Germany, September 1, 2017, Revised Selected Papers 1
Srivastava, N., . Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958.
Tian, X., Suo, Y., 2023. Research on ship trajectory prediction method based on difference long short-term memory. J. Mar. Sci. Eng. 11 (9), 1731.
United Nations Conference on Trade and Development. Review of Maritime Transport2022, Chapter 1. [Online] Available: https://unctad.org/system/files/official-document/rmt2022-en.
Virjonen, P., Nevalainen, P., Pahikkala, T., Heikkonen, J., 2018. Ship movement prediction using k-NN method. In: 2018 Baltic Geodetic Congress (BGC Geomatics). IEEE, pp. 304–309.
Wei, Y., Chen, Z., Zhao, C., Chen, X., 2023. Deterministic ship roll forecasting model based on multi-objective data fusion and multi-layer error correction. Appl. Soft Comput. 132, 109915.
Yang, C.H., Lin, G.C., Wu, C.H., Liu, Y.H., Wang, Y.C., Chen, K.C., 2022. Deep learning for vessel trajectory prediction using clustered AIS data. Mathematics 10 (16), 2936.
Yu, J.Y., Sghaier, M.O., Grabowiecka, Z., 2020. Deep learning approaches for AIS data association in the context of maritime domain awareness. In: 2020 IEEE 23rd International Conference on Information Fusion (FUSION). IEEE, pp. 1–8.
Zaman, B., Marijan, D., Kholodna, T., 2023. Interpolation-based inference of vessel trajectory waypoints from sparse AIS data in maritime. J. Mar. Sci. Eng. 11 (3), 615.
Zeng, F., Ou, H., Wu, Q., 2022. Short-term drift prediction of multi-functional buoys in inland rivers based on deep learning. Sensors 22 (14), 5120.
Zhang, C., Bin, J., Wang, W., Peng, X., Wang, R., Halldearn, R., Liu, Z., 2020. AIS data-driven general vessel destination prediction: a random forest-based approach. Transp. Res. Part C Emerg. Technol. 118, 102729.
Zhang, Z., 2018. Improved adam optimizer for deep neural networks. In: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). IEEE, pp. 1–2.
Zhao, J., Yan, Z., Zhou, Z., Chen, X., Wu, B., Wang, S., 2023. A ship trajectory prediction method based on GAT and LSTM. Ocean Eng. 289, 116159.
Zhao, W., Wang, D., Gao, K., Wu, J., Cheng, X., 2023. Large-scale long-term prediction of ship AIS tracks via linear networks with a look-back window decomposition scheme of time features. J. Mar. Sci. Eng. 11 (11), 2132.
Zhou, H., Chen, Y., Zhang, S., 2019. Ship trajectory prediction based on BP neural network. J. Artif. Intell. 1 (1), 29.