AIS-based Hybrid Vessel Trajectory Prediction for Enhanced Maritime Navigation

Ons Aouedi*, Flor Ortiz*, Thang X. Vu*, Alexandre Lefourn†, Felix Giese†,
Guillermo Gutierrez†, Symeon Chatzinotas*
* SnT, University of Luxembourg, 1855 Kirchberg, Luxembourg † OHB LuxSpace, 6832 Betzdorf, Luxembourg
Corresponding Author: ons.aouedi@uni.lu

***Abstract*—The growing integration of non-terrestrial networks (NTNs), particularly low Earth orbit (LEO) satellite constellations, has significantly extended the reach of maritime connectivity, supporting critical applications such as vessel monitoring, navigation safety, and maritime surveillance in remote and oceanic regions. Automatic Identification System (AIS) data, increasingly collected through a combination of satellite and terrestrial infrastructures, provide a rich source of spatiotemporal vessel information. However, accurate trajectory prediction in maritime domains remains challenging due to irregular sampling rates, dynamic environmental conditions, and heterogeneous vessel behaviors. This study proposes a velocity-based trajectory prediction framework that leverages AIS data collected from integrated satellite–terrestrial networks. Rather than directly predicting absolute positions (latitude and longitude), our model predicts vessel motion in the form of latitude and longitude velocities. This formulation simplifies the learning task, enhances temporal continuity, and improves scalability, making it well-suited for resource-constrained NTN environments. The predictive architecture is built upon a Long Short-Term Memory network enhanced with attention mechanisms and residual connections (`LSTM-RA`), enabling it to capture complex temporal dependencies and adapt to noise in real-world AIS data. Extensive experiments on two maritime datasets validate the robustness and accuracy of our framework, demonstrating clear improvements over state-of-the-art baselines.**

***Index Terms*—Non-terrestrial networks, time-series data, vessel trajectory prediction, Deep Learning, Automatic Identification System.**

I. Introduction

Non-terrestrial networks (NTNs), powered by Low-Earth Orbit (LEO) satellite constellations, have emerged as a cornerstone of global communication infrastructure [1]. These networks offer persistent and wide-area coverage, making them indispensable for mission-critical operations across remote and oceanic regions. In maritime domains, NTNs support a variety of applications such as vessel tracking, collision avoidance, and traffic management, enabling situational awareness and navigational safety in dynamic and high-risk environments [2]. A core component of maritime monitoring systems is the Automatic Identification System (AIS), a standardized communication protocol that enables vessels to broadcast their identity, position, speed, and navigational status at regular intervals. AIS transmits data between ships, coastal stations, and increasingly via satellite-based receivers, thereby supporting both terrestrial and non-terrestrial communication infrastructures. This real-time information exchange is critical for collision avoidance, traffic coordination, and networked maritime applications [3]. With the increasing availability of AIS data collected through both terrestrial and satellite infrastructures, researchers and practitioners have gained unprecedented access to large-scale, global maritime datasets. The integration of satellite-based and terrestrial-based AIS data with geospatial and temporal context opens new direction for developing predictive models. However, this integration also presents several fundamental challenges. AIS data often suffers from irregular sampling intervals, especially in low-traffic regions due to sparse infrastructure or in congested zones where signal collisions may occur. Moreover, intentional silence by vessels, such as disabling AIS transmissions for privacy, security, or illicit activities, further exacerbates data sparsity. In such cases, no update packets are transmitted or received, effectively interrupting communication and causing gaps in trajectory information, which poses significant challenges for real-time prediction and monitoring [4]. Moreover, noisy and inconsistent observations due to varying transmission intervals and environmental interference complicate the learning process. In such conditions, traditional trajectory prediction methods—deterministic or statistical—struggle to achieve robust performance [5] [6].

To achieve reliable trajectory predictions, we propose a robust and scalable deep learning (DL)-based vessel velocity prediction framework. Unlike conventional methods that directly predict absolute vessel positions [7] [8], our model reformulates the prediction task to focus on latitude and longitude velocities. This velocity-based formulation simplifies the learning objective, reduces cumulative errors during inference, and offers enhanced temporal consistency. Importantly, it also allows for more effective learning from noisy or irregular AIS samples. The architecture is built upon a Long Short-Term Memory (LSTM) network with integrated attention mechanisms and residual connections, called `LSTM-RA`, which captures long-range dependencies and adapts to complex vessel motion dynamics. Moreover, we emphasize that data preprocessing plays a critical role in trajectory prediction accuracy. Our experiments show that using segmentation-based trajectory grouping, which divides vessel movements into temporally coherent sub-sequences, yields significantly lower prediction errors than vessel ID (MMSI)-based grouping. The latter often results in fragmented, inconsistent, or non-uniform trajectories, which reduce model reliability. By contrast, segmentation facilitates more structured learning and better generalization across diverse regions.

This novel framework presents several benefits for NTN-based maritime applications. By relying on a minimal set of reliable features, namely timestamp, latitude, and longitude, it ensures both adaptability and computational efficiency. Importantly, we incorporate the relative time of the predicted position as an explicit input, enabling the model to predict vessel locations for any given future time with greater precision. This velocity-based prediction paradigm further enhances scalability for real-time, resource-constrained systems. Through comprehensive experiments on two large-scale, integrated AIS datasets, our model consistently outperforms state-of-the-art baselines, achieving superior accuracy and robustness while maintaining minimal feature requirements. As a result, this work contributes a scalable, adaptive, and efficient solution for maritime trajectory prediction, facilitating improved decision-making and safety within next-generation NTN-enabled ecosystems. The key contributions of this paper are:

- We propose the `LSTM-RA` model, which integrates *residual connections* and a *temporal attention mechanism* within a multi-layer LSTM architecture. This hybrid design enhances the propagation of long-term dependencies and effectively mitigates error accumulation in irregular and noisy AIS time series—addressing a well-known limitation in conventional LSTM predictors.
- We design a *segmentation-based preprocessing* method that organizes raw AIS streams into *temporally coherent sub-trajectories*, thereby preserving contextual vessel motion patterns and substantially improving prediction accuracy compared to traditional MMSI-based grouping.
- We introduce a *temporal conditioning* strategy by incorporating the *relative time offset* of the predicted position into the model's input, improving its *temporal awareness* and extrapolation accuracy for longer prediction horizons.
- We perform *comprehensive evaluations* on two real large-scale datasets—*Atlantic* and *Malaisie*—covering both *in-domain* and *zero-shot generalization* scenarios. Results show consistent performance gains over position-based, moving-average, standard baselines (LSTM, GRU, Transformer).

The paper is presented as follows: related works are briefly reviewed in section II, section III introduces the problem formulation, mathematical notations, and our proposition. More details about the experiment setup is given in section IV, empirical studies are reported in section V, section VI, section VII. Finally, section VIII concludes our work and discusses future works.

II. Related work

Vessel trajectory prediction is essential for enhancing navigational safety in both maritime and inland waterway transportation. Existing prediction solutions can be broadly categorized into three groups: (i) *physical model-based approaches*, (ii) *machine learning-based methods*, and (iii) *deep learning-based models*.

A. Physical Model-Based Vessel Trajectory Prediction

Physical model-based methods are grounded in physical laws and kinematic principles. Without leveraging historical trajectory patterns, they estimate a vessel's future position based solely on its current state, location, and velocity. Typical techniques in this category include Kalman filters, particle filters, and Markov models. For example, the authors in [9] introduced a Kalman filter-based approach that enhances trajectory prediction accuracy by effectively filtering vessel position data. Their method proves particularly effective in handling complex navigation scenarios, such as curved inland waterways. To further improve the accuracy of trajectory prediction, the study in [10] combined particle filtering with historical AIS data to predict trajectories effectively in narrow waterways.

B. Machine Learning-Based Vessel Trajectory Prediction

Machine learning-based methods predict future vessel trajectories by learning patterns from historical AIS data. Common techniques include Support Vector Machines (SVMs) and Random Forest (RF), each offering distinct strengths and limitations. For instance, the study in [11] incorporated vessel speed and heading into an SVM framework, improving predictive accuracy. Nevertheless, SVMs may struggle with generalization, particularly in out-of-distribution scenarios, where the test data exhibit patterns or dynamics not observed during training, such as new geographical regions, different vessel behaviors, or varying traffic densities. Moreover, an RF has been used in order to predict vessel destinations by comparing similarities between current and historical trajectories, demonstrating the algorithm's effectiveness in handling large and diverse datasets [12].

C. Deep Learning-Based Vessel Trajectory Prediction

DL has emerged as a powerful tool for vessel trajectory prediction, offering data-driven solutions capable of learning complex spatial and temporal dependencies from AIS data. Compared to traditional deterministic or statistical approaches, DL models provide superior adaptability to non-linear dynamics, noisy data, and irregular sampling [13]. Recurrent neural networks (RNNs), especially Long Short-Term Memory (LSTM) networks, have become foundational components in many vessel trajectory forecasting frameworks due to their ability to model long-range dependencies in time-series data. In this context, several advanced LSTM-based architectures have been proposed to further improve prediction accuracy. For example, [14] introduced the SFM-LSTM model, which integrates the Social Force Model with LSTM networks to predict spatiotemporal vessel trajectories with high robustness. A mixed-loss function was proposed to enhance generalization across different navigation contexts. Similarly, [15] presented a ConvLSTM-based Seq2Seq model that jointly captures spatial and temporal patterns, demonstrating superior performance over traditional Seq2Seq and Bi-Attention-LSTM architectures on real AIS data. Graph-based learning has also been explored in this domain. [16] developed a spatio-temporal multigraph convolutional network (STMGCN) leveraging three distinct graph representations based on social force, closest point of approach, and vessel size. Extending this idea, [8] proposed DAA-SGCN, which integrates motion encoding, spatio-temporal feature extraction, and trajectory prediction using MMSI, timestamp, position, speed over ground, and heading features. More recently, [17] introduced ST-MGT, a multi-graph transformer architecture that blends GCNs, LSTM units, and transformer layers to accurately capture vessel interactions and spatiotemporal behaviors. Combining attention mechanisms and RNNs, [18] designed a hybrid model using graph attention networks (GAT) for spatial encoding and LSTM for temporal sequence learning. Additionally, [4] conducted a comparative study of DL models for direct position-based trajectory prediction, examining their effectiveness in complex maritime environments. Generative modeling has also gained traction; for example, [19] proposed a conditional variational autoencoder (CVAE)-based architecture that captures vessel behavior by modeling multiple input features, including MMSI, speed over ground, heading, and positional data. More innovatively, [20] used a large language model (LLM)-based approach, where a token and motion embedding layer is used to process AIS sequences, followed by convolutional fusion and fine-tuning of a pre-trained LLM for trajectory prediction.

Despite significant progress in data-driven vessel trajectory prediction, most existing approaches rely on complex feature sets and directly predict absolute positions, making them highly sensitive to noise, irregular sampling, and data sparsity—issues that are intrinsic to AIS data, particularly in satellite-integrated systems. Moreover, several studies assume uniform data quality and underestimate the impact of preprocessing strategies such as trajectory segmentation. Conventional MMSI-based grouping, commonly adopted in prior work, often produces fragmented or inconsistent sequences that fail to capture the temporal continuity of vessel motion. To address these limitations, we propose a velocity-based prediction framework that uses only the essential spatio-temporal inputs—timestamp, latitude, and longitude—without relying on handcrafted or auxiliary features. Instead of predicting absolute coordinates, our model estimates the velocity components $(v_{\text{lat}}, v_{\text{lon}})$, which better capture vessel dynamics and yield superior robustness and computational efficiency. In addition, we demonstrate that segmentation-based preprocessing, which groups AIS data into temporally coherent sub-trajectories, substantially enhances predictive stability compared to MMSI-based grouping. Building on this foundation, the proposed AIS-enhanced maritime navigation system integrates satellite-collected AIS data with the velocity-learning model to enable scalable, real-time trajectory prediction in 6G/NTN-enabled maritime environments. Leveraging an LSTM architecture with residual and attention mechanisms, the system ensures temporal smoothness and resilience to irregular sampling while remaining lightweight enough for deployment on resource-constrained edge or satellite nodes. Moreover, the inclusion of relative prediction time as an auxiliary input enhances temporal awareness, enabling asynchronous multi-horizon prediction under heterogeneous maritime communication conditions.

III. Trajectory Prediction

In this section, we formally introduce the problem of vessel trajectory prediction and the formulation of the sequence-to-sequence learning approach used to address the prediction task. We present a data-driven approach to finding an approximate solution based on recurrent networks for sequence modeling in order to encode information from past data and generate future predictions. Fig. 1 illustrates the end-to-end workflow of our proposition.

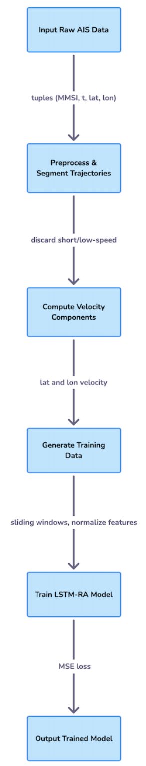


Fig. 1: Flowchart of the proposed velocity-based trajectory prediction framework. The process includes data preprocessing, velocity computation, training data generation, and model training using the LSTM-RA architecture.

A. Problem Definition

A dataset of N vessel trajectories can be represented as a collection of temporally ordered sequences of tuples:

$$\mathcal{C} = \{\mathcal{S}_i, \mathcal{T}_i\}_{i=1}^{N}.$$

Each trajectory is described by

- A sequence $\mathcal{S}_i$ of vessel states:
$$\mathcal{S}_i = \{\mathbf{s}_k \mid k = 1, \dots, T_i\},$$
- A corresponding sequence of time points $\mathcal{T}_i = \{t_1, \dots, t_{T_i}\}$, with $t_1 < t_2 < \dots < t_{T_i}$.

Here, T_i denotes the number of time steps in the trajectory i, and each state vector $\mathbf{s}_k = \mathcal{S}_i(t_k) \in \mathbb{R}^d$ encodes the vessel's dynamic condition at time t_k. The dimensionality d of the state vector may include various spatial and kinematic features, such as latitude, longitude, velocity components, speed, or heading. The specific choice of state representation depends on the prediction objective and available data.

The goal of trajectory prediction is to learn a spatio-temporal mapping from historical vessel movements in order to predict future states over a fixed prediction horizon. This is formulated as a supervised sequence-to-sequence learning task, where the model generates a future sequence of vessel states given a past sequence of observations. In this study, we focus on predicting future velocity components (in the latitude and longitude directions), from which future positions are reconstructed. To note, our framework deliberately uses a minimal feature set, such as timestamps, latitude, and longitude, to ensure scalability, generalizability, and applicability in real-time, resource-constrained environments. Although auxiliary information such as vessel speed, size, or type could theoretically impose bounds on feasible velocity values, such features are often missing, noisy, or inconsistently reported in AIS data. Therefore, we avoid incorporating them to maintain robustness and prevent introducing bias from unreliable metadata. A data segmentation process is applied to extract input-output sequence pairs, enabling efficient model training (see Section III-E).

B. Trajectory Prediction Task

The goal of trajectory prediction is to learn a mapping from historical vessel observations to specific future features, such as velocity components or positions, over a fixed prediction horizon H. Formally, given a fixed-length input sequence of ℓ past observations:

$$\mathbf{X}_k = \{\mathbf{s}_\tau \mid \tau = k - \ell + 1, \ldots, k\},$$

the task is to predict the corresponding output sequence of future target features:

$$\mathbf{Y}_k = \{\mathbf{y}_\tau \mid \tau = k + 1, \ldots, k + H\},$$

where each $\mathbf{y}_\tau \in \mathbb{R}^d$ represents the predicted features of interest at time τ, such as latitude-velocity and longitude-velocity, or alternatively, geographic coordinates.

To address the irregular sampling patterns typical in AIS data, a data segmentation procedure is applied to convert variable-length trajectories into fixed-length input-output pairs. This is accomplished using a sliding window approach, which extracts overlapping segments from each trajectory. The result is a training dataset of examples:

$$\mathcal{D} = \{(\mathbf{X}_k, \mathbf{Y}_k)\}_{k=1}^{n},$$

where n denotes the total number of sliding windows extracted. This preprocessing step enables the model to effectively learn from sequential patterns while accommodating challenges such as missing entries, inconsistent intervals, and trajectory fragmentation.

C. Position-Based Trajectory Prediction

Traditional trajectory prediction methods typically aim to directly forecast future geospatial coordinates:

$$\{\text{lat}_{T+h}, \text{lon}_{T+h}\}_{h=1}^{H},$$

where T denotes the final observed time step and H is the prediction horizon. These methods treat trajectory prediction as a sequence regression problem over absolute positions.

While intuitive and widely adopted, position-based prediction approaches are often limited by the quality and regularity of the input data. In real-world maritime settings, particularly in NTN environments, AIS data can suffer from irregular sampling intervals, missing entries, and noisy signals due to environmental disruptions or vessel behavior. These factors significantly impair the ability of position-based models to generalize, leading to poor trajectory accuracy and unstable predictions.

D. Velocity-Based Trajectory Prediction

To overcome the limitations of direct position prediction, we propose a velocity-based modeling framework. Instead of predicting geospatial coordinates, the model learns to predict the velocity components:

$$\{v_{\text{lat},h}, v_{\text{lon},h}\}_{h=1}^{H},$$

where $v_{\text{lat},h}$ and $v_{\text{lon},h}$ denote the predicted velocities in the latitude and longitude directions, respectively. These velocity predictions are then integrated over time to reconstruct future positions:

$$\text{Lat}_{\text{pred},h} = \text{Lat}_{\text{last}} + v_{\text{lat},h} \cdot \Delta t, \quad \text{Lon}_{\text{pred},h} = \text{Lon}_{\text{last}} + v_{\text{lon},h} \cdot \Delta t,$$

where Δt represents the time interval between predictions, and $(\text{Lat}_{\text{last}}, \text{Lon}_{\text{last}})$ are the most recent observed coordinates.

Velocity computation: For each vessel trajectory, AIS messages are first sorted chronologically to ensure temporal consistency. Instantaneous velocity components are then derived between consecutive observations as:

$$v_{\text{lat},i} = \frac{\text{lat}_{i+1} - \text{lat}_i}{t_{i+1} - t_i}, \qquad v_{\text{lon},i} = \frac{\text{lon}_{i+1} - \text{lon}_i}{t_{i+1} - t_i},$$

where $(\text{lat}_i, \text{lon}_i)$ are consecutive positions and t_i is the timestamp in hours. Latitude and longitude are converted into kilometers.

This preprocessing ensures consistent spatial scaling across vessels operating in different regions, allowing the model to generalize effectively across heterogeneous AIS sources.

Advantages of the velocity-centric design:

- **Resilience to irregular sampling:** By explicitly modeling Δt, the method adapts to uneven temporal intervals, improving robustness under AIS irregularities caused by satellite coverage gaps or intentional shutoffs.
- **Smoother learning signals:** Velocity trajectories exhibit smoother dynamics than absolute position sequences, providing the model with more stable and predictable temporal patterns.
- **Dynamic interpretability:** Predicting velocity components allows the system to capture vessel dynamics

(speed, direction, or maneuver changes), facilitating real-time interpretability for navigation and anomaly detection.

By shifting the modeling focus from positions to velocities, the proposed approach achieves higher robustness, interpretability, and generalization in complex maritime environments.

E. Data Segmentation

AIS data is often characterized by irregular temporal sampling, data noise, and missing values, which pose significant challenges for training predictive models. Moreover, raw AIS trajectories vary in length and consistency due to the asynchronous nature of transmissions. To convert this heterogeneous data into a structured format suitable for supervised sequence modeling, we implement a rigorous segmentation strategy that integrates both temporal and spatial constraints, followed by a sliding window approach.

The historical trajectory dataset $\mathcal{C} = \{\mathcal{S}_i, \mathcal{T}_i\}_{i=1}^{N}$ contains sequences of observations $\mathcal{S}_i = \{\mathbf{s}_k\}_{k=1}^{T_i}$ and corresponding timestamps $\mathcal{T}_i = \{t_1, \ldots, t_{T_i}\}$, where $\mathbf{s}_k$ denotes the vessel's state at time t_k. Due to inconsistencies in sampling intervals, we first preprocess each sequence as follows:

Preprocessing and Thresholding: For each vessel, observations are sorted chronologically. Time differences $\Delta t_i = t_{i+1} - t_i$ and spatial distances $\Delta d_i = \text{geodesic}(\mathbf{s}_{i+1}, \mathbf{s}_i)$ are computed between consecutive points. A new segment is initiated if either $\Delta t_i > \delta_t$ or $\Delta d_i > \delta_d$, where δ_t and δ_d are empirically determined thresholds. After several visualizations and exploratory analyses of vessel trajectories, we selected threshold values that best separated continuous navigation patterns from discontinuous or noisy segments. In particular, the chosen thresholds reflect the observed distribution of time gaps and displacement distances between consecutive AIS messages.

Additionally, points with near-zero velocity (e.g., speed < 2 km/h) are excluded, as they do not contribute to meaningful movement dynamics. Segments shorter than a minimum length θ are discarded to ensure the quality of extracted patterns.

Sliding Window Generation: Once clean segments are formed, we apply a sliding window to generate input-output training pairs. Specifically, for each segment, we extract overlapping windows of historical states:

$$\mathbf{X}_k = \{\mathbf{s}_\tau \mid \tau = k - \ell + 1, \ldots, k\},$$

along with their corresponding prediction targets:

$$\mathbf{Y}_k = \{\mathbf{s}_\tau \mid \tau = k + 1, \ldots, k + H\},$$

where ℓ is the input sequence length and H is the prediction horizon.

The final training dataset becomes a structured collection of samples:

$$\mathcal{D} = \{(\mathbf{X}_k, \mathbf{Y}_k)\}_{k=1}^{n},$$

where n denotes the number of generated training windows. This segmentation framework ensures temporal coherence and spatial consistency, thereby facilitating accurate learning of trajectory patterns from real-world AIS data.

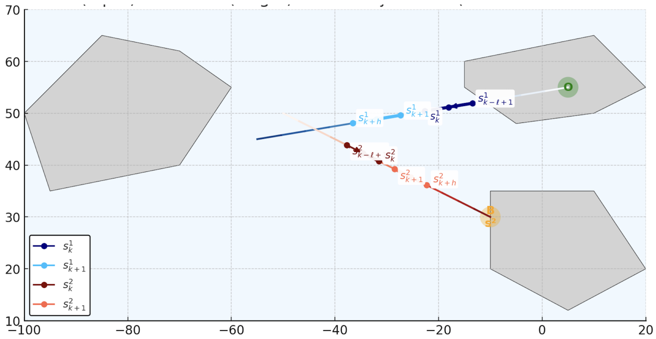


Fig. 2: Sliding window-based segmentation for two example trajectories S^1 and S^2, showing the transformation of raw AIS data into fixed-length input-output pairs for training.

F. Sequence-to-Sequence Modeling Approach

We adopt a sequence-to-sequence DL approach to address the trajectory prediction problem introduced in Section III-A. A sequence-to-sequence model maps a fixed-length input sequence to a fixed-length output sequence, where the input and output lengths may differ. From a probabilistic perspective, the goal is to learn the following predictive distribution:

$$p(\mathbf{Y}_k \mid \mathbf{X}_k).$$

Once this predictive distribution is learned, the sequence-to-sequence model directly generates a target sequence $\hat{\mathbf{Y}}_k$ given the input sequence $\mathbf{X}_k$. The supervised learning task becomes a sequence regression problem, where the objective is to train a neural network $F_{\ell,H}$ to predict the target sequence:

$$\hat{\mathbf{Y}}_k = F_{\ell,H}(\mathbf{X}_k) = \arg\max_{\mathbf{Y}} p(\mathbf{Y}_k \mid \mathbf{X}_k).$$

The model $F_{\ell,H}(\mathbf{X}_k; \theta)$ is parameterized by θ, the set of trainable parameters. To train the model, we optimize a task-specific loss function $\mathcal{L}$ to find the parameters $\hat{\theta}$ that minimize the prediction error over the dataset $\mathcal{D}$:

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{n} \sum_{k=1}^{n} \mathcal{L}\big(F_{\ell,H}(\mathbf{X}_k; \theta), \mathbf{Y}_k\big),$$

where n is the total number of training samples. A common choice for the loss function $\mathcal{L}$ in regression tasks is the mean squared error (MSE), which penalizes the squared differences between predictions and ground truth. This sequence-to-sequence framework is designed to capture the underlying spatio-temporal dependencies in historical trajectories, enabling robust and accurate prediction of future vessel states. By focusing on velocity components, the model can better handle noisy and irregularly sampled data, offering improved resilience and interpretability in maritime trajectory forecasting.

G. LSTM with Attention and Residual Connections

To effectively model complex spatiotemporal dependencies in maritime trajectory data, we propose an architecture based on Long Short-Term Memory (LSTM) units, enhanced with attention mechanisms and residual connections. This design enables robust modeling of sequential patterns, particularly when predicting velocity-based vessel states (e.g., latitude- and longitude-velocity components).

a) LSTM Component: The core of the model is a multi-layer LSTM network, which captures temporal dependencies across input sequences [21]. The gated architecture of the LSTM enables selective information retention, mitigating issues such as vanishing gradients that are prevalent in traditional RNNs. This makes it well-suited for long-range dependency modeling in maritime data streams.

b) Attention Mechanism: To improve the model's ability to focus on relevant portions of the input sequence, we incorporate an attention mechanism. Rather than relying on a fixed-length context vector, the attention module dynamically weights input states based on their relevance to the prediction [22]. Given an input sequence $\mathbf{X}_k = \{\mathbf{s}_\tau \mid \tau = k-\ell+1, \ldots, k\}$, the attention mechanism computes a context vector $\mathbf{c}_t$ at output time step t as:

$$\mathbf{c}_t = \sum_{\tau=k-\ell+1}^{k} \alpha_{t,\tau}\, \mathbf{s}_\tau,$$

where the attention weights $\alpha_{t,\tau}$ are obtained through a softmax operation:

$$\alpha_{t,\tau} = \frac{\exp(e_{t,\tau})}{\sum_{\tau'=k-\ell+1}^{k} \exp(e_{t,\tau'})}, \quad e_{t,\tau} = \text{score}(\mathbf{h}_t, \mathbf{s}_\tau).$$

Here, $\mathbf{h}_t$ is the LSTM hidden state at time t, and $\text{score}(\cdot)$ is a learnable function that quantifies the alignment between $\mathbf{h}_t$ and each input $\mathbf{s}_\tau$. This mechanism allows the model to prioritize temporally important signals, improving its ability to track vessel motion under irregular and noisy observation patterns.

c) Residual Connections: To further stabilize training and facilitate information flow across layers, we introduce residual connections between the input embeddings and the corresponding LSTM outputs [23]. This design supports the learning of both shallow and deep temporal features by allowing direct gradient propagation during optimization. Residual connections also enhance model convergence and robustness, particularly in long sequences or high-dimensional data settings.

Together, these architectural components enable the proposed `LSTM-RA` model to effectively model nonlinear, temporally dependent vessel behaviors. The combination of memory mechanisms, dynamic attention weighting, and residual learning ensures robustness to noise, irregular sampling, and complex maritime dynamics.

H. Velocity-Based Prediction and Reconstruction

Conventional position-based models directly estimate future geospatial coordinates. While intuitive, these methods often struggle with irregular sampling intervals and noise conditions common in maritime environments. To mitigate these challenges, we adopt a velocity-based prediction framework that focuses on modeling motion dynamics rather than absolute positions.

a) Velocity Prediction.: The proposed model predicts velocity components in the latitude and longitude directions, denoted as $\hat{v}_{\text{lat}}^{(h)}$ and $\hat{v}_{\text{lon}}^{(h)}$, for each future step $h \in \{1, \ldots, H\}$. The loss function used during training minimizes the discrepancy between the predicted and ground-truth velocities:

$$\left(v_{\text{lat}}^{(h)} - \hat{v}_{\text{lat}}^{(h)}\right)^2 + \left(v_{\text{lon}}^{(h)} - \hat{v}_{\text{lon}}^{(h)}\right)^2.$$

The input to the model consists of a fixed-length sequence of historical states, each represented by relative time scalars and spatial coordinates:

$$\mathcal{S}_{\text{past},k} = \{\mathbf{s}_\tau = (t_{\text{rel},\tau}, \text{lat}_\tau, \text{lon}_\tau) \mid \tau = k-\ell+1, \ldots, k\},$$

where $t_{\text{rel},\tau} = t_\tau - t_{k-\ell+1}$ denotes the relative time with respect to the start of the window. For each future time step, we append a predicted relative time scalar:

$$t_{\text{rel,pred},h} = t_{\text{pred},h} - t_k, \quad \text{where } t_{\text{pred},h} = t_k + h \cdot \Delta t_{\text{pred}}.$$

This design offers two primary advantages:

- *Trajectory alignment:* Using relative time scalars normalizes trajectories, reducing model sensitivity to absolute timestamps and improving generalization.
- *Temporal awareness:* The model is explicitly informed of the prediction horizon for each output step, enhancing its ability to model time-dependent vessel dynamics.

b) Trajectory Reconstruction.: After predicting velocity components, we reconstruct the future positions through iterative integration over the prediction horizon:

$$\text{Lat}_{\text{pred},h} = \text{Lat}_{\text{last}} + \hat{v}_{\text{lat}}^{(h)} \cdot \Delta t, \quad \text{Lon}_{\text{pred},h} = \text{Lon}_{\text{last}} + \hat{v}_{\text{lon}}^{(h)} \cdot \Delta t,$$

where $(\text{Lat}_{\text{last}}, \text{Lon}_{\text{last}})$ are the coordinates at the final observed timestep, and Δt is the time interval between predictions. This recursive multistep integration ensures temporal coherence and minimizes error propagation, especially in irregular and noisy sequences.

Overall, this velocity-centric formulation provides a lightweight yet expressive framework for trajectory prediction, requiring only timestamps and positional data as inputs. This simplifies deployment in real-world NTN-enabled maritime systems while maintaining strong predictive accuracy.

I. Loss Function for Velocity-Based Prediction

To train the sequence-to-sequence model described in Section III-F, we adopt the mean squared error (MSE) as the objective function. This loss function penalizes deviations between predicted and ground-truth velocity components over a prediction horizon of length H. The loss is formally defined as:

$$\mathcal{L} = \frac{1}{H} \sum_{h=1}^{H} \left[\left(v_{\text{lat}}^{(h)} - \hat{v}_{\text{lat}}^{(h)}\right)^2 + \left(v_{\text{lon}}^{(h)} - \hat{v}_{\text{lon}}^{(h)}\right)^2 \right],$$

where $v_{\text{lat}}^{(h)}$ and $v_{\text{lon}}^{(h)}$ denote the ground-truth velocity components in the latitude and longitude directions at time step h, and $\hat{v}_{\text{lat}}^{(h)}$, $\hat{v}_{\text{lon}}^{(h)}$ are the corresponding model predictions.

This formulation helps the model to accurately learn the motion dynamics of vessels over time. By focusing on velocity rather than absolute positional coordinates, the loss function inherently supports smoother transitions and improves robustness to irregular temporal sampling, an inherent characteristic of AIS data in real-world maritime environments. In practical settings, velocity components are computed from consecutive AIS observations by measuring geospatial displacement over time. These components are typically expressed in interpretable physical units (e.g., kilometers per hour), derived by applying appropriate scaling factors to account for Earth's curvature and latitude-dependent distance distortions. Such normalization ensures consistency between model outputs and real-world measurements, enhancing the interpretability and deployability of the learned model.

J. Algorithm Overview

To summarize the proposed trajectory prediction pipeline, we present Algorithm 1, which outlines the core stages, including data preprocessing, sequence segmentation, velocity computation, and model training. This process ensures a consistent preparation of AIS data for sequence-to-sequence learning using the proposed velocity-based framework.

Algorithm 1 Velocity-Based Vessel Trajectory Prediction

Input: Raw AIS dataset $\mathcal{C}$ with tuples (MMSI, t, lat, lon)
Output: Trained model $F_{\ell,H}(\cdot)$ for velocity prediction
Step 1: Preprocess and Segment Trajectories
for each vessel trajectory $\mathcal{S}_i$ in $\mathcal{C}$ **do**
    Sort $\mathcal{S}_i$ by timestamp
    Compute Δt and Δd between observations
    Start new segment if $\Delta t > \delta_t$ or $\Delta d > \delta_d$
    Discard short segments or with low speed
end for
Step 2: Compute Velocity
for each segment **do**
    Compute $v_{\text{lat}} = \Delta\text{lat}/\Delta t$, $v_{\text{lon}} = \Delta\text{lon}/\Delta t$
end for
Step 3: Generate Training Data
for each segment **do**
    Apply sliding window of length ℓ for inputs and H for outputs
    Normalize lat, lon, and time features using min-max scaling
    Generate and store $(\mathbf{X}_k, \mathbf{Y}_k)$
end for
Step 4: Train Model
Initialize model $F_{\ell,H}(\cdot)$ (LSTM with Attention and Residual Connections)
Train using MSE loss on predicted vs. ground-truth velocity components
return Trained model $F_{\ell,H}(\cdot)$

Computational Complexity. The computational complexity of the proposed `LSTM-RA` model, which integrates L stacked LSTM layers with an attention mechanism. For a given input sequence length ℓ, hidden dimension d, and batch size B, the forward-pass complexity per layer is $\mathcal{O}(B \times \ell \times d^2)$, corresponding to the recurrent matrix multiplications within each LSTM cell. The self-attention block contributes an additional $\mathcal{O}(B \times \ell^2 \times d)$ term, which remains moderate given the relatively short AIS temporal windows used in our experiments ($\ell = 12$, $H = 5$). Hence, the overall model complexity can be expressed as:

$$\mathcal{O}(L \times B \times \ell \times d^2 + B \times \ell^2 \times d).$$

The memory complexity scales linearly with ℓ and L, as each LSTM layer maintains hidden and cell states of size $\mathcal{O}(d)$. These characteristics show that the proposed approach remains computationally efficient and scalable for deployment in large-scale maritime monitoring systems.

IV. Experimental Setup

This section details the data preprocessing pipeline, model configuration, and training procedures employed to evaluate the proposed velocity-based trajectory prediction framework. Given the irregular and often noisy nature of AIS data, careful preprocessing is essential to ensure reliable learning signals. We first describe how raw AIS messages are transformed into structured input-output pairs suitable for sequence-to-sequence modeling. This includes MMSI-based grouping, segmentation based on spatio-temporal thresholds, velocity computation, and sliding window sampling. The goal of this preparation is to align raw vessel trajectories with the specific requirements of our deep learning framework while preserving critical motion dynamics.

A. Dataset Preparation

The datasets used in this study comprise two distinct maritime regions: the Atlantic and Malaysia areas. The Atlantic dataset includes over 50,000 trajectory segments, while the Malaysia dataset contains more than 5,000 segments, both derived from real AIS messages collected between 2024 and 2025. Each vessel trajectory is derived from AIS data, which contains timestamped position messages (latitude, longitude) uniquely identified by the vessel's MMSI number. The main steps are as follows:

1) **Grouping by MMSI:** AIS messages are first sorted chronologically and grouped by their MMSI numbers, yielding individual vessel-specific trajectories in time-ordered format.
2) **Trajectory Segmentation:** To manage irregular sampling and potential anomalies (e.g., large temporal or spatial gaps), each vessel trajectory is split into smaller, continuous segments whenever time or distance thresholds are exceeded. Segments with insufficient points are discarded to ensure reliability for velocity estimation and subsequent modeling. Then the AIS messages are sorted and grouped by their segment id instead of MMSI numbers.
3) **Velocity Computation:** Within each segment, velocity components in the latitude and longitude directions are computed from consecutive positional observations, capturing how the vessel's latitude and longitude change over time.

4) **Sliding Window:** A sliding window technique is then applied to each segmented trajectory to generate fixed-length input-output pairs. Each input sequence covers ℓ historical observations (including the associated velocities), while each target sequence spans H future steps. By systematically advancing this window across each segment, overlapping pairs are created for model training and evaluation.

This preprocessing step consolidates the raw AIS messages into well-defined segments and windows that align with the velocity-based prediction framework. By grouping, segmenting, computing velocity, and forming consistent input-output pairs, the data is made ready for training sequence-to-sequence models aimed at robust and accurate maritime trajectory prediction.

B. Scenario Description

In this study, we use two distinct AIS datasets covering maritime operations in (i) the Atlantic region and (ii) the Indonesia–Malaysia region, exclusively on cargo vessel types. Both areas exhibit complex traffic patterns, irregular reporting intervals, and noise in AIS transmissions, making them challenging yet representative environments for testing trajectory prediction models. The Atlantic dataset spans from January to December 2023, featuring variable time intervals between reported positions, with an average of approximately one hour. Due to the high volume of traffic and diverse vessel routes, data segmentation criteria (Section III-E) are applied to break trajectories into more homogeneous segments. This ensures consistent temporal and spatial properties across the resulting segments. After segmentation, velocity components are computed to capture vessel motion dynamics. Similar preprocessing steps are conducted on a second dataset collected from the Indonesia–Malaysia region, which also covers a range of vessel types and routes with varying sampling intervals. This dataset was collected in 2024. The same segmentation and velocity computation procedures are employed to maintain methodological consistency, allowing for comparative evaluation across two distinct maritime domains. For each segmented trajectory in both datasets, a sliding window approach (Section III-E) is applied to generate fixed-length input and output sequences. Specifically, each input sequence has length, ℓ and each target sequence has length H. The combination of segmentation, velocity computation, and windowing produces final datasets of fixed-size input-output pairs suitable for training and evaluating sequence-to-sequence prediction models. By leveraging two geographically and operationally diverse datasets, we demonstrate the robustness and generalizability of the proposed velocity-based trajectory prediction framework using an advanced LSTM-based model.

C. Models

We compare the performance of different DL-based models—LSTM, BiLSTM, GRU, and Transformer—that can be applied to trajectory prediction problems. We test different DL models, whose hyperparameters have been selected after an extensive hyperparameter optimization procedure aimed at finding the most suitable configurations.

D. Training Settings

The final goal of the learning process is to find the optimal approximation for the predictive function, i.e., the function that maps past trajectories into future ones, both composed of continuous values. We first normalize the data, and the training was performed for a maximum of 150 epochs with a learning rate of 0.001, 250 hidden units, 7 LSTM layers, a batch size of 256, and 0.2 dropout. These hyperparameters were selected through empirical tuning based on validation performance. We initially experimented with multiple configurations, including learning rates in $\{1 \times 10^{-3},\ 5 \times 10^{-4},\ 1 \times 10^{-4}\}$, hidden units in $\{64,\ 128,\ 250,\ 512\}$, and the number of LSTM layers ranging from 2 to 10. The final configuration yielded the best trade-off between convergence speed, predictive accuracy, and training stability across both the Atlantic and Malaysia datasets. We use the early-stopping criterion, evaluating the error measure (MSE between prediction and ground truth sequences) on the validation set and guiding how many iterations we can perform before the model begins to overfit.

E. Experimental Setup

In all experiments, the input sequence length (l) and prediction horizon (H) were empirically set to l = 12 and H = 5, corresponding respectively to 12 historical AIS observations and 5 future prediction steps. Model training and evaluation were performed on an Ubuntu 22.04 workstation equipped with two NVIDIA RTX A6000 GPUs (48 GB VRAM each), an 80-core (40 physical) CPU, and 503 GB of RAM, running Python 3.11.5, PyTorch 2.2 + CUDA 12.1. Energy consumption reported in Table II was monitored using the CodeCarbon library[1]. This configuration ensures reproducible runtime and energy measurements and allows fair comparison among models.

V. Performance Evaluation on the Atlantic Dataset

A. Velocity-Based Prediction vs. Direct Position Prediction

In this section, using our `LSTM-RA` model, we present a comparative analysis of two distinct prediction strategies evaluated on the Atlantic dataset: (i) velocity-based prediction and (ii) direct position (latitude and longitude) prediction. Figures 3 and 4 depict the error distributions for the two approaches. To note, the `LSTM-RA` model operates only over three input features—latitude, longitude, and timestamps—for both strategies.

The velocity-based prediction method, shown in Figure 3, exhibits a highly concentrated error distribution, with most errors clustered at lower magnitudes. This method achieves a mean positional error of 2.62 km, which attests to its high level of accuracy. By predicting incremental velocity components and integrating them over time, the approach effectively minimizes cumulative errors. In contrast, the direct position prediction approach, illustrated in Figure 4, yields a considerably broader error distribution, with an average error of 55.43 km. This method demonstrates lower predictive accuracy and a

[1] https://github.com/mlco2/codecarbon

much higher dispersion of errors. The disparity in performance can be attributed to the velocity-based method that benefits from a stepwise estimation process that mitigates cumulative errors, whereas direct position prediction—requiring the model to predict absolute spatial coordinates in a single step—is inherently more susceptible to error propagation. Moreover, the direct position prediction approach could achieve more precise results if augmented with additional features, such as vessel heading, speed, or environmental conditions. However, obtaining these supplementary features in a clean and reliable form is often challenging, as they are typically provided as reported values (or from the vessel captain) that may lack consistency and accuracy. Thus, the results highlight that the velocity-based prediction strategy not only delivers enhanced precision and consistency but also circumvents the difficulties associated with relying on auxiliary features for direct position prediction.

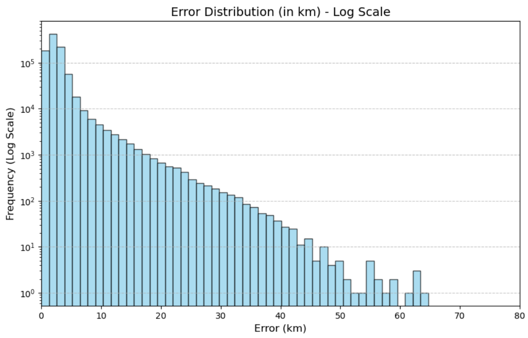


Fig. 3: Error distribution for velocity-based prediction.

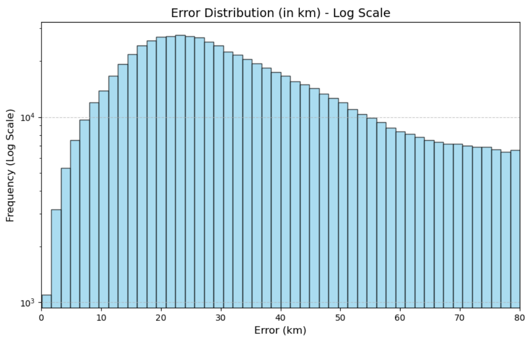


Fig. 4: Error distribution for direct position (latitude and longitude) prediction.

Additionally, the comparative analysis of vessel trajectory prediction strategies presented in Figure 5 highlights the clear advantages of the velocity-based prediction approach over direct position prediction. The results demonstrate that the velocity-based method consistently produces trajectories that closely follow the actual vessel movement, maintaining high accuracy across different trajectory patterns. This stability is due to its incremental nature, where predicted velocity components are integrated over time, reducing the risk of cumulative errors. In contrast, the direct position prediction approach exhibits significant deviations, particularly in segments with complex or nonlinear paths, where errors propagate more rapidly. The observed discrepancies suggest that predicting absolute latitude and longitude values directly is inherently more challenging, as it requires the model to capture spatial and temporal dependencies simultaneously without leveraging incremental updates. Moreover, direct position prediction lacks the contextual information necessary to correct deviations over time, which further limits its reliability. Therefore, by leveraging velocity predictions, the proposed method not only improves accuracy but also ensures greater robustness, making it a more suitable framework for large-scale vessel trajectory forecasting and autonomous navigation applications.

B. Comparison with DL-based models

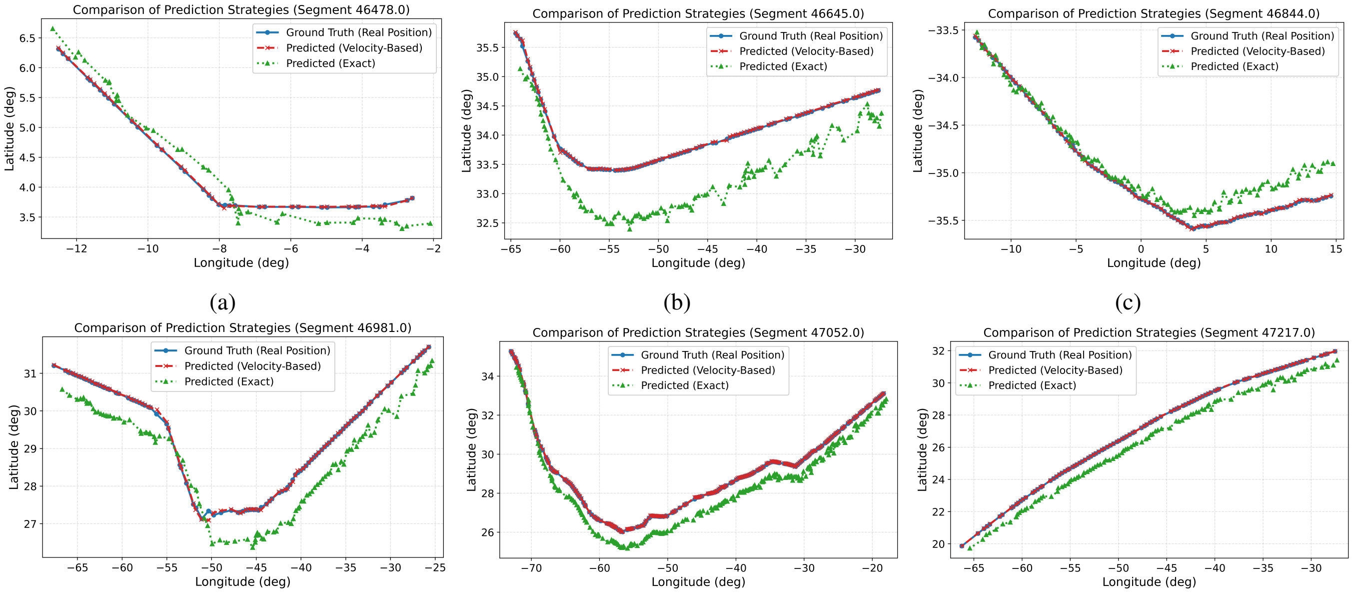
In this section, we compare our `LSTM-RA` model against several DL-based models such as transformer, standard LSTM, BiLSTM, and GRU, which are widely used in state-of-the-art maritime trajectory prediction [3].

The results presented in Table I demonstrate the superior performance of our proposed model, which integrates an attention mechanism and residual connections into the LSTM framework. Our model achieves an MAE of 0.0164, an MSE of 0.00063, and an RMSE of 0.02523, corresponding to an average positional error of 2.62 km. In comparison, the standard LSTM model reports an average error of 5.74 km, and the GRU model yields an average error of 3.21 km. Thus, our model reduces the average error by approximately 54% relative to the standard LSTM (from 5.74 km to 2.62 km) and by roughly 18% compared to the GRU model. These improvements can be attributed to two key architectural features. First, the attention mechanism effectively identifies and emphasizes critical temporal patterns in the input data, enabling the model to focus on the most informative segments of a vessel's trajectory. Second, the integration of residual connections facilitates improved gradient flow across the network, allowing for better capture of long-term dependencies and reducing the risk of vanishing gradients.

TABLE I: Prediction performance for GRU, LSTM, and BiLSTM Models.

Model	mae	mse	rmse	average error (km)
BiLSTM	0.0176	0.0007	0.0270	2.81
GRU	0.0205	0.0008	0.0289	3.21
LSTM	0.0356	0.0022	0.0479	5.74
Transformer	0.0212	0.0006	0.0254	3.41
LSTM-RA (Our model)	**0.0164**	**0.0006**	**0.0252**	**2.62**

Table II provides a comparative analysis of the computational costs associated with each model. While our model requires 2211 seconds for training and 593 seconds for inference, with an energy consumption of 0.318 Wh, these values are competitive when balanced against the substantial gains in prediction accuracy. For instance, even though the GRU model offers faster training and inference times with lower energy consumption, it does so at the expense of a higher average error. Overall, our proposed approach achieves an excellent trade-off between predictive performance and computational efficiency, making it well-suited for real-world maritime trajectory prediction tasks.



(a) (b) (c)

(d) (e) (f)

Fig. 5: Comparison of vessel trajectory prediction strategies with some segments in the Atlantic area.

TABLE II: Time (s) and energy cost (Wh).

Model	Training time	Inference	Energy
BiLSTM	2460	453	0.321
GRU	1141	134	0.133
LSTM	1604	201	0.191
Transformer	13996	238	1.92
LSTM-RA (Our model)	2211	593	0.318

VI. Performance Evaluation on the Malaysia Dataset

In this section, we extend our evaluation to the Malaysia dataset with dual scenarios. In the first scenario, the `LSTM-RA` model is fully trained on the Malaysia dataset, allowing it to learn the unique spatiotemporal dynamics and patterns of the region. This setup establishes a robust in-domain performance benchmark by enabling the model to tailor its predictions to the specific characteristics of the dataset. In the second scenario, we test the cross-domain generalization capabilities of the model trained on the Atlantic area, called zero-shot learning. In this zero-shot evaluation setting, the model is directly applied to the Malaysia data without additional retraining or fine-tuning, providing valuable insights into its ability to generalize across geographically distinct domains.

The comparative results are presented in Figure 6, showing a clear performance difference between the two approaches. The in-domain training achieves an average error of 0.91 km, demonstrating its ability to learn and adapt to the specific dynamics of the Malaysia dataset. In contrast, the zero-shot learning yields a slightly higher average error of 1.26 km, indicating a reasonable level of cross-domain generalization despite the lack of retraining. These findings demonstrate the effectiveness of our `LSTM-RA` model and velocity-based prediction approach, as they demonstrate competitive performance in both in-domain and cross-domain scenarios. While in-domain training naturally achieves superior accuracy due to its dataset-specific optimization, the zero-shot learning results affirm the robustness and the generalization of our approach, which performs well even without adaptation.

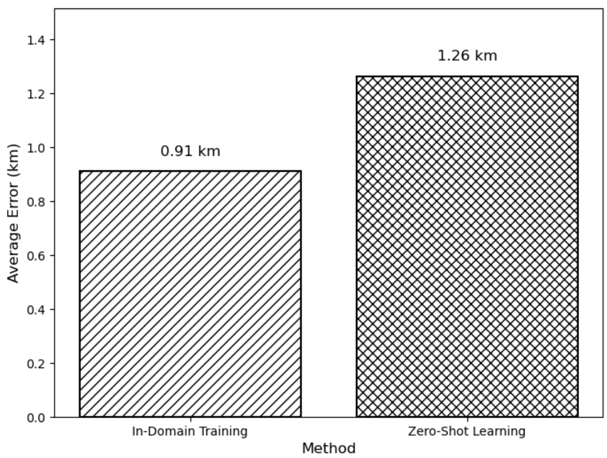


Fig. 6: Average prediction errors (in km) for the Malaysia dataset.

VII. Ablation Study

This section analyzes two complementary aspects that influence the performance of the proposed framework: (i) the impact of evaluation strategies on prediction accuracy, and (ii) the contribution of the architectural components (residual connections and attention mechanisms) to model effectiveness.

A. Impact of Evaluation Strategies

Figure 7 compares the proposed model with a conventional Moving Average (MA) baseline for velocity-based trajectory prediction. A significant performance gap is observed between the two approaches. On the *Atlantic* dataset, the MA model yields an average prediction error of 23.59 km, while our proposed model achieves only 2.62 km. Similarly, on the *Malaysia* dataset, the MA model records an error of 9.08 km. The poor performance of the MA model demonstrates its inability to capture nonlinear and time-varying vessel dynamics, particularly in regions with complex traffic patterns such as the Atlantic. In contrast, our model, enhanced with residual connections and attention mechanisms, effectively learns temporal dependencies and nonlinear velocity variations, leading to substantially improved predictive accuracy.

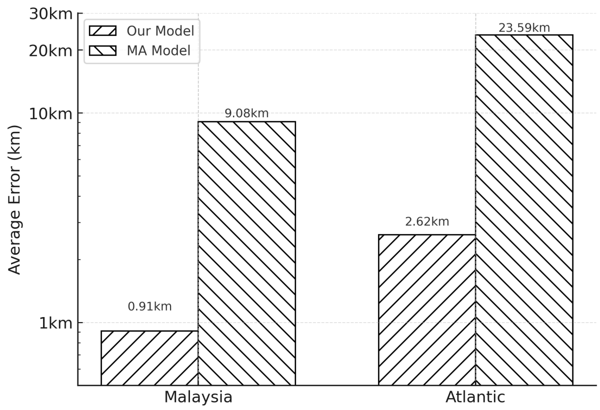


Fig. 7: Performance evaluation of our model vs. the MA model for velocity-based prediction.

To further assess the influence of evaluation methodology, Figure 8 compares results obtained under MMSI-based (without segmentation) and segmentation-based evaluation schemes across both datasets. When adopting the MMSI-based approach—treating each vessel's full trajectory as a single sequence—the average prediction error increases to 2.92 km for the Atlantic region and 5.36 km for the Malaysia dataset. This degradation occurs because AIS messages are temporally heterogeneous: the same vessel (MMSI) can report irregularly, experience signal gaps, or represent multiple voyages in a single trajectory. Consequently, MMSI-based grouping often merges discontinuous trips, distorting temporal dependencies and impairing model learning. In contrast, segmentation-based evaluation partitions AIS data into temporally coherent sub-trajectories, better capturing intra-vessel motion continuity and producing more reliable error estimates. These results demonstrate that segmentation-based evaluation provides a more strong measure of predictive performance in real-world AIS conditions.

B. Impact of Model Components

To assess the contribution of individual architectural elements, we performed a detailed ablation analysis by selectively removing the residual connections and attention mechanism from the proposed `LSTM-RA` model. The results, summarized in Table III, clearly show the incremental improvements brought by each component.

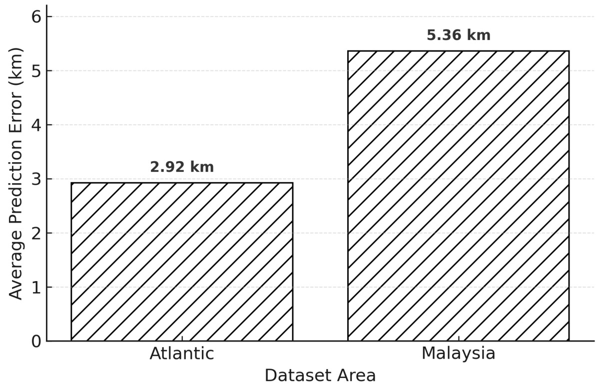


Fig. 8: Comparison of average prediction errors (km) for MMSI-based evaluation on the Atlantic and Malaysia datasets.

The baseline LSTM exhibits the highest error (5.74 km), reflecting its limited capacity to capture long-term temporal dependencies in noisy AIS sequences. Introducing residual connections (*LSTM + Residual*) significantly reduces the error to 3.39 km by improving gradient flow, mitigating vanishing gradients, and stabilizing training. Incorporating only the attention mechanism (*LSTM + Attention*) yields a further improvement to 2.98 km, as the model learns to focus on the most relevant temporal observations within each input window. The combination of both mechanisms in the final `LSTM-RA` model achieves the lowest prediction error of 2.62 km, confirming that residual and attention components complement each other by enhancing both convergence stability and temporal selectivity.

TABLE III: Ablation study results for the proposed LSTM-RA model.

Model Variant	MAE	MSE	RMSE	Average Error (km)
LSTM (Baseline)	0.0356	0.0022	0.0479	5.74
LSTM+Residual	0.0209	0.0009	0.0301	3.39
LSTM+Attention	0.0187	0.0007	0.0265	2.98
LSTM-RA (our model)	**0.0164**	**0.0006**	**0.0252**	**2.62**

Table IV further compares the training and inference efficiency of each variant. Although adding attention slightly increases computation time and energy cost, the performance gain justifies the trade-off, especially for deployment scenarios requiring reliable long-horizon predictions. As illustrated in Figure 9, the proposed `LSTM-RA` achieves the fastest and most stable convergence across epochs, demonstrating efficient learning dynamics and robust generalization compared to its reduced variants.

TABLE IV: Training and inference efficiency of ablation variants.

Model Variant	Training Time (s)	Inference (s)	Energy (Wh)
LSTM (Baseline)	1604	201	0.191
LSTM+Residual	1875	243	0.226
LSTM+Attention	2030	317	0.261
LSTM-RA (our model)	**2211**	**593**	**0.318**

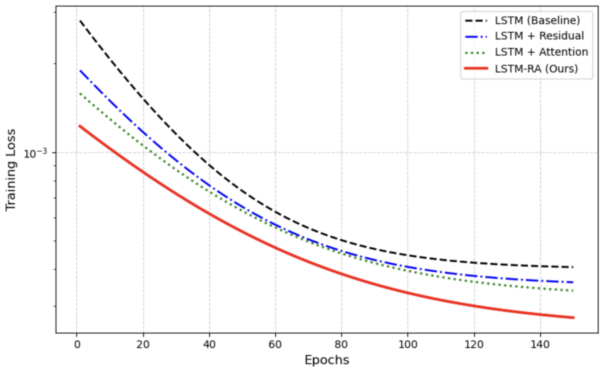


Fig. 9: Training loss convergence of ablation variants.

Overall, the ablation analysis validates that both residual learning and temporal attention are key enablers of the proposed `LSTM-RA` framework, jointly improving accuracy, convergence speed, and robustness in velocity-based vessel trajectory prediction.

VIII. Conclusion

In this work, we proposed a robust and scalable deep learning framework for maritime trajectory prediction based on velocity estimation, tailored to the challenges posed by noisy and irregularly sampled AIS data. By shifting the modeling focus from direct position forecasting to latitude- and longitude-velocity prediction, our approach improves accuracy, stability, and interpretability. The use of LSTM units enhanced with attention mechanisms and residual connections enables the model to effectively capture complex temporal dependencies in vessel movement, even under sparse or fragmented input conditions. Extensive experiments conducted on realistic datasets from the Atlantic and Malaysia regions demonstrated that our velocity-based model significantly outperforms conventional baselines, including moving average and direct position-prediction models. The ablation studies further showed the importance of careful data segmentation and trajectory grouping strategies, highlighting that segmentation-based structuring consistently yields superior predictive accuracy compared to MMSI-based grouping. Overall, this study underscores the value of velocity-centric learning and principled data preprocessing in enhancing predictive performance in maritime applications. Our results pave the way for more reliable and computationally efficient trajectory forecasting frameworks in next-generation satellite-driven and AIS-integrated NTN ecosystems. Future work will explore multimodal learning strategies incorporating environmental and vessel-specific features to further improve generalization in complex operational scenarios.

IX. Acknowledgement

This work is partially supported by the Luxembourg National Research Fund, and by the European Space Agency, ESA ESTEC, Noordwijk, The Netherlands, under the project title ”MAGINET: Geospatial Information Networking Techniques for Maritime Awareness Services”. Opinions, interpretations, recommendations, and conclusions expressed herein are those of the authors and are not necessarily endorsed by the European Space Agency.

This research was funded by the Luxembourg National Research Fund (FNR), grant reference NCER22/IS/16570468. For the purpose of open access, and in fulfillment of the obligations arising from the grant agreement, the authors have applied a Creative Commons Attribution 4.0 International (CC BY 4.0) license to any Author Accepted Manuscript version arising from this submission.

References

[1] M. M. Azari, S. Solanki, S. Chatzinotas, O. Kodheli, H. Sallouha, A. Colpaert, J. F. Mendoza Montoya, S. Pollin, A. Haqiqatnejad, A. Mostaani *et al.*, “Evolution of non-terrestrial networks from 5G to 6G: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 24, no. 4, pp. 2633–2672, 2022.

[2] S. Saafi, O. Vikhrova, G. Fodor, J. Hosek, and S. Andreev, “AI-aided integrated terrestrial and non-terrestrial 6G solutions for sustainable maritime networking,” *IEEE Network*, vol. 36, no. 3, pp. 183–190, 2022.

[3] D. Yang, L. Wu, S. Wang, H. Jia, and K. X. Li, “How big data enriches maritime research—a critical review of automatic identification system (AIS) data applications,” *Transport Reviews*, vol. 39, no. 6, pp. 755–773, 2019.

[4] H. Li, H. Jiao, and Z. Yang, “AIS data-driven ship trajectory prediction modelling and analysis based on machine learning and deep learning methods,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 175, p. 103152, 2023.

[5] ——, “Ship trajectory prediction based on machine learning and deep learning: A systematic review and methods analysis,” *Engineering Applications of Artificial Intelligence*, vol. 126, p. 107062, 2023.

[6] H. Li, W. Xing, H. Jiao, Z. Yang, and Y. Li, “Deep bi-directional information-empowered ship trajectory prediction for maritime autonomous surface ships,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 181, p. 103367, 2024.

[7] X. Zhang, X. Fu, Z. Xiao, H. Xu, and Z. Qin, “Vessel trajectory prediction in maritime transportation: Current approaches and beyond,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 19 980–19 998, 2022.

[8] S. Wang, Y. Li, H. Xing, and Z. Zhang, “Vessel trajectory prediction based on spatio-temporal graph convolutional network for complex and crowded sea areas,” *Ocean Engineering*, vol. 298, p. 117232, 2024.

[9] T. Xiaopeng, C. Xu, S. Lingzhi, M. Zhe, and W. Qing, “Vessel trajectory prediction in curving channel of inland river,” in *2015 International Conference on Transportation Information and Safety (ICTIS)*. IEEE, 2015, pp. 706–714.

[10] F. Mazzarella, V. F. Arguedas, and M. Vespe, “Knowledge-based vessel position prediction using historical AIS data,” in *2015 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*. IEEE, 2015, pp. 1–6.

[11] J. Liu, G. Shi, and K. Zhu, “Vessel trajectory prediction model based on AIS sensor data and adaptive chaos differential evolution support vector regression (ACDE-SVR),” *Applied Sciences*, vol. 9, no. 15, p. 2983, 2019.

[12] C. Zhang, J. Bin, W. Wang, X. Peng, R. Wang, R. Halldearn, and Z. Liu, “AIS data driven general vessel destination prediction: A random forest based approach,” *Transportation Research Part C: Emerging Technologies*, vol. 118, p. 102729, 2020.

[13] C.-H. Yang, G.-C. Lin, C.-H. Wu, Y.-H. Liu, Y.-C. Wang, and K.-C. Chen, “Deep learning for vessel trajectory prediction using clustered AIS data,” *Mathematics*, vol. 10, no. 16, p. 2936, 2022.

[14] R. W. Liu, M. Liang, J. Nie, W. Y. B. Lim, Y. Zhang, and M. Guizani, “Deep learning-powered vessel trajectory prediction for improving smart traffic services in maritime internet of things,” *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 5, pp. 3080–3094, 2022.

[15] W. Wu, P. Chen, L. Chen, and J. Mou, “Ship trajectory prediction: An integrated approach using ConvLSTM-based sequence-to-sequence model,” *Journal of Marine Science and Engineering*, vol. 11, no. 8, p. 1484, 2023.

[16] R. W. Liu, M. Liang, J. Nie, Y. Yuan, Z. Xiong, H. Yu, and N. Guizani, “STMGCN: Mobile edge computing-empowered vessel trajectory prediction using spatio-temporal multigraph convolutional network,” *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 7977–7987, 2022.

[17] R. W. Liu, W. Zheng, and M. Liang, “Spatio-temporal multi-graph transformer network for joint prediction of multiple vessel trajectories,” *Engineering Applications of Artificial Intelligence*, vol. 129, p. 107625, 2024.

[18] J. Zhao, Z. Yan, Z. Zhou, X. Chen, B. Wu, and S. Wang, “A ship trajectory prediction method based on GAT and LSTM,” *Ocean Engineering*, vol. 289, p. 116159, 2023.

[19] P. Han, M. Zhu, and H. Zhang, “Interaction-aware short-term marine vessel trajectory prediction with deep generative models,” *IEEE Transactions on Industrial Informatics*, vol. 20, no. 3, pp. 3188–3196, 2023.

[20] T. C. A. Nguyen, T. M. Tao, C. Lee, and C.-H. Youn, “TM-LLM: Token-motion embedding fusion for large language model-based vessel trajectory prediction,” in *2024 15th International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2024, pp. 115–120.

[21] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[23] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1492–1500.