Proceedings of the 4th International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/2026.37375

Vessel Trajectory Deviation Detection Using a Temporal-Attention LSTM Autoencoder

Kehao Bao

School of Computer Science, Sun Yat-sen University, Guangzhou, China
baokh3@mail2.sysu.edu.cn

Abstract. Abnormal vessel trajectories are critical indicators of potential collision risks, illegal navigation behaviors, and unexpected route deviations. This study uses labelled Automatic Identification System (AIS) trajectories from Danish waters to examine a Long Short-Term Memory (LSTM) autoencoder with temporal attention for vessel trajectory deviation detection. Latitude, longitude, speed, and course are processed as fixed-length sequences, and the model is trained only with normal trajectories. Reconstruction error is then used as the anomaly score. The model is compared with Isolation Forest, a dense autoencoder, a basic LSTM autoencoder, and a position-only LSTM model under the same Maritime Mobile Service Identity (MMSI)-based data split. Multi-seed repeated experiments over five independent runs show that the attention LSTM obtains a mean receiver operating characteristic area under the curve (ROC-AUC) of 0.859, recall of 0.725, and F1-score of 0.724. The improvement over the basic LSTM is small, while the dense autoencoder gives a slightly higher average precision. The results also show that speed and course are important for detecting abnormal movement. A sequence length and hidden size of 32, together with the 95th-percentile threshold, give the most balanced results. The dataset contains few abnormal samples, so further testing on larger datasets and different sea areas is still needed.

Keywords: vessel trajectory anomaly detection, automatic identification system, LSTM autoencoder, temporal attention, maritime safety

1. Introduction

Maritime transportation is closely related to international trade and the safety of people and property at sea [1]. With the development of global trade, the number of ships and maritime activities continues to increase, and the problem of ship collision, illegal operation and unusual route change gradually receives more attention. Therefore, it is meaningful to detect the abnormal movement of ships as early as possible. The Automatic Identification System (AIS) can record the identity, position, speed and course of ships [2]. AIS data can not only help maritime departments understand the traffic situation, but can also become a basis for identifying abnormal vessel behaviors. Currently, how to use historical AIS data to detect abnormal vessel trajectories has become an important issue in maritime traffic analysis [3].

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.37375

At present, machine learning and deep learning are widely used in vessel trajectory analysis [3, 4]. However, traditional studies are based on rule formulation, distance calculation and trajectory clustering. In these methods, the threshold and the definition of abnormal behaviour need expert experience and may not be suitable for every sea area. Under this background, the autoencoder is used in anomaly detection because it can learn the main features of normal data with limited labeled abnormal samples [5]. When the autoencoder is trained by normal trajectories, it tries to reconstruct the input trajectory. A normal trajectory is similar to the data learned by the model, so its reconstruction error is usually small, while an abnormal trajectory may have a larger error. Long Short-Term Memory (LSTM) can process sequential data and learn the relationship between different time points [6]. Some studies have also used LSTM-related models in vessel trajectory prediction and abnormal behaviour detection [7].

Although the LSTM autoencoder has the advantage in processing sequential trajectory data, its performance will be influenced by the data and input features. Different ships have different normal speeds and routes, and traffic patterns also change among sea areas. Real abnormal vessel trajectories are rare, and it is difficult to obtain enough data with reliable labels [4, 8]. Meanwhile, a complicated model may not always perform better when the number of training samples is limited. The Technical University of Denmark (DTU) published the AIS Trajectories from Danish Waters for Abnormal Behavior Detection dataset, which contains normal and abnormal vessel trajectories [8, 9]. The abnormal trajectories are related to a collision, Search and Rescue activity, law-enforcement vessels and commercial ships which changed their routes. Therefore, this dataset provides authentic real-world samples for trajectory anomaly detection. However, the labelled trajectories mainly come from one day and one accident event, so it is still necessary to compare different methods and analyse whether the LSTM structure is useful.

Given the above, this paper studies vessel trajectory deviation detection based on AIS data. The LSTM autoencoder uses latitude, longitude, speed and course to learn the normal trajectory pattern, and the reconstruction error is used as the anomaly score. In the experiment, the LSTM autoencoder will be compared with Isolation Forest and a standard autoencoder under the same data division. A position-only experiment will also analyse the influence of speed and course. Through the comparison of different methods, this paper aims to analyse the performance of the LSTM autoencoder, providing valuable references for subsequent related research.

2. Related work

Existing studies on vessel trajectory anomaly detection mainly use rule-based methods, statistical analysis, trajectory clustering and deep learning. A recent review summarized the main methods used in maritime anomaly detection and identified AIS as a common data source in this field [4]. Rule-based methods usually identify abnormal speed, course or sailing area according to selected thresholds. Such methods are intuitive and interpretable, whereas their threshold settings heavily rely on domain expertise and require adaptive adjustments for different sea areas and vessel types. In contrast, data-driven methods automatically mine normal navigation patterns from historical trajectory data and achieve superior generalization performance.

Trajectory clustering is a common data-driven approach. Pallotta et al. extracted normal vessel routes from AIS data and adopted the learned traffic patterns for trajectory prediction and anomaly identification [3]. A trajectory may be considered abnormal when it is very different from normal routes in the same area. Olesen et al. divided trajectory information into position and kinematic information and then clustered normal traffic patterns to detect abnormal speed and course with contextual information [8]. Although clustering-based methods can explicitly characterize

DOI: 10.54254/2755-2721/2026.37375

mainstream maritime traffic routes, their performance is sensitive to distance metrics, hyperparameter settings, and complex local traffic conditions.

Autoencoders provide another way to detect anomalies when labelled abnormal samples are limited. Sakurada and Yairi used an autoencoder to learn a low-dimensional representation of normal data and detected anomalies according to reconstruction error [5]. A model trained on normal samples usually reconstructs normal inputs more accurately, while unusual inputs may produce larger errors. This idea is suitable for AIS data because real abnormal vessel trajectories are rare. For vessel trajectories, reconstruction error can be calculated for the whole trajectory or for each time point. The first form gives a simple trajectory-level decision, while the second can help locate where an abnormal change begins. However, a standard autoencoder does not directly describe the order between different time points.

LSTM networks are designed for sequential data and can learn relationships over time [6]. Venskus et al. used an LSTM-related method for vessel trajectory prediction, showing that temporal information can be used to model vessel movement [7]. Zhao and Shi combined Density-Based Spatial Clustering of Applications with Noise (DBSCAN) with an LSTM trajectory predictor and identified abnormal speed, course, and route by measuring the discrepancy between predicted and observed trajectories [10]. Liang et al. developed an unsupervised deep-learning framework that maps vessel trajectories into images and derives anomaly scores from a generative adversarial network and an encoder [11]. These studies show that one method can predict a trajectory, while another can reconstruct an observed trajectory and use the error as an anomaly score. However, such data-hungry models require sufficient training samples, and overly complex architectures easily suffer from performance degradation on small-scale maritime datasets.

In summary, existing methods have their own advantages and limitations. Rule-based and clustering methods are relatively easy to explain, while deep learning methods can learn more complicated trajectory changes. However, previous studies use different sea areas, datasets and definitions of abnormal behaviour, so their results are difficult to compare directly. Therefore, this paper uses the same public DTU dataset and data division to compare the LSTM autoencoder with a standard autoencoder and Isolation Forest. Isolation Forest is used as the traditional anomaly-detection baseline [12]. The standard autoencoder helps to examine whether the LSTM structure provides additional value. A position-only LSTM experiment is also conducted to analyse the influence of speed and course. Recent work has also combined clustering with deep recurrent networks for real-time ship anomaly detection [13]. Broader surveys of machine-learning and deep-learning trajectory prediction methods identify recurrent networks as a common approach for AIS sequence modelling [14]. This comparison also helps to distinguish whether any improvement comes from the LSTM temporal structure or simply from the use of an autoencoder, while keeping the evaluation conditions consistent across the models. The purpose is to provide a simple and fair evaluation of the LSTM autoencoder for vessel trajectory anomaly detection.

3. Method

An LSTM autoencoder with temporal attention is used to detect vessel trajectory deviation. The model gives one anomaly score for each complete trajectory. Normal trajectories in the same sea area usually have similar route and motion patterns. After learning these patterns, the model reconstructs normal trajectories with small errors, while unusual changes in position, speed, or course tend to produce larger errors. The encoder outputs a hidden state at every time point, and the attention layer combines all hidden states into a trajectory vector. This allows information from the

DOI: 10.54254/2755-2721/2026.37375

beginning and middle of a trajectory to take part in reconstruction instead of relying only on the last hidden state. The complete process is shown in Figure 1.

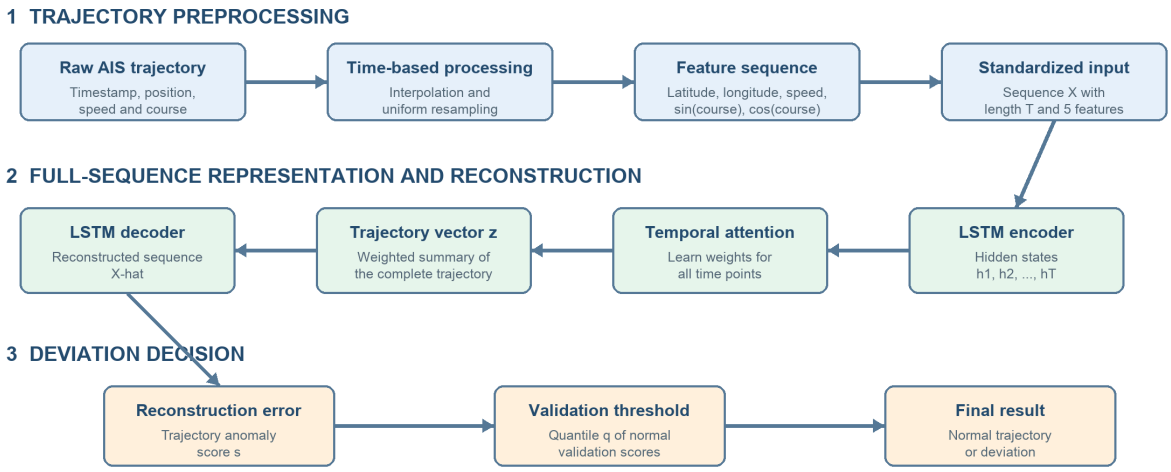

Figure 1. Framework of the temporal-attention LSTM autoencoder for vessel trajectory deviation detection (picture credit: original)

3.1. Trajectory preprocessing

Each AIS trajectory is first sorted by timestamp, and duplicate timestamps are removed. A trajectory contains latitude, longitude, speed, and course at a number of observation times. Missing values between two valid records are filled by linear interpolation. A trajectory is removed if one feature is missing in all its records, because replacing a complete feature with an artificial constant would change the movement pattern.

AIS trajectories have different lengths, and their reporting intervals are not always equal. Resampling is therefore based on the original timestamps rather than on record order alone. The target timestamps are placed at equal intervals between the first and last observation, and each target value is calculated from its two neighbouring records.

$$u_t = \tau_1 + \frac{t-1}{T-1}(\tau_n - \tau_1), t = 1, \ldots, T \tag{1}$$

$$\overline{y}_t = y_k + \frac{u_t - \tau_k}{\tau_{k+1} - \tau_k}(y_{k+1} - y_k), \tau_k \le u_t \le \tau_{k+1}$$

In Equation (1), T denotes the common sequence length. The second row gives the linear interpolation between two neighbouring observations. This operation keeps the first and last observation time and reduces the distortion caused by irregular AIS reporting intervals. The default sequence length is 32, which is close to the median length of the raw trajectories and avoids excessive upsampling or downsampling for most samples. Sequence lengths of 16, 32, and 64 are evaluated in the experiment.

Course is a circular variable. For example, 359 degrees and 0 degrees describe nearly the same direction although their numerical difference is large. Course is therefore changed into sine and cosine before resampling. Each time point has five input features: latitude, longitude, speed, course sine, and course cosine. The complete input is a time-ordered matrix with T rows and five columns.

DOI: 10.54254/2755-2721/2026.37375

$$x_t = [\varphi_t, \lambda_t, v_t, sin(\theta_t), cos(\theta_t)] \tag{2}$$

$$X = [x_1, \ldots, x_T]^T \in R^{T \times F}, F = 5$$

$$\widetilde{x}_{t,j} = \frac{x_{t,j} - \mu_j}{\sigma_j{}^*}$$

$$\sigma_j{}^* = \sigma_j \quad if \quad \sigma_j \geq 10^{-6}; otherwise 1$$

The first two rows of Equation (2) define the input vector and trajectory matrix, while the last two rows describe feature standardization. Since the five features have different units and ranges, the mean and standard deviation of each feature are calculated separately from the normal training set. A standard deviation below 10^{-6} is replaced by 1 to avoid unstable division. The same training-set statistics are used for the validation and test sets. Data are divided by vessel Maritime Mobile Service Identity (MMSI), so trajectories from one vessel do not appear in different subsets. Sample numbers and subset proportions are reported in the experiment section.

3.2. Attention-based LSTM autoencoder

The autoencoder contains an LSTM encoder, a temporal attention layer, an LSTM decoder, and a linear output layer. The encoder reads the standardized sequence from the first time point to the last. At each step, the input, forget, and output gates control how much new information is written, kept, and passed to the hidden state. The calculation follows the standard LSTM formulation [6, 15].

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$

$$g_t = tanh(W_g x_t + U_g h_{t-1} + b_g)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t$$

$$h_t = o_t \odot tanh(c_t)$$

In Equation (3), the first three rows calculate the input, forget, and output gates, and the fourth row gives the candidate memory. The last two rows update the cell state and hidden state. The weight matrices and bias vectors are learned during training. The sigmoid function controls the amount of information passing through each gate, while the hyperbolic tangent limits the candidate values.

A basic LSTM autoencoder often represents the complete trajectory by the last hidden state. Although the LSTM gates can keep long-term information, this final compression may weaken changes that occur near the beginning or middle of a route. In the proposed structure, the encoder retains the hidden state of every time point, and the attention layer assigns a weight to each state.

$$e_t = {v_a}^T tanh(W_a h_t + b_a), \alpha_t = \frac{exp(e_t)}{\sum_{k=1}^{T} exp(e_k)} \tag{4}$$

$$z = \sum_{t=1}^{T} \alpha_t h_t, \widehat{X} = D(R_T(z)), \widehat{X} \in R^{T \times F}$$

DOI: 10.54254/2755-2721/2026.37375

The first row of Equation (4) calculates an importance score for every hidden state and normalizes the scores by softmax. The resulting attention weights sum to one. The second row forms the trajectory vector from the weighted hidden states. This vector is repeated T times and sent to the decoder, and a linear layer changes the decoder output back to five features. The reconstructed matrix therefore has the same shape as the input. The attention weights also show which time periods contribute more to the trajectory representation.

The default hidden size is 32, which is a middle setting between 16 and 64 units. A smaller hidden state may not represent different movement patterns sufficiently, whereas a larger state adds more parameters and increases the risk of overfitting on a small dataset. The three hidden sizes are compared by normal validation loss.

3.3. Model training and deviation decision

Only normal trajectories are used for model training. The available abnormal labels come from a small number of events, and these events do not follow one common pattern. Including them in reconstruction training may allow the autoencoder to reproduce abnormal movement and reduce the difference between normal and abnormal scores. The learned normal pattern is limited to the selected DTU sea area and is not intended to cover all possible normal behaviour. Normal trajectories from MMSI values unseen during training form the validation set, which tests whether the representation can be transferred to other vessels in the same area.

The model minimizes the mean squared reconstruction error over all normal training trajectories, time points, and features.

$$L(\Theta) = \frac{1}{NTF} \sum_{i=1}^{N} \sum_{t=1}^{T} \sum_{j=1}^{F} (x_{i,t,j} - \hat{x}_{i,t,j})^2 \quad (5)$$

Equation (5) averages the squared reconstruction error over all normal training trajectories, time points, and features. The model with the smallest normal validation loss is retained. Training and validation losses are examined together. High values for both losses indicate insufficient model capacity, while a low training loss together with a much higher validation loss indicates overfitting. The learning rate, batch size, optimizer, maximum epochs, early-stopping rule, and regularization settings are reported in the experiment section.

After training, the reconstruction error of each trajectory is used as its anomaly score. A larger score means that the trajectory is less similar to the learned normal patterns. The decision threshold is obtained from the score distribution of the normal validation set.

$$s_i = \frac{1}{TF} \sum_{t=1}^{T} \sum_{j=1}^{F} (x_{i,t,j} - \hat{x}_{i,t,j})^2 \quad (6)$$

$$\tau_q = Q_q(\{s_i : i \in V_{normal}\}), \hat{y}_i = I(s_i > \tau_q)$$

The first row of Equation (6) calculates the mean reconstruction error of one trajectory. The second row defines the threshold as a quantile of the normal validation scores and gives the final decision rule. A trajectory is marked as a deviation when its score is higher than the threshold. The 90th, 95th, and 99th percentiles are evaluated in the experiment, and abnormal test labels are not used to calculate these thresholds. The error at each time point is also retained to locate the beginning of an unusual change.

DOI: 10.54254/2755-2721/2026.37375

4. Experiment

Experiments were carried out on the labelled DTU AIS dataset to evaluate the performance of the proposed temporal attention-enhanced LSTM autoencoder. The main experiment compares it with several baseline models. Other experiments change the input features, sequence length, hidden size, and anomaly threshold. All competing models were trained and evaluated under identical preprocessing pipelines and MMSI-based data splitting strategies to ensure fair comparison.

4.1. Dataset and experimental settings

The dataset was published by the Technical University of Denmark [9]. It contains 423 trajectories from Danish waters, with 399 normal trajectories and 24 abnormal trajectories. The shortest raw trajectory has 6 AIS points and the longest one has 70 points. The median length is 37 points. Among these data, 16 normal trajectories do not have any valid course value. Since course is one of the input features, these trajectories were removed instead of being filled with an artificial value. After this step, 407 valid trajectories from 281 vessels were used, including 383 normal trajectories and all 24 abnormal trajectories.

The normal data were divided into training, validation, and test sets by MMSI. All abnormal trajectories were placed in the test set. When a vessel had an abnormal trajectory, its normal trajectories were also placed in the test set. In this way, one vessel does not appear in both training and testing. Taking seed 42 as an example, there are 242 normal trajectories in the training set and 62 normal trajectories in the validation set. The test set contains 103 trajectories, of which 79 are normal and 24 are abnormal. The main comparison was run with five seeds: 42, 52, 62, 72, and 82. The parameter experiment used three seeds, namely 42, 52, and 62, because each parameter setting needs to train the model again. Both the data split and the initial network weights change with the seed. For this reason, the mean and standard deviation are reported instead of one result.

Isolation Forest was selected as a traditional non-sequential baseline [12]. A dense autoencoder was included as another non-sequential baseline. A basic LSTM autoencoder without attention was used to check whether the attention layer is helpful. A position-only LSTM was also tested. It only uses latitude and longitude, while the full model also uses speed, course sine, and course cosine. The sequence length and hidden size were set to 32 in the main experiment. The split and standardization method were the same for every model, and only normal validation scores were used to calculate the threshold.

Table 1. Training settings of the proposed model

Setting	Value
Input features	Latitude, longitude, speed, course sine, course cosine
Sequence length	32
Encoder and decoder hidden size	32
Loss function	Mean squared error
Optimizer	Adam

DOI: 10.54254/2755-2721/2026.37375

Table 1. (continued)

Learning rate	0.001
Batch size	32
Maximum epochs	180
Early-stopping patience	20 epochs
Dropout and weight decay	0.10 and 1×10^{-5}
Default threshold	95th percentile of normal validation scores

The training settings are listed in Table 1. In each run, the model with the lowest validation reconstruction loss was saved. Receiver operating characteristic area under the curve (ROC-AUC) and average precision (AP) measure the ranking of the anomaly scores and do not depend on one selected threshold. Precision, recall, and F1-score were calculated after using the validation threshold. Due to severe class imbalance with extremely limited anomalous samples, overall accuracy is excluded from core evaluation metrics to avoid biased assessment.

4.2. Comparison with baseline models

Table 2. Anomaly detection results over five runs

Model	ROC-AUC	AP	Precision	Recall	F1-score
Isolation Forest	0.643 ± 0.044	0.339 ± 0.069	0.187 ± 0.049	0.050 ± 0.019	0.075 ± 0.013
Dense autoencoder	0.858 ± 0.005	0.668 ± 0.017	0.694 ± 0.044	0.708 ± 0.029	0.700 ± 0.017
Basic LSTM autoencoder	0.848 ± 0.015	0.636 ± 0.024	0.745 ± 0.050	0.692 ± 0.086	0.714 ± 0.053
Attention LSTM autoencoder	0.859 ± 0.011	0.656 ± 0.033	0.730 ± 0.099	0.725 ± 0.037	0.724 ± 0.053
Position-only attention LSTM	0.635 ± 0.052	0.296 ± 0.031	0.090 ± 0.124	0.025 ± 0.037	0.039 ± 0.057

Table 2 reports the mean and standard deviation of five runs. The attention LSTM autoencoder has the highest ROC-AUC of 0.859, recall of 0.725, and F1-score of 0.724. The basic LSTM autoencoder obtains an ROC-AUC of 0.848 and an F1-score of 0.714. Thus, adding attention increases both values by about 0.011. This gain is small, so the result does not show a clear advantage of attention. It only suggests that the hidden states from different time points may provide some extra information on this dataset. More data and tests in other sea areas are needed before a stronger conclusion can be made.

The dense autoencoder also gives a good result. Its AP is 0.668, which is slightly higher than the attention LSTM value of 0.656, while its F1-score of 0.700 is lower than 0.724. AP measures the ranking of anomaly scores over different thresholds, whereas F1-score is calculated at the selected 95th-percentile threshold. Therefore, a model can have a higher AP but a lower F1-score at one threshold. Another possible reason is that the trajectories are short and the dataset is small, so the

DOI: 10.54254/2755-2721/2026.37375

simpler dense autoencoder may be easier to train and can still rank the abnormal samples well. The temporal model is therefore not better on every measure. Isolation Forest has an F1-score of only 0.075 and detects 5.0% of the abnormal trajectories at the 95th-percentile threshold. Its flattened feature representation does not directly describe how a trajectory changes over time.

After speed and course were removed, the position-only model obtained an ROC-AUC of 0.635 and an F1-score of 0.039. These values are much lower than those of the five-feature model. Therefore, latitude and longitude alone cannot describe all the labelled abnormal trajectories in this dataset. Velocity and course features provide critical kinematic information, capturing abnormal motion variations even when spatial trajectories remain normal.

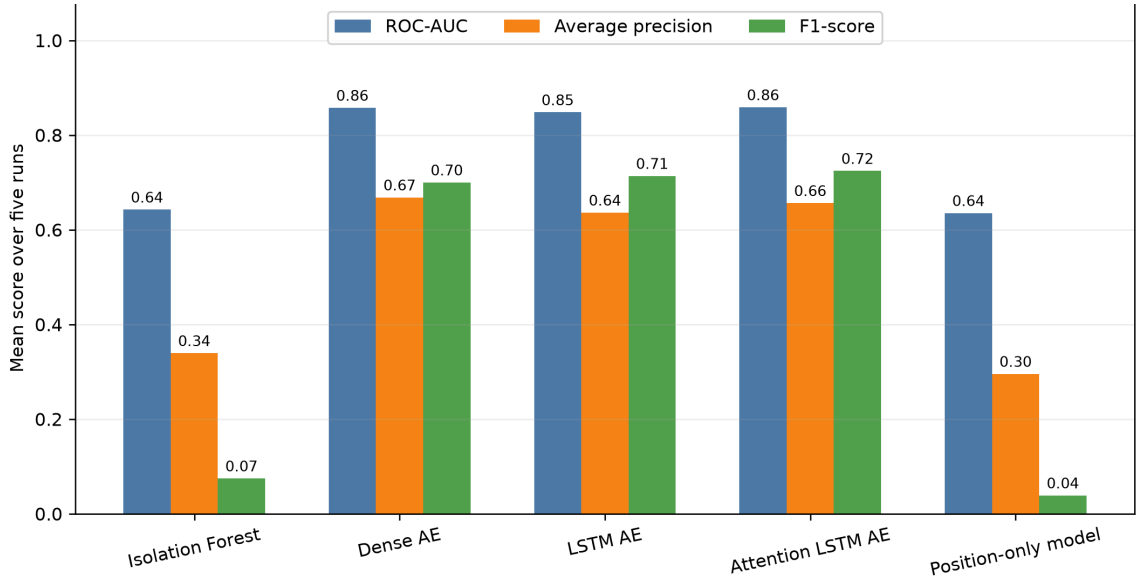

Figure 2. Performance comparison of the five anomaly detectors (picture credit: original)

The difference between the models can also be seen in Figure 2. The dense autoencoder and the two full-feature LSTM models are relatively close, while Isolation Forest and the position-only model are lower. Among these models, the attention LSTM has the highest F1-score and ROC-AUC.

4.3. Parameter and threshold analysis

Table 3. Parameter sensitivity over three runs

Parameter	Value	Validation loss	ROC-AUC	AP	F1-score
Sequence length	16	0.141 ± 0.024	0.857 ± 0.010	0.657 ± 0.044	0.697 ± 0.042
Sequence length	32	0.141 ± 0.022	0.864 ± 0.003	0.666 ± 0.023	0.724 ± 0.047
Sequence length	64	0.143 ± 0.025	0.857 ± 0.005	0.650 ± 0.028	0.695 ± 0.025
Hidden size	16	0.160 ± 0.016	0.844 ± 0.012	0.635 ± 0.039	0.692 ± 0.032
Hidden size	32	0.141 ± 0.022	0.864 ± 0.003	0.666 ± 0.023	0.724 ± 0.047
Hidden size	64	0.137 ± 0.023	0.847 ± 0.005	0.657 ± 0.042	0.711 ± 0.040

DOI: 10.54254/2755-2721/2026.37375

Table 3 evaluates the sensitivity of the pre-specified default settings and is not used for test-set-based hyperparameter selection. It compares sequence lengths and hidden sizes of 16, 32, and 64, and the same changes are shown in Figure 3. When the sequence length is 32, the mean ROC-AUC is 0.864 and the F1-score is 0.724, which are the highest values in this comparison. A length of 16 has a similar validation loss but a lower F1-score. Increasing the length to 64 also gives no further improvement. Since the median raw trajectory has 37 points, the pre-specified sequence length of 32 provides a practical balance between temporal coverage and computational efficiency.

The model with 64 hidden units has the lowest validation reconstruction loss. However, it is only 0.004 lower than the loss with 32 units, and its ROC-AUC and F1-score are lower. This is not necessarily contradictory because the validation loss only measures reconstruction on normal trajectories, while anomaly detection depends on the score difference between normal and abnormal trajectories. With more model capacity, the 64-unit model may also reconstruct some abnormal patterns more accurately, which reduces this difference. This can be regarded as overfitting or over-generalization for the detection task even though the normal validation loss becomes lower. The model with 16 units also gives lower detection scores. Thus, the sensitivity results support the robustness of the pre-specified setting of 32 hidden units.

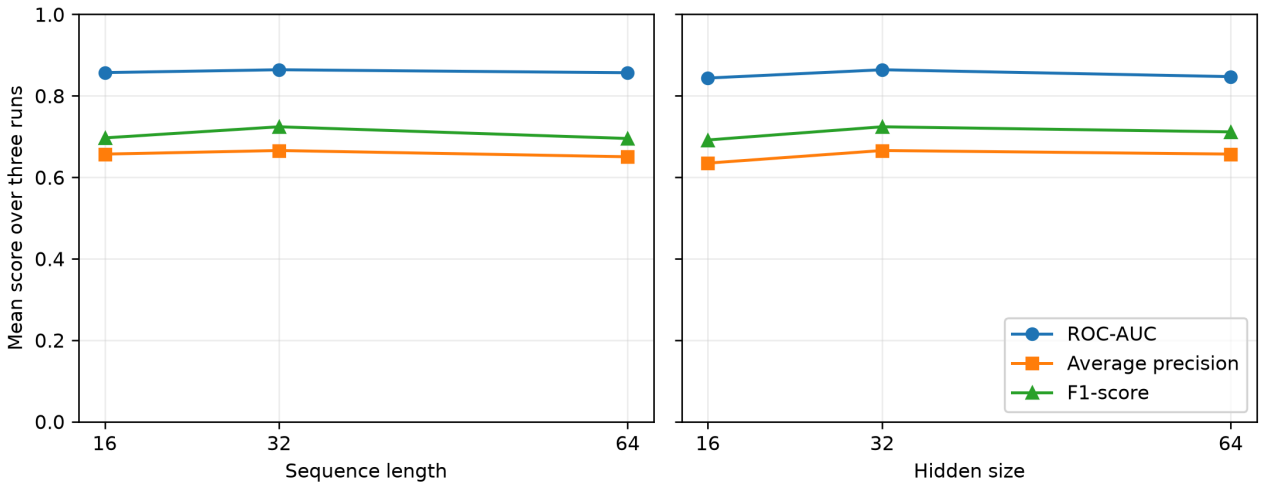

Figure 3. Influence of sequence length and hidden size (picture credit: original)

Table 4. Influence of the validation-score threshold

Percentile	Precision	Recall	F1-score	False positive rate
90th	0.674 ± 0.109	0.750 ± 0.000	0.707 ± 0.057	0.116 ± 0.048
95th	0.730 ± 0.099	0.725 ± 0.037	0.724 ± 0.053	0.086 ± 0.042
99th	0.778 ± 0.070	0.517 ± 0.158	0.614 ± 0.127	0.043 ± 0.011

Table 4 and Figure 4 compare the 90th, 95th, and 99th percentiles. At the 90th percentile, the model detects 75.0% of abnormal trajectories, but the false positive rate reaches 11.6%. The 99th percentile increases precision to 0.778 and reduces the false positive rate to 4.3%. At the same time, its recall falls to 0.517. The 95th percentile has the highest mean F1-score of 0.724, and its precision and recall are close. It was used as the default threshold for the main experiment. For scenarios prioritizing high anomaly coverage, the 90th percentile threshold is a feasible alternative.

DOI: 10.54254/2755-2721/2026.37375

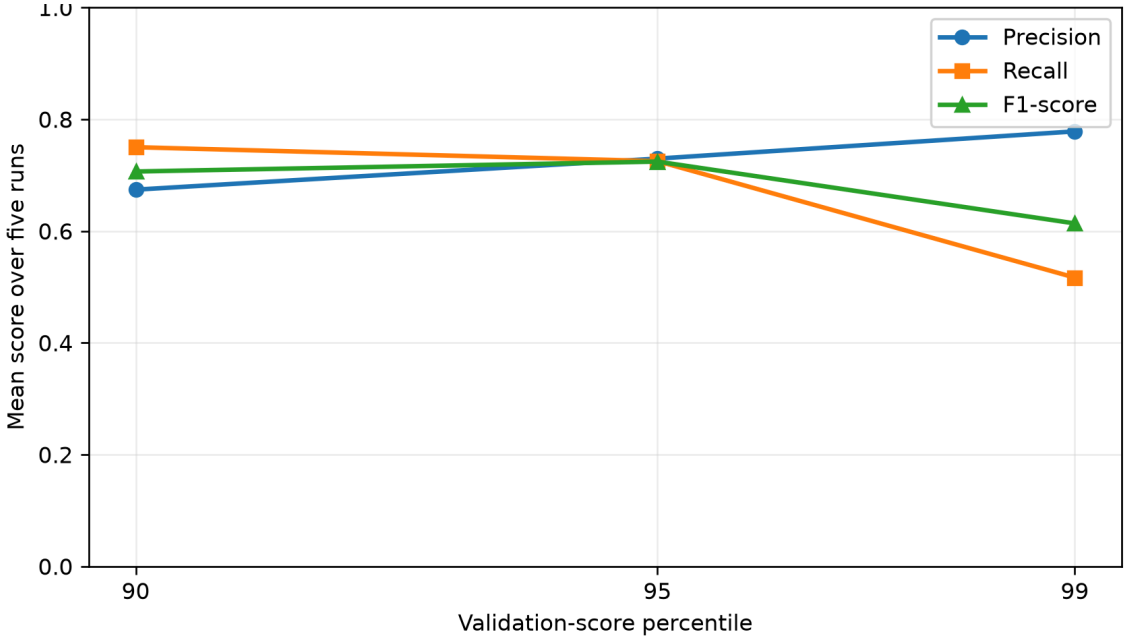

Figure 4. Precision, recall, and F1-score at three thresholds (picture credit: original)

4.4. Training process and limitations

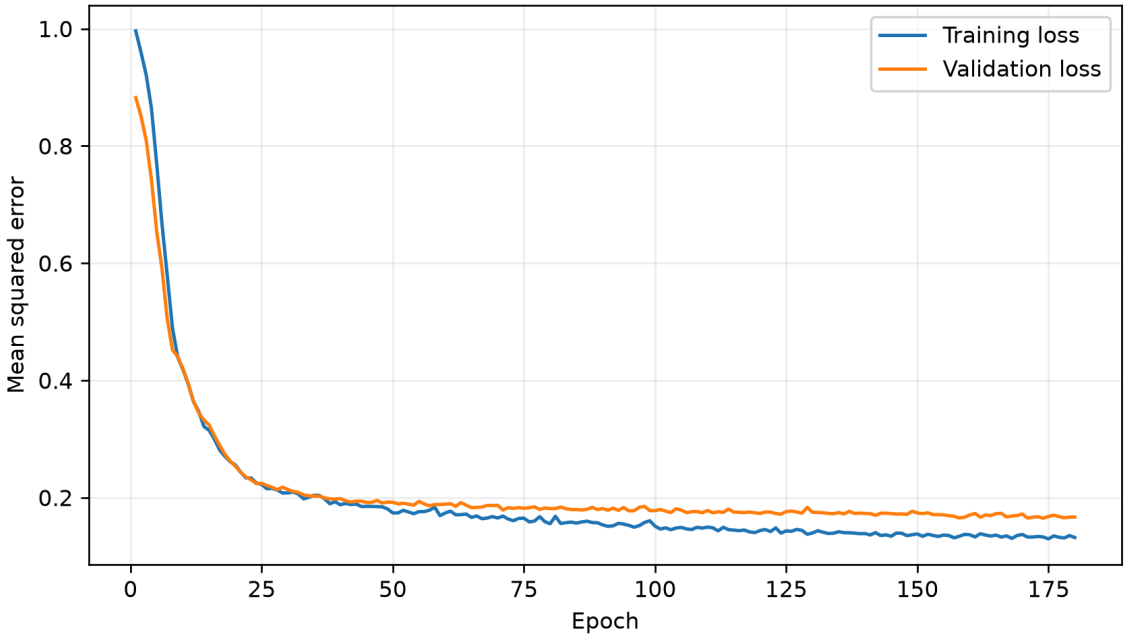

Figure 5. Training and validation loss of the proposed model for seed 42 (picture credit: original)

There are only 24 abnormal trajectories in the dataset. If several of them are classified differently, precision and recall may change noticeably. Using several MMSI splits makes the result more reliable, but it cannot replace a larger labelled dataset. Another problem is that these data come from one day and one sea area. The model has learned the normal movement patterns in this dataset, but it has not been tested with data from another region or season. Accordingly, the reported performance

DOI: 10.54254/2755-2721/2026.37375

is specific to the DTU dataset, and model retuning or threshold recalibration is required for practical deployment in diverse maritime scenarios.

Figure 5 gives the loss curves of the attention LSTM model with seed 42. Both losses decrease quickly during the first 25 epochs. After that, the change becomes smaller. The validation loss stays close to the training loss and does not continue to rise, so serious overfitting is not found in this run. The validation loss still decreases slowly near the end, and the maximum of 180 epochs is reached before early stopping. Dropout and weight decay were still used because there are only 242 training trajectories in the seed 42 split.

5. Conclusion

This study examined an attention-enhanced LSTM autoencoder for vessel trajectory deviation detection using labelled AIS data from Danish waters. After removing trajectories without valid course information, 407 trajectories were used, including 383 normal trajectories and 24 abnormal trajectories. The model learned normal movement patterns from latitude, longitude, speed, course sine, and course cosine, and reconstruction error was used as the anomaly score. Under MMSI-based data splits, the attention LSTM obtained a mean ROC-AUC of 0.859, recall of 0.725, and F1-score of 0.724. Its result was slightly higher than that of the basic LSTM, but the improvement was only about 0.011. The dense autoencoder also performed well and obtained a higher AP. Therefore, the experiment shows that the temporal model is useful for this dataset, but it does not prove that it is better on every metric. In addition, the substantially degraded performance of the position-only baseline verifies that motion features including speed and course provide critical discriminative information for abnormal trajectory identification.

The current study still has several limitations. The dataset contains only 24 abnormal trajectories and mainly covers a single day and specific sea area, so a small change in the detected samples can noticeably affect the results. The selected model and threshold have also not been tested in other regions or seasons. Future work can collect AIS data from multiple time periods, sea areas, vessel types, and anomalous navigation scenarios to test whether the model can generalize. The reason why the dense autoencoder ranks anomalies well should also be studied on a larger dataset. In addition, simpler attention structures and different model capacities can be compared to reduce the risk of excessive reconstruction of abnormal trajectory patterns. Finally, an adaptive threshold and point-level reconstruction error can be studied for online detection, so that an unusual change can be reported before the whole trajectory is completed.

References

[1] United Nations Conference on Trade and Development. (2024). *Review of maritime transport 2024*. United Nations.

[2] International Maritime Organization. (n.d.). AIS transponders. *International Maritime Organization*. https://www.imo.org/en/ourwork/safety/pages/ais.aspx

[3] Pallotta, G., Vespe, M., & Bryan, K. (2013). Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction. *Entropy, 15*(6), 2218–2245.

[4] Ribeiro, C. V., Paes, A., & de Oliveira, D. (2023). AIS-based maritime anomaly traffic detection: A review. *Expert Systems with Applications, 231*, 120561.

[5] Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis* (pp. 4–11).

[6] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

[7] Venskus, J., Treigys, P., & Markevičiūtė, J. (2021). Unsupervised marine vessel trajectory prediction using LSTM network and wild bootstrapping techniques. *Nonlinear Analysis: Modelling and Control, 26*(4), 718–737.

DOI: 10.54254/2755-2721/2026.37375

https://doi.org/10.15388/namc.2021.26.23056

[8] Olesen, K. V., Boubekki, A., Kampffmeyer, M. C., Jenssen, R., Christensen, A. N., Hørlück, S., & Clemmensen, L. H. (2023). A contextually supported abnormality detector for maritime trajectories. *Journal of Marine Science and Engineering, 11*(11), 2085. https://doi.org/10.3390/jmse11112085

[9] Olesen, K., Christensen, A. N., & Clemmensen, L. K. H. (2023). AIS trajectories from Danish waters for abnormal behavior detection [Data set]. Technical University of Denmark. https://doi.org/10.11583/DTU.c.6287841.v1

[10] Zhao, L., & Shi, G. (2019). Maritime anomaly detection using density-based clustering and recurrent neural network. *The Journal of Navigation, 72*(4), 894–916.

[11] Liang, M., Weng, L., Gao, R., Li, Y., & Du, L. (2024). Unsupervised maritime anomaly detection for intelligent situational awareness using AIS data. *Knowledge-Based Systems, 284*, 111313.

[12] Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In *Proceedings of the 2008 IEEE International Conference on Data Mining* (pp. 413–422).

[13] Zhang, B., Hirayama, K., Ren, H., Wang, D., & Li, H. (2023). Ship anomalous behavior detection using clustering and deep recurrent neural network. *Journal of Marine Science and Engineering, 11*(4), 763.

[14] Li, H., Jiao, H., & Yang, Z. (2023). Ship trajectory prediction based on machine learning and deep learning: A systematic review and methods analysis. *Engineering Applications of Artificial Intelligence, 126*, 107062.

[15] Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation, 12*(10), 2451–2471. https://doi.org/10.1162/089976600300015015