Machine Learning with Applications 18 (2024) 100590

Contents lists available at ScienceDirect

Machine Learning with Applications

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

Detecting drug transfers via the drop-off method: A supervised model approach using AIS data

Britt van Leeuwen [a,b,*], Maike Nutzel [b,c]

[a] Centrum Wiskunde en Informatica (Stochastics group), Science Park 123, Amsterdam, Netherlands
[b] Vrije Universiteit Amsterdam, Boelelaan 1111, Amsterdam, Netherlands
[c] Team Maritieme Politie, Rijkszee- en Marinehaven, Den Helder, Netherlands

ARTICLE INFO

Keywords:
Sea-based drug transfers
Automatic Identification System (AIS) data
Long Short-Term Memory (LSTM) model
Drop-off method

ABSTRACT

Maritime security is of tremendous importance in countering drug trafficking, particularly through sea-based routes. In this paper, we address the pressing need for effective detection methods by introducing a novel approach utilizing Automatic Identification System (AIS) data. Our focus lies on detecting the 'drop-off' method, a prevalent technique for contraband smuggling at sea. Unlike existing research, primarily employing unsupervised methods, we propose a supervised model specifically tailored to this illicit activity, with a particular emphasis on its application to fishing vessels.

Our model significantly reduces the number of data points requiring classification by the observer by 70% , thereby enhancing the efficiency of the drop-off detection process. By employing a Long Short-Term Memory (LSTM) model, our approach demonstrates a change from traditional methods and offers advantages in capturing complex temporal patterns inherent in 'drop-off' activities. The rationale behind choosing LSTM lies in its ability to effectively model sequential data, which is essential for detecting drug traffic activities at sea where patterns are subtle and dynamic.

Moreover, this model holds the potential for integration into real-time surveillance systems, thereby enhancing operational capabilities in detecting and preventing drug traffic. The generalizability of our model makes for considerable potential in enhancing maritime security efforts and providing assistance in countering drug traffic on a global scale. Importantly, our model outperforms both baseline models, underscoring its effectiveness and superiority in addressing the specific challenges posed by 'drop-off' detection. For more information and access to the code repository, please visit this link.

1. Introduction

1.1. Motivation

Recently, the BBC published an article about three men who were put under investigation two weeks after their rescue on the coast of Australia (Housden & Guinto, 2023). Suspicions were raised after almost 400 kg of cocaine was found at their rescue location. Although the men claimed to have fallen overboard while fishing, the authorities suspected their story may not hold water. In fact, it is believed that the men were attempting to retrieve cocaine from the ocean and transfer it onto their fishing vessel as part of a drug smuggling operation. This transfer of contraband through the ocean is known as the 'drop-off' or 'pick-up' method. Although this BBC article was focused on an incident along the Australian coastline, this particular method of drug trafficking is observed in various parts of the world. Despite the relatively low number of arrests made in relation to drop-offs in the Netherlands, the Dutch police suspect that the 'drop-off' method is being employed with increasing frequency along the country's coast. This suspicion is supported by the growing number of packages discovered in the ocean or washed ashore (Mehlbaum et al., 2021).

In 2000, the International Maritime Organisation (IMO) adopted a requirement that obliged certain vessels to be equipped with an *Automatic Identification System* (AIS). It concerns passenger ships and all vessels larger than 300 tonnage on international voyage, and larger than 500 tonnage not engaged in international voyage (IMO, 2024). Through AIS, these vessels are obliged to transmit certain dynamic and static information: identity, type, position, course, speed, navigational status, and other safety-related messages. Furthermore, the vessel automatically receives the same data from other vessels on the AIS. The idea behind the implementation of this regulation was to significantly strengthen the safety and security of ships while also facilitating more effective monitoring by the authorities. In 2016, The Dutch

* Corresponding author at: Centrum Wiskunde en Informatica (Stochastics group), Science Park 123, Amsterdam, Netherlands.
E-mail addresses: b.e.van.leeuwen@cwi.nl (B. van Leeuwen), maike.nutzel@politie.nl (M. Nutzel).

https://doi.org/10.1016/j.mlwa.2024.100590
Received 19 December 2023; Received in revised form 23 September 2024; Accepted 6 October 2024
Available online 11 October 2024

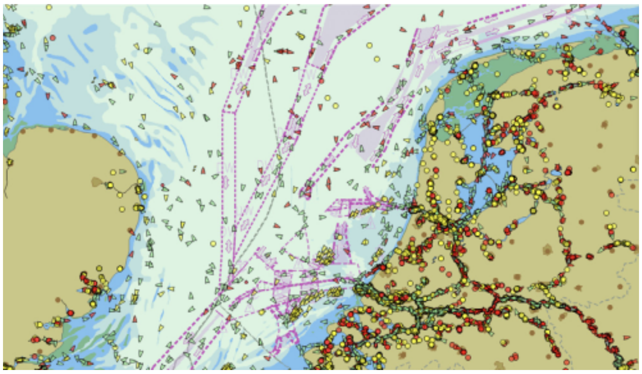

Fig. 1. Snapshot of all vessels equipped with AIS on the coast of The Netherlands (2023). Every arrow and dot represents a vessel.

government further mandated this regulation for all vessels operating commercially and all pleasure crafts longer than 20 meters (Ministerie van Infrastructuur en Waterstaat, 2022). The sea is incredibly busy, 90% of all international transport occurs at sea (Nguyen, Vadaine, Hajduch, Garello, & Fablet, 2019). Most of these transporters are large cargo ships carrying containers, which are obliged to sail within a designated shipping lane. Additionally, many other vessels sail the coast every day. Unlike cargo ships, these vessels are not bound by the shipping lanes and consequently tend to follow more unpredictable and chaotic paths. Therefore, detecting a pick-up with the current methods in place is possible, but challenging. The concentration of vessels is high, especially in regions around coasts (as shown in Fig. 1), and the trajectories are complex.

The suspected number of drop-offs occurring is high and both historical and real-time data are available. Utilizing machine learning for automated surveillance holds significant promise. There have already been worldwide implementations of automated anomaly detection using Recurrent Neural Networks (RNNs) or cluster methods. However, these studies primarily focus on unsupervised approaches, which concern general trajectory anomalies. Research on the detection of anomalies at sea using supervised approaches is very limited, especially concerning the detection of the drop-off method. This is a gap in this field that we are aiming to address with this research. The goal of this research is to develop a machine-learning algorithm that can detect the possibility of a drop-off taking place. The algorithm could be integrated into real-time data systems for future use. It would serve as a trigger to alert Team Maritime Police (TMP) and Coastguard when there is a high likelihood of a drug transfer occurring.

1.2. Background

The type of drug transfer this paper focuses on is the drop-off method, performed at sea. It is important to note that while this method is commonly known as the drop-off method, the method actually consists of two actions: (1) a drop-off of the contraband into the water, and (2) a pick-up of the contraband out of the water. A mother vessel sails through a designated shipping lane outside the coast of The Netherlands. Then at a pre-determined time and location, the mother vessel drops a package of illegal drugs into the water, usually without changing its speed or course. The daughter vessel is often already present in the shipping lane or the area. Following the drop-off, this vessel picks up the contraband with either nets or a hook, thereby executing a pick-up. The organizers of the drop-offs are aware that the daughter vessel's live AIS activities are being monitored. For this reason, the contraband is frequently transferred to at least one other vessel before being brought to land. Generally speaking, to execute a pick-up the daughter vessel must maneuver to the location of the package, decrease speed, and, after performing the pick-up, often makes a U-turn. This behavior differs from vessels occupied with normal activities, such as fishing. Since the mother vessel maintains its course and speed during the drop-off, its involvement is very challenging to trace using AIS data. However, the daughter vessel does make several movements that are anomalous in the AIS data. On top of that, fishing vessels are known to be used for the smuggling of migrants, drug trafficking, weapon trafficking, and acts of terrorism (de Coning, 2011). Fishermen are often recruited for their knowledge of the sea and are generally not the masterminds of the criminal act. They are more susceptible to these recruitments due to financial difficulties caused by the declining fish stocks in many regions of the world (Mehlbaum et al., 2021).

Accordingly, the anomaly detection is focused on the daughter vessel. Since this research is limited to anomaly detection related to the daughter vessel, the 'pick-up' action rather than the 'drop-off' action is of relevance. When discussing the method, we will use the term 'drop-off', and when referring to the anomaly, we will refer to 'pick-up'.

1.3. Related work

In the context of all types of maritime anomaly detection, there exists a wide variety of classification and prediction methods. In 2022, Wolsing et al. survey on anomaly detection of AIS tracks (Wolsing, Roepert, Bauer, & Wehrle, 2022). To categorize the existing methods, they adopted the same classification as described by Lane, Nevell, Hayward, and Beaney (2010): 'route deviation', 'unexpected activity', 'port arrival', 'close approach', and 'zone entry'. From these five categories, the classification of potential pick-ups falls under the 'route deviation' category, although 'close approach' and 'unexpected activity' are also closely related. Accordingly, the remainder of this section will focus on the existing research related to these three categories.

The vast majority of papers researching anomaly detection with AIS data focuses on deviation anomalies (Yang, Liu, Li, Zhang, & Liu, 2024). Venscus et al. introduced a method using a Long Short Term Memory network (LSTM) with bootstrapping to predict route deviation anomalies (Venskus, Treigys, & Markevičiūtė, 2021) in an unsupervised manner. The resulting model accurately detected abnormal movements such as 'vessel slowdown', 'turn around', 'sharp direction change', and 'unplanned stopping'. It is noted that this works for predictable paths such as cargo vessels, but that it is much more challenging with fishing vessels' complex trajectories. However, it is anticipated that this challenge concerning complex trajectories manifest in other deep learning models too. Therefore, LSTM remains a promising choice for anomaly detection in maritime environments. Its ability to model sequential data and capture temporal dependencies makes it well-suited for identifying subtle deviations in vessel behaviors over time. LSTM's capability to learn long-term dependencies enables it to discern complex patterns inherent in maritime activities.

Nguyen et al. (2019) developed GeoTrackNet, an anomaly detection model in route deviations. It employs probabilistic notation of tracks with a Variational Recurrent Neural Network trained for anomaly detection. Unlike previous models that try to detect route deviations, GeoTrackNet can be applied to any type of vessel, including fishing vessels. The authors do note that their patterns are the most complex of all vessels and thereby pose the greatest challenge for the model to learn their deviations. It is interesting to note their novel approach to feature representation in the use of a four-hot encoding vector as input to the neural network, albeit with resource-intensive computational requirements. The four-hot encoding is a concatenation of four one-hot-encoded vectors of the ship's latitude, longitude, Speed Over Ground (SOG), and Course Over Ground (COG). Continuous values are discretized into bins, and each bin is represented by a one-hot vector. This is argued to be more appropriate than using the conventional numerical input for a neural network, since the encoding better represents the geo-spatial meaning of these variables (Nguyen et al., 2019).

An interesting research under the 'unexpected activity' anomalies is by Singh and Heymann (2020). They propose a neural network framework to detect intentional on-and-off switching of the AIS transmitter. It uses the numerical values of latitude, longitude, SOG, and COG as input for their model. The authors test their model by comparison to known power outages and reach a 99,99% accuracy in detecting whereas the AIS that is transmitted is normal, intentionally turned off, or is due to a power outage, only missing a few instances. This highlights the significance of considering various types of unexpected activities beyond traditional anomalies, which aligns with our research objective of developing a comprehensive approach for detecting sea-based drug transfers, including behaviors such as intentional AIS manipulation.

Within the 'close approach' category, the identification of collision of vessels was researched by Liu, Wang, Cai, Liu, and Liu (2020). In this research, an RNN is implemented to predict collisions of two vessels at sea based on interpolating geographical information. While the research shows promising results, it is suggested to use other variables such as SOG and COG in further research. Fang, Yu, Ke, Shaw, and Peng (2019) do consider SOG and COG as variables in addition to longitude and latitude and approach the classification with spatial indexing. A grid is defined on the spatial resolution and a close approach is determined by checking pairs of vessels in the same or adjacent squares. The results of the research are compared to official precautionary areas, revealing discrepancies in certain areas while showcasing favorable outcomes in areas pertinent to the task of detecting maritime based drug-traffic. Notably, the selection of feature representation yielded positive results, further validating its efficacy for our objectives.

Most research done in anomaly detection for fishing vessels is regarding illegal fishing activities. While this is not the goal of this research, these models are built with the same trajectory data and so address the issue of complex trajectories. Arasteh et al. (2020) introduced FishNet, a Convolutional Neural Network tailored for classifying fishing trajectories. While FishNet achieves a commendable F-1 score of 92.35%, surpassing the 90% score of the highest baseline, its inability to distinguish anomalous events from routine activities, such as port arrivals or specific route deviations, highlights the need for further advancements in anomaly detection algorithms.

Recently do Nascimento, Alves, de Farias, and Dutra (2024) proposed an approach that combines navigation behavior models with expert rules to detect illegal maritime activities. This model integrates AIS data and expert knowledge by using active learning. This reduces the required manual work as the methods are focused on detecting illegal fishing and suspicious behavior. However, it relies heavily on expert rules and labeled data, limiting adaptability to new or evolving threats. Additionally, it does not differentiate between various types of suspicious behavior, making intervening by the maritime authorities more challenging.

These related works are summarized in Table 1.

Table 1
Literature survey in tabular form.

Category	Paper	Keywords
Review paper	Yang et al. (2024)	Machine Learning Using AIS
Categories of classification	Wolsing et al. (2022)	Survey of anomalies Using AIS
	Lane et al. (2010)	Route deviation, Unexpected activity, Port arrival, Close approach, Zone entry
Route deviation	Venskus et al. (2021)	LSTM, unsupervised
	Nguyen et al. (2019)	GeoTrackNet, Any vessel
Unexpected activity	Singh and Heymann (2020)	On-and-off switching of AIS
Close approach	Liu et al. (2020)	Collisions, RNN
	Fang et al. (2019)	Variable choice: SOG, COG
Fishing Vessels	Arasteh et al. (2020)	Illegal fishing, Anomaly detection, FishNet
	do Nascimento et al. (2024)	Illegal fishing, Suspicious behavior, Active Learning

1.4. Our proposal

As of the time of composing this paper, no prior research on supervised 'drop-off' method detection has been identified by the authors. Our research differs from previous research in multiple ways. Of the existing research on illegal activities at sea, most focus on illegal fishing. This concerns mainly forbidden navigational maneuvers. This makes illegal fishing an issue of trajectory, whereas the drop-off method is an issue of behavioral patterns (including, but not limited to trajectory patterns). Other research focuses on anomaly detection, taking the entirety of illegal activities into account. Due to the lack of labeled data, stemming from confidentiality concerns, the models proposed for this are unsupervised. However, unsupervised models must rely on identifying deviations from expected patterns, which can be complex and dynamic in maritime environments. Moreover, the lack of ground truth labels makes it difficult to assess the performance of these models accurately. This underscores the need for innovative strategies to address the unique intricacies of detecting illicit activities at sea. This research aims to bridge this gap. This research is novel for several reasons. Firstly, unlike much of the prior work which uses unsupervised methods, our model utilizes labeled data, improving the precision of anomaly detection. Secondly, the study introduces a focus on the specific 'drop-off' method used in maritime based drug traffic, which has not been sufficiently explored in existing research. Thirdly, the model is designed to be adaptable to different maritime regions, making it versatile for different geographic contexts. Lastly, by utilizing a sliding window approach, our model is optimized for real-time anomaly detection. Our approach is designed to detect suspicious behavior at an early stage, enabling maritime authorities to intervene in a timely manner and address emerging threats more effectively.

A significant challenge in this study is dealing with imbalanced classification. In the context of maritime drug trafficking, instances of illegal activities are rare compared to the vast amount of legitimate maritime traffic, leading to a highly imbalanced dataset. To address this, we have employed techniques such as class weighting and specialized metrics that focus on performance for the minority class.

Furthermore, the nonstationary nature of maritime data adds another layer of complexity. Maritime environments are subject to dynamic changes due to varying traffic patterns, environmental conditions, and regulatory measures over time. By choosing the sliding window approach with an LSTM model, we ensure that the model is always focusing on the most recent and relevant data, helping to address challenges posed by nonstationarity.

Building upon prior work in anomaly detection for fishing vessels, our research leverages advancements in deep learning techniques to develop a model specifically tailored for detecting maritime based drug transfers. We have determined a feature selection and model refinement based on insights gained from existing literature.

2. Data

2.1. Automatic identification system

Every Fishing vessel (longer than 15 m) is obliged to transmit an Automatic Identification System, AIS for short. These transmits consist

Table 2
All the AIS variables, together with their type and numerical range.

Name	Type	Description
Date and UTC Time	Datetime	The date and UTC time of the transmission.
MMSI	Integer	Unique 9 digit identifier number for radio station(s)
Status	Integer $\in[1, 15]$	Status of the vessel by manual input, e.g., 'Fishing'.
Latitude (°)	Float $\in[-90, 90]$	Position coordinate
Longitude (°)	Float $\in[-180, 180]$	Position coordinate
SOG (knots)	Float	Speed over Ground, the speed relative to the ground.
HDG (°)	Float $\in[0, 365]$	Heading, the direction in which the vessel is pointing
COG (°)	Float $\in[0, 365]$	Course over Ground, the direction which the vessel is heading
ROT ($°/min$)	Float $\in[-720, 720]$	Rate of Turn, the position of the steering wheel
IMO number	String	Unique 7 digit number for propelled sea going merchant ships of more than 100 Gross Tonnage
Name	String	Name of the vessel
Call Sign	String	Unique alphanumeric vessel identity
Length Bow (m)	Integer	Vessel measurement
Length Stern (m)	Integer	Vessel measurement
Length Overall (m)	Integer	Vessel measurement
Width Port (m)	Integer	Vessel measurement
Width Starboard (m)	Integer	Vessel measurement
Width Overall (m)	Integer	Vessel measurement
Draught (m)	Integer	Vessel measurement
Destination	String	The intended destination
Vessel Type	String	e.g., 'Vessel'
Extra info	String	e.g., 'Fishing'

of static and dynamic information. The static information has been manually entered by the vessel's crew and consists of information such as the Maritime Mobile Service Identity (MMSI) number of the ship, its name, its status, dimensions of the boat, destination, and vessel type. The dynamic information is transmitted automatically and consists of the date and time, the vessel's SOG, its location represented in longitude and latitude coordinates, its COG, its Heading (HDG), and Rate of Turn (ROT). The variables, together with their type and numerical range can be observed in Table 2. Unlike the other variables in the table, the range for the SOG is not fixed. AIS data often includes unrealistically high SOG values (Eriksen, Greidanus, & Delaney, 2018). To filter out this noise from the dataset during pre-processing of the data, a threshold is used based on research by Greig, Hines, Cope, and Liu (2020). Hence, the truncation threshold for fishing vessels is set at 15 knots.

Although AIS data is widely accessible, it has its limitations. The primary issue associated with using AIS data is that it can be unreliable. AIS sends out signals every 2 – 12 seconds, depending on the ship's SOG as well as gaps in the reception (IMO, 1998). This is usually a result of saturation of the system due to a high concentration of vessels in an area or insufficient quality of the transmissions caused by the equipment of the receiver and/or the transmitter (Ford, Peel, Kroodsma, Hardesty, Rosebrock, & Wilcox, 2018; Wolsing et al., 2022). As a result, the trajectories often contain significant gaps.

Besides this, the missing information can also be caused by the fishing vessel's crew, not cooperating with the AIS system. It is possible that a fishing vessel's captain has, intentionally or not, never activated its AIS system or incorrectly entered static information. Furthermore, the AIS transmitter can be turned off. This can be related with the goal of (drug) trafficking or more innocent crimes such as the prevention of disclosing a good fishing spot to other fishers in the area (Ford et al., 2018; Wolsing et al., 2022). However, this does generally get easily noticed by the authorities.

2.2. Data set

2.2.1. Data extraction

The dataset used for this research was sourced from MadeSmart, a trusted repository for historical AIS data (MadeSmartGroup, 2014). We define data points as segments of a vessel's route. Positive data points represent segments that were part of a route undertaken by a vessel involved in drug trafficking during that specific journey. Specifically, all segments of tracks where ships were performing a pick-up at any point during that track are included. Therefore, segments are also labeled positive if there was no pick-up during that specific segment, but there was a pick-up during the entire track. This approach is used because we aim to detect suspicious ships before the drop-off occurs, to be able to intervene on time. Negative data points represent any other route segments in the data. The negative data points were obtained by downloading all the tracks available in 2022, with two filters. The first filter required the tracks to have occurred in the polygon shown in Fig. 2. This polygon was determined in collaboration with TMP. Additionally, the polygon encompasses a large shipping lane which cargo ships navigate through. Therefore, a majority of the pick-ups are expected to occur in this lane. The second filter is that on ship type. Only fishing vessels are exported. Since the vessel type in the AIS system relies on manual input from the skipper, it is susceptible to human error. Nonetheless, the filter was necessary to effectively restrict the amount of downloaded data. The resulting negative dataset consists of AIS information from almost 40 million tracks. To ensure our model captures the full range of maritime activity, this dataset spans an entire year. Thus, different seasonal patterns, varying weather conditions, and fluctuating sea states are taken into account. These temporal variations allow the model to account for a wide range of behaviors, making it more robust in real-world scenarios where nonstationary factors such as weather and tides may influence ship movements.

The positive dataset is built on 18 tracks of known pick-ups. The tracks cover the estimated time of suspicious activity, spanning from 3 h to 4 days. The whole positive dataset is much smaller, consisting

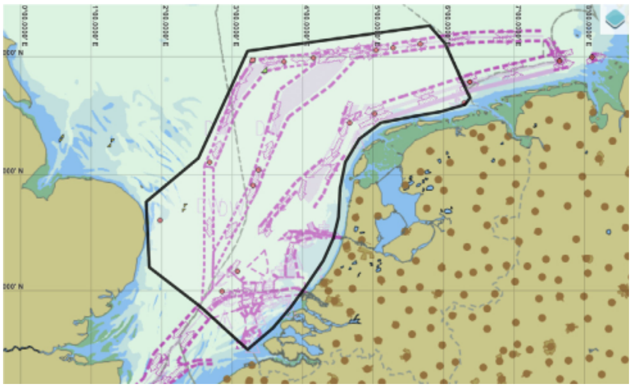

Fig. 2. The polygon filter on downloaded data for the negative dataset: an area with significant traffic activity.

of nearly 200 thousand data points. The majority of the tracks occurred within the past three years, with the earliest one dating back to 2013. The positive dataset, representing segments of routes associated with drug trafficking activities, was meticulously curated based on direct evidence from TMP investigations or compelling suspicions from expert analysis. This manual labeling process ensured that positive instances were authentic representations of confirmed or highly probable illicit activities. As a result, the quantity of positive data points is relatively limited. Using the TMP's knowledge of known pick-ups, the data from that day was extracted from MadeSmart.

2.2.2. *Class imbalance*

Due to the small availability of the positive dataset, the dataset suffers from extreme class imbalance. The minority class makes up 0.5% of the total dataset. There is no statistics on the actual percentage of tracks that represent pick-ups per year. It can, however, be assumed that a class imbalance is realistic. Only a small amount of the traffic at sea is en route to a pick-up. To enhance generalizability, the training data should reflect the real world and this imbalance must remain (van den Goorbergh, van Smeden, Timmerman, & Calster, 2022). This is crucial for applications where the imbalance itself is a natural and significant feature of the problem, such as maritime traffic, where pick-up events are rare. It is challenging to pinpoint the exact threshold at which the minority class becomes too under-represented. Nonetheless, previous research suggests that, in general, when the minority class constitutes less than 1% of the dataset, it leads to significant class imbalance issues and can result in poor model performance (Krawczyk, 2016; Sun, Wong, & Kamel, 2011). We deal with the extreme dataset imbalance by using data augmentation, where the negative class is undersampled resulting in a minority set that makes up 1% of the complete dataset. We opt for undersampling rather than oversampling methods like SMOTE (Chawla, Bowyer, Hall, & Kegelmeyer, 2002), since it generates synthetic samples, which can sometimes lead to less realistic data points, particularly if the minority class has complex patterns like those of the fishing vessels. Moreover, using undersampling makes the model less prone to overfitting than oversampling and it is generally simpler to implement.

2.3. *Dataset preprocessing*

For the proposed model, the dynamic variables *SOG*, *COG*, *Latitude*, *Longitude*, as well as the static variables *Date and UTC Time*, and *MMSI* are considered. The other AIS variables are removed from the dataset. During pre-processing some general errors should be addressed (Yang et al., 2022). This includes incorrect MMSI lengths, duplicate data points, and abnormal latitude, longitude, SOG, and COG values. Possible errors such as large gaps in the reception of latitude and longitude

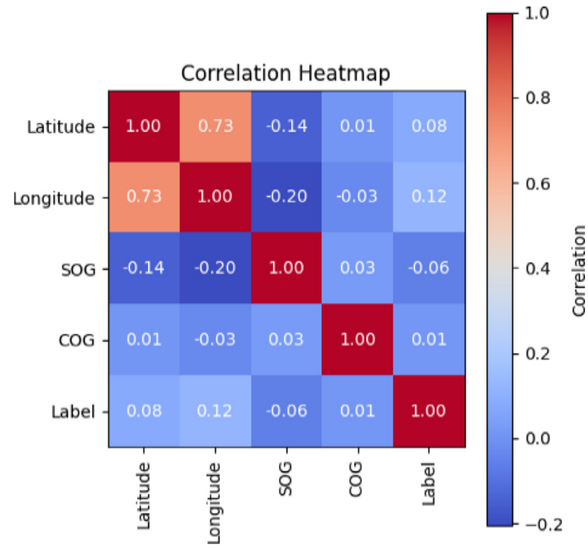

Fig. 3. Pearson correlation coefficients between variables.

data are not to be removed from the dataset. The large gaps in the data could be an indicator of intentional tampering with the material. By removing these gaps, important information might get lost. The model must focus on finding anomalies in the trajectories. To achieve this, the latitude and longitude values undergo normalization to not contain positional information. This normalization process involves converting latitude and longitude to relative values by subtracting each coordinate from the initial coordinates of the track.

Summarized, the input data for the algorithm comprises normalized dynamic variables such as SOG and COG, as well as static variables including date, time, and MMSI, derived from historical AIS data. These variables represent segments of vessel routes, with positive instances indicating segments associated with drug trafficking activities and negative instances representing other route segments.

2.4. *Data properties*

The correlation heatmap provided in Fig. 3 shows the Pearson correlation coefficients between different variables. Based on this heatmap, we can infer certain properties of the dataset.

Latitude and Longitude have a strong positive correlation of 0.73, which is expected since they represent spatial coordinates and are often correlated.

Latitude and SOG have a weak negative correlation of −0.14, suggesting a slight tendency for the speed to decrease as latitude increases. Moreover, Longitude and Speed Over Ground have a weak positive correlation of 0.2, indicating a slight tendency for the speed to increase as longitude increases. This is probably due to the navigational routes in the specific area we are investigating as shown in Fig. 2. Although, looking at factors such as sea currents and wind patterns could also explain this.

The weak correlations between the label variable and other variables suggest that there may not be strong linear relationships between the features and the target variable. This underscores the need for more sophisticated models like LSTM, capable of capturing complex, nonlinear patterns in sequential data.

Sailing duration varies significantly for each fishing vessel. This leads to great disparity in the number of data points for each track. Fig. 4 shows the minimum, first quartile, median, third quartile and maximum of the distribution on number of data points for both labels, suspicious and not suspicious. From this, it is clear that the dataset contains tracks with an extreme amount of data points, ranging from

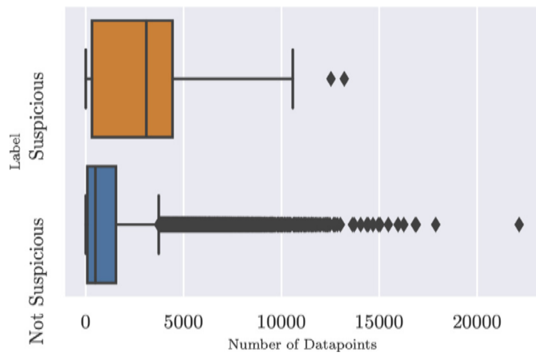

Fig. 4. Distribution of track length for both labels.

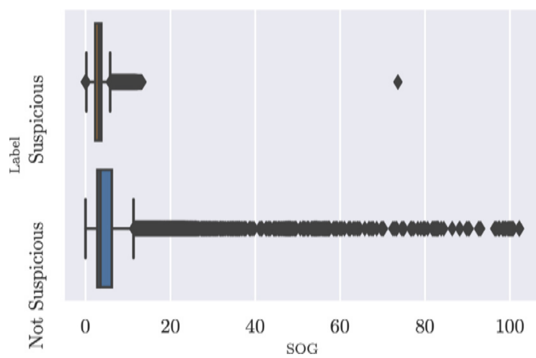

Fig. 5. Distribution of the SOG.

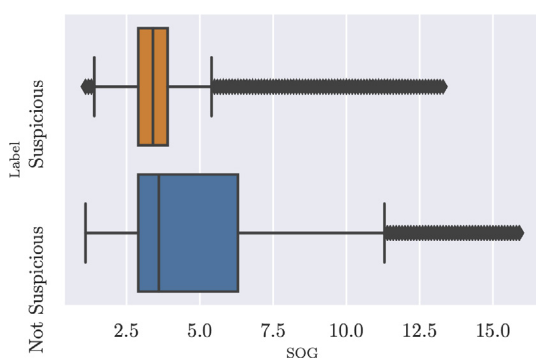

Fig. 6. Filtered distribution of the SOG.

tracks with 1 data point to tracks with 20,000 data points. In pre-processing all tracks are truncated with the sliding window approach. As a result, none of these extreme points should not be removed.

The boxplot in Fig. 5 summarizes the distribution of the SOG for both the positive and negative datasets. We can conclude that, as found in earlier research by Eriksen et al. (2018), Greig et al. (2020), the SOG suffers from many erroneous outliers. The truncated SOG of a fishing vessel is 15 knots; therefore, outliers above this value must be removed during pre-processing. In order to get a better idea of the distribution of the actual moving speed, the data is trimmed for speeds higher than 1 knot and lower than 16 knots. The boxplot for this trimmed version can be observed in Fig. 6. This results in a realistic median moving speed between 3 and 4 knots for both the positive and negative datasets. The upper quartile and maximum values are higher in the negative dataset compared to the positive dataset. This could be attributed to the limited number of tracks available in the positive dataset, which reduces the likelihood of including a fast vessel.

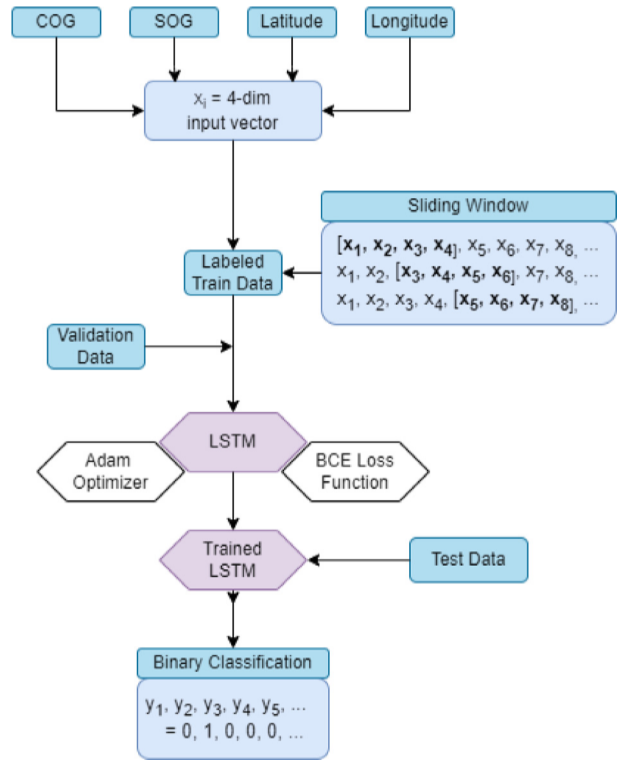

Fig. 7. Flowchart of proposed method.

3. Model description

The proposed method is illustrated as a flowchart in Fig. 7 at the end of this section, after the discussion of its parts.

3.1. Proposed model: LSTM

The selected model for this research is a (stacked) LSTM model (Hochreiter & Schmidhuber, 1997). LSTMs are a specialized form of RNNs (Yu, Si, Hu, & Zhang, 2019). It is an extension of traditional RNNs, aiming to address the vanishing gradient problem. The vanishing gradient problem arises when training on long sequences, as the gradients approach zero over time. It prohibits the model's ability to capture long-term dependencies. The LSTM model aims to improve the long-term dependencies by the incorporation of a memory cell. This enables the model to selectively retain or discard information over long sequences. This capability makes LSTM especially useful in (spatial) time-series analysis (Karim, Majumdar, Darabi, & Chen, 2018). The input to an LSTM model is commonly a sequence of data, where each element in the sequence represents a time step. In the classification of dynamic AIS data, the sequence is (a segment of a) trajectory and the element AIS data at a certain point in time. In summary, the selection of LSTM is justified by its ability to effectively model sequential data, which is essential for detecting drug traffic activities at sea where patterns are subtle and dynamic.

The model's architecture was experimented with both single-layer and two-layer configurations. The choice of configuration depends on the complexity of the task. For more complex tasks that require capturing long-term dependencies or intricate patterns in the input sequence, a two-layer LSTM model could be beneficial. The additional layer allows the model to learn more abstract representations and capture higher-level features. This increased depth can enhance the model's capacity to handle complex sequences. If the task is not complex and a two-layer model is used, overfitting is likely to happen. Since the complexity of the data is difficult to determine, the optimal

Table 3
Hyperparameters LSTM model.

Hyperparameter	Parameter Space
Number of Layers	$\sim dU(1, 2)$
Number of units layer 1	$\sim dU(32, 256)$
Number of units layer 2	$\sim dU(32, 256)$
Dropout	$\sim U(0.5, 0.65)$
Learning Rate	$\sim U(0.0001, 0.01)$
Batch Size	$\sim dU(2048, 4096)$

Table 4
Parameters LSTM model. The parameters with an asterisk (*) were found with hyperparameter tuning, using the hyperopt package.

Parameter	Chosen Value
Number of Layers*	1
Number of units in layer 1*	154
Dropout*	0.64
Learning Rate*	0.002
Batch Size*	2048
Epochs	300
Window size	16
Overlap between windows	50%

configuration is found using hyperparameter tuning. This was done with Hyperopt (Bergstra, Yamins, & Cox, 2013), which explores the hyperparameter search space by making use of Sequential Model-based Optimization, a Bayesian optimization technique. As the method assesses the quality of the hyperparameters using an objective function, it is crucial to select an appropriate objective function like the AUC-PR.

Table 3 contains the tuned input for Hyperopt. The $U(a, b)$ indicates the uniform distribution and $dU(a, b)$ the discrete (integer) uniform distribution between the values of a and b. The parameters found through tuning can be observed in Table 4.

3.2. Sliding window

The classification model will be trained and tested on historic AIS data. This entails that the LSTM model is trained on previously completed trajectories. However, in further development, the model is meant to be employed as a live classification model. In that case, the complete trajectories are not yet known and thus cannot be used as input for the model. As a solution, a sliding window will determine the input sequence. A sliding window involves moving a fixed-size window over every nth received AIS message.

3.3. Adam optimization

Adam regularization (Kingma & Ba, 2017) is implemented as an optimization algorithm. Adam is chosen as the preferred optimizer due to its speed and early convergence capabilities. The β_0 and β_1, ϵ, and decay are kept as is standard for Keras (Chollet et al., 2015) 0.9, 0.99, 10^{-8}, and 0 consecutively. The learning rate is optimized with hyperparameter tuning.

3.4. Loss function

In our research, we employed the standard binary cross-entropy (BCE) loss function for model training. This decision was driven by BCE's well-established effectiveness in binary classification tasks, which aligns with our objective of distinguishing between two classes. BCE's simplicity and robustness make it a preferred choice, ensuring stable convergence during training. Additionally, its capacity to handle imbalanced datasets, by adjusting class weights, provides a flexible and reliable framework for optimizing our model's performance. Hence, leveraging BCE loss facilitated efficient and precise training, directly contributing to the overall accuracy and reliability of our results. The BCE function is given in Eq. (1). N is the number of data points, y is the binary label (1 for suspicious or 0 for normal behavior), and $p(y_i)$ is the predicted probability of the label of point i being suspicious, or belonging to the positive class, 1.

$$H_p = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \tag{1}$$

3.5. Performance measure

The output of the LSTM model is a sequence of predictions, where each prediction corresponds to a time step in the input trajectory, indicating the likelihood of a given segment being associated with drug trafficking activities.

Given the significantly skewed distribution of the data, accuracy alone can be misleading, making recall and precision more informative metrics. For our implementation, prioritizing recall over precision is preferred to ensure that potentially suspicious tracks are not overlooked, aligning with the system's preference for false alarms over missed detections. Consequently, the Area Under the Precision–Recall Curve (AUC-PR) serves as an optimal objective function for our research, as it can be tailored to weigh in favor of either recall or precision. Additionally, other metrics such as Balanced Accuracy and $F - \beta$ Score are selected to strike a balance between detecting as many pick-ups as possible while minimizing false alarms. By choosing the β to be 3, we aimed to enhance the model's sensitivity to the positive class, which aligns better with our goal of detecting suspicious ships before the drop-off takes place, allowing for timely intervention.

In order to measure overall performance, the AUC-PR is computed during training and after testing. AUC-PR traditionally evaluates the performance of a soft classifier, which is a classifier that predicts probabilities instead of discrete classes. However, by setting a threshold, it can be used as a measure for hard classifiers. The threshold is set at 0.35 to reflect the specific trade-offs we need to manage in this classification task. This lower threshold is chosen because of the high class imbalance in our dataset, where the positive class (e.g., suspicious ships) is much rarer than the negative class. By setting the threshold at 0.35, we aim to increase recall, while managing precision.

The Balanced Accuracy and F$-\beta$ are also computed for the test set. Balanced accuracy is an alternative to the classic accuracy measure which takes the arithmetic mean of sensitivity and specificity. By doing so, it aims to balance out the impact of class imbalance. F$-\beta$ takes the harmonic mean of recall and precision with parameter β. A higher β increases the weight of the recall as opposed to precision.

3.6. Baseline

To evaluate our model using a performance score, a baseline must be established. The objective of this research is not to enhance an existing state-of-the-art model but to introduce a model for detecting pick-ups. To the best of the authors' knowledge, comparing this model to any state-of-the-art model would be improper, due to the inherent differences between them. Due to this unavailability, we have chosen two different baselines.

Firstly, we employ the Dutch Draw (DD) (van de Bijl et al., 2022) as our first baseline. This research by Van de Bijl et al. proposes a universal baseline for binary classification models (van de Bijl et al., 2022). They found a way to determine a baseline that facilitates cross-paper comparisons.

Secondly, we employ a baseline by training a Logistic Regression (LR) model with balances weights. Logistic Regression is a simple yet powerful linear model commonly used for binary classification tasks. By employing LR as a baseline, we establish a straightforward and interpretable benchmark against which we can compare the performance of our more complex LSTM model. Balancing the class weights in the LR model addresses the issue of class imbalance present in the dataset.

4. Experimental design

The trajectories are split up with the sliding window approach. Initial hyperparameter tuning through grid search found the ideal length to be 16 and the ideal overlap to be 50%. Since the sliding window is of size 16, all tracks with fewer than 16 data points are removed. This ensures that the model receives sufficiently detailed input data for analysis and is not subjected to underfitting. The dataset is shuffled and split into 60% train set, 20% validation, and 20% test set. A larger proportion of the data is allocated to validation and testing than usual to ensure that these sets contain enough positive data points, given their rarity. Each set is ensured to have a sufficient number of positive data points. The variables $Latitude$, $Longitude$, SOG, and COG are put into a four-dimensional vector, to be used as input of the LSTM.

The class weights of the BCE loss function are computed with a function from the Scikit-learn package (Pedregosa et al., 2011). With the addition of these weights, the loss function is weighted such that samples from each class carry equal weight. To prevent overfitting, Early Stopping has been implemented. Early Stopping monitors the model's performance using the validation set. If the AUC-PR validation score has stagnated at a maximum, the model stops training. It is implemented with a patience of 10 epochs.

The final parameters for the model can be observed in Table 4. The grid search was conducted with a maximum of 15 evaluations for each hyper-parameter configuration, ensuring a robust search for the optimal parameters while considering computational resource limitations. The holdout validation method was used in this process: the dataset was split into training and validation sets with a 70–30 ratio. This approach allows us to train the model on a substantial portion of the data while using the remaining portion to validate the model's performance. This balance helps in effectively tuning the hyper-parameters and assessing the model's ability to generalize to new data.

Fig. 8. Training Loss over 500 Epochs.

Fig. 9. AUC-PR Score During Training over 500 Epochs.

5. Results

5.1. BCE loss

After 500 epochs, the model achieves 0.0091 loss on the training set and 0.0612 on the validation set. Fig. 8 depicts the progression of the BCE over all the epochs. The training and validation loss curves reach a stable and nearly flat state at approximately epoch 200. During the epochs that follow, there is a minimal difference in the loss values, indicating that the model learns very little additional information. From that point on, the model is more inclined to overfit if trained longer. The AUC-PR was chosen as the metric for early stopping, thereby decreasing the chance of over-fitting by ensuring that if the validation AUC-PR performance stagnates, the model's learning process comes to a halt.

The plot displaying the validation loss consistently remains slightly higher than the training loss without intersecting. This pattern indicates a well-fit model. Since there is no large gap between validation and training loss, the training set does not seem underrepresented. From the graph, it can be seen that both the validation and training loss oscillate while converging. This is to be expected due to mini-batch gradient descent, a method that divides the training data set into small batches to calculate model error and update model coefficients (Brownlee, 2019).

Table 5
Performance LSTM model.

Metric	Training	Testing
Recall (TPR)	0.9993	0.9703
Precision (PPV)	0.7391	0.7097
AUC-PR	0.7107	0.7096
Balanced Accuracy	–	0.9832
F-3	–	0.9360

Table 6
Confusion matrix after training.

		Predicted	
		Yes	No
Real	Yes	9578	3381
	No	7	956 431

5.2. Performance measures

Table 5 shows the final values of the performance measures after training and testing. Tables 6 and 7 display the confusion matrices after training and testing. A final recall score of 97% is high. It represents the model only missing out on 3% of drop-offs. The final precision score is 71%. It is significantly lower than the recall. This is expected due to the weight in the loss function favoring recall over precision. A precision of 71% indicates that the model classifies 29% movements as pick-ups when the trajectory of the vessel was in reality not normal.

After 500 epochs, the model achieves a ROC-PR score of 71% on the training set and 67% on the test set. Fig. 9 illustrates the progression of the ROC-PR curve during training. From this curve, it can be seen that the model does not seem to over- or under-fit on the data. The training and validation scores increase but reach a plateau near the end of the epochs. After testing, the balanced accuracy scores 98% and the F-3 scores 94%.

Table 7
Confusion matrix after testing.

		Predicted	
		Yes	No
Real	Yes	8291	3392
	No	253	849 751

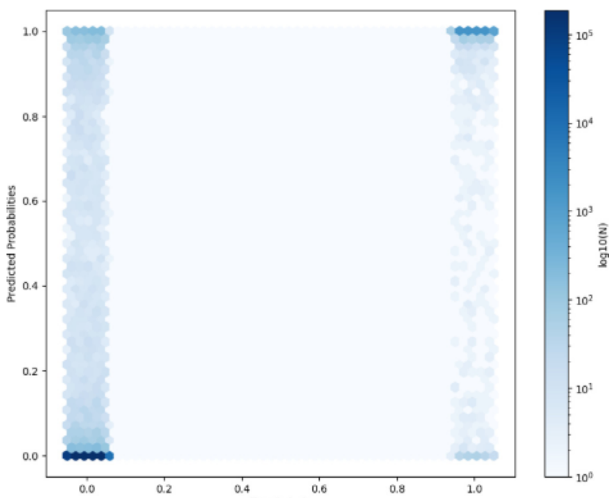

Fig. 10. Scatterplot comparing the true labels against the predicted probabilities generated by the LSTM model.

Table 8
Results with Dutch Draw scores, Logistic Regression scores and Proposed model scores.

	Dutch Draw		Log Reg	Model
	argmin{$\mathbb{E}$}	argmax{$\mathbb{E}$}		
Recall	–	–	0.819	0.9704
Specificity	–	–	0.831	0.9960
Miss Rate	–	–	0.181	0.0296
Fall Out	–	–	0.168	0.0040
Precision	0.0099	0.0099	0.048	0.7097
NPV	0.9901	0.9901	0.997	0.9997
FDR	0.9901	0.9901	0.951	0.2903
FOR	0.0099	0.0099	0.002	0.0003
f_3	0.0910	0.0000	0.409	0.9360
J	0.0000	0.0000	0.047	0.9664
Markedness	0.0000	0.0000	0.045	0.7094
Accuracy	0.9901	0.0099	0.826	0.9958
Balanced Accuracy	0.5000	0.5000	0.825	0.9832
MCC	0.0000	0.0000	0.045	0.8280
Cohen	0.0000	0.0000	–	0.9958
FM	0.0996	0.0001	0.187	0.8298
G-mean 2	0.0000	0.0000	0.825	0.9831
Threat Score	0.0099	0.0000	0.047	0.6946

Table 9
Model metrics based on the predicted probability of the LR and the LSTM.

Metric	LR testing	LSTM testing
R2	−14.1471	0.6928
MAE	0.3446	0.0036
RMSE	0.3947	0.0549

Additionally, we have included a scatterplot in Fig. 10 which represents the relationship between the true labels (actual outcomes) and the predicted probabilities generated by the model. Each hexagon in the plot corresponds to a group of data points with similar true labels and predicted probabilities. For improved visualization, jittering has been applied to the true labels, slightly dispersing the data points.

The intensity of color in each hexagon represents the density of data points within that region. Darker shades indicate a higher concentration of data points, while lighter shades indicate lower density. The logarithmic scale color bar on the right side of the plot provides a quantitative measure of the density, with higher values indicating higher density.

5.3. Comparison to baseline

Table 8 presents the results for both baselines and the model's score. For the Recall, Specificity, Miss rate, and Fall Out metrics, a direct comparison with the DD is not possible. As can be observed from Table 8, the model scores better than both baselines for all the performance measures.

The aforementioned metrics utilize the predicted labels. In contrast, the metrics presented in Table 9 are derived from the predicted probabilities assigned to the labels by the models. The Coefficient of determination (R2) assesses the goodness of fit of models by quantifying the proportion of explained variance (Di Bucchianico, 2008). Mean Absolute Error (MAE) measures the average absolute difference between predicted and actual values in a regression model, providing a straightforward assessment of prediction accuracy without considering the direction of errors. Root Mean Squared Error (RMSE) is a measure of the average magnitude of the errors between predicted and actual values in a regression model (Chai, Draxler, et al., 2014).

These metrics were only calculable for the LR and LSTM models, not for the DD model. Likewise to the predicted label metrics, these metrics demonstrate superior results for the proposed LSTM when compared to the LR baseline. However, R2, MAE and RMSE are not well-suited for classification tasks, thus its interpretation in such contexts should be cautious.

6. Discussion

6.1. Major findings

Four dynamic variables from the historic AIS data have been used. All data from fishing vessels operating off of the Dutch coast in 2022 was used for the negative dataset. The minority positive class contained historic AIS data from 18 known cases. The data was segmented using a sliding window and pre-processed. Subsequently, an LSTM model was tuned and fit to the training segment of the resulting data. The activation function was implemented with a weight to ensure that the model favored detecting pick-ups over missing potential pick-ups. The model achieves a 67% ROC-PR score on the test set, a recall of 97% and a precision of 71%. The model only misses 3% of occurring pick-ups and miss-classifies 29% of normal tracks as suspicious. The model outperforms both baseline models, implying a promising classification model for detecting drop-offs. This result is anticipated and desired. The 29% of normal tracks wrongly classified as suspicious is expected due to the unpredictable behavior of fishing vessels. Furthermore, a human observer will act as a second layer of detection and can disregard the model's positive classification upon further inspection. Despite the model still having a relatively high error score, it significantly reduces the number of data points requiring classification by the observer, thereby enhancing the efficiency of the drop-off detection process.

6.2. Limitations

Certain factors must be taken into account when interpreting the model's performance. First, the training data on which the model is trained consists of many negative data points and a very small number of positive data points. Given that the model is only trained on 18 examples of suspicious behavior, it is highly probable that a new occurrence may not resemble any previously seen cases. The model could have a significantly lower likelihood of picking up on the anomaly. Maritime data often reflects a mix of seasonal variations and operational changes, so the model may require periodic retraining to stay current with new patterns of vessel activity. Implementing online learning or adaptive learning techniques could help the model better cope with the nonstationary nature of this domain in the future.

Second, the model has been trained on only cases which are known. Since the model is supervised, it will only detect anomalies that resemble previous cases known by police. Therefore, it is possible that unknown instances of successful pick-ups are included in the negative dataset. As a result, the model may have been told certain movements are normal, even though they may actually indicate a pick-up unbeknownst to the police. This is not reflected in the performance measures presented in this research but will become apparent when applied in real life.

Lastly, the model's reliance solely on fishing vessels' data confines its applicability, restricting its use to this specific vessel type. Our aspiration is to develop a more versatile model capable of detecting drop-offs across all types of maritime traffic in the future.

7. Conclusion

7.1. Method and findings summarized

This study is a significant advancement in bolstering maritime security along the coast of The Netherlands by deploying a robust supervised machine learning model using AIS data. Focusing on detecting drop-offs, a swiftly emerging tactic for smuggling contraband, the research fills a crucial gap in law enforcement efforts to counter illicit activities at sea.

Using the dynamic variables found in AIS data, the study meticulously developed and fine-tuned an LSTM model tailored specifically for maritime applications. Notably, the model was carefully adjusted to prioritize reducing false alerts while maximizing the identification of missed pick-up instances. This targeted approach reflects a deep understanding of the operational needs and challenges faced by law enforcement agencies responsible for maritime surveillance.

The effectiveness of the model is evident from its impressive performance metrics. With a commendable 67% ROC-PR, 97% recall, and 71% precision score, the model significantly outperforms traditional methods and even surpasses the performance of both the DD and the LR baseline. These metrics not only demonstrate the model's reliability but also highlight its practical usefulness in real-world scenarios.

7.2. Importance of this research

By achieving such performance, this research positions the supervised machine learning model as a powerful tool for enhancing maritime security. Its proficiency in precisely identifying drop-offs serves as a crucial initial layer of filtration, with profound implications for law enforcement. It facilitates the proactive interception of illegal activities and shields coastal areas from criminal behavior.

7.3. Future research

As our model is applied and further refined by law enforcement agencies, it is anticipated that more instances of positive data, representing drug trafficking activities, will become available for analysis. These additional data points offer an opportunity to enhance the model's effectiveness through retraining on expanded datasets. With a more balanced distribution between positive and negative instances, the model's performance is expected to improve significantly. Additionally, it is advantageous that we do not have to downsample, as this ensures the preservation of valuable data integrity. Furthermore, the model's capacity to recognize and generalize patterns beyond the specific scenarios it was initially trained on will also increase. This broader generalization is crucial for real-world applications, where novel variations of drop-off methods may occur.

Additionally, exploring positively classified instances within the negative dataset presents an intriguing avenue for investigation. By examining instances where the model identifies suspicious behavior in routes initially labeled as benign, we may uncover previously overlooked instances of illicit activity. Implementing an active learning framework alongside our model could streamline this process, allowing the model to iteratively improve its performance by selectively focusing on instances most likely to enhance its understanding of illicit activities. Overall, these approaches have the potential to significantly enhance the model's efficacy and contribute to more effective detection and prevention efforts in maritime law enforcement.

Furthermore, the current model relies on numerical values of dynamic AIS variables as input. These are latitude, longitude, SOG, and COG. While effective, there is potential to explore alternative representations, such as a four-hot encoded vector, which has shown promise in research. However, it is important to note that implementing this encoding method would require substantial computational resources due to its memory-intensive nature. Another suggestion is to incorporate more features into the model to open up avenues for enhancing its predictive capabilities. Specifically, considering additional variables beyond the existing AIS parameters, such as HDG and ROT, could provide valuable insights into the shipper's intentions, enriching the predictive model. Likewise, incorporating external factors like wind direction and current as predictors could offer further context and improve the model's ability to anticipate unusual behavior.

Next to these advancements, future research could explore transfer learning as a means of leveraging knowledge from related tasks or domains to improve the model's performance. Specifically, pre-training the LSTM model on a larger dataset or a similar task, such as anomaly detection in another maritime domain, could yield valuable insights and accelerate the learning process. By initializing the LSTM network with pre-trained weights, the model can effectively transfer knowledge and patterns learned from the source task to enhance its adaptability and effectiveness in detecting anomalies in sea-based drug transfers.

7.4. Concluding

The model developed in this research is a model that is trained for the classification of the 'drop-off' method and that is easily applicable. If implemented as a live model in further application, it could serve as a first layer of a monitoring system, thereby aiding human observers in the detection of illegal activities. The model could also serve as a baseline for future models in detecting pick-ups or, with adequate tuning, be implemented for other illegal route deviation activities such as human trafficking.

8. Abbreviations

All abbreviations used in this paper can be found in Table 10.

Table 10
Abbreviations and their meaning.

Abbreviation	Meaning
AIS	Automatic Identification System
LSTM	Long Short-Term Memory
IMO	International Maritime Organisation
RNN	Recurrent Neural Networks
TMP	Team Maritime Police
SOG	Speed Over Ground
COG	Course Over Ground
MMSI	Maritime Mobile Service Identity
HDG	Heading
ROT	Rate of Turn
BCE	Binary Cross-Entropy
AUC-PR	Area Under the Precision–Recall Curve
DD	Dutch Draw
LR	Logistic Regression
R2	Coefficient of Determination
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error

CRediT authorship contribution statement

Britt van Leeuwen: Conceptualization, Validation, Formal analysis, Resources, Writing – original draft, Writing – review & editing, Supervision, Project administration, Software, Visualization. **Maike Nutzel:** Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank Rob van der Mei and Sandjai Bhulai for their useful comments on drafts of this paper. We also thank TMP for the use of their data and their insights.

Data availability

The data that has been used is confidential.

References

Arasteh, S., Tayebi, M. A., Zohrevand, Z., Glasser, U., Shahir, A. Y., Saeedi, P., et al. (2020). Fishing vessels activity detection from longitudinal AIS data. In *Proceedings of the 28th international conference on advances in geographic information systems* (pp. 347–356). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3397536.3422267.

Bergstra, J., Yamins, D., & Cox, D. D. (2013). Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th international conference on international conference on machine learning - volume 28* (pp. I–115–I–123). JMLR.org.

Brownlee, Jason (2019). A gentle introduction to mini-batch gradient descent and how to configure batch size. https://machinelearningmastery.com/gentle-introduction-mini-batch-gradient-descent-configure-batch-size/.

Chai, Tianfeng, Draxler, Roland R., et al. (2014). Root mean square error (RMSE) or mean absolute error (MAE). *Geoscientific Model Development Discussions*, *7*(1), 1525–1534.

Chawla, Nitesh V, Bowyer, Kevin W, Hall, Lawrence O, & Kegelmeyer, W Philip (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357.

Chollet, F., et al. (2015). *Keras*. GitHub, https://github.com/fchollet/keras.

de Coning, E. (2011). *Transnational organized crime in the fishing industry: Focus on : Trafficking in persons, smuggling of migrants, illicit drugs trafficking* (pp. 126–138). United Nations Office on Drugs and Crime (UNODC).

Di Bucchianico, Alessandro (2008). Coefficient of determination (R 2). In *Encyclopedia of statistics in quality and reliability*. Wiley Online Library.

do Nascimento, Vinicius D., Alves, Tiago A. O., de Farias, Claudio M., & Dutra, Diego Leonel Cadette (2024). A hybrid framework for maritime surveillance: Detecting illegal activities through vessel behaviors and expert rules fusion. *Sensors*, *24*(17), http://dx.doi.org/10.3390/s24175623, URL https://www.mdpi.com/1424-8220/24/17/5623.

Eriksen, T., Greidanus, H., & Delaney, C. (2018). Metrics and provider-based results for completeness and temporal resolution of satellite-based AIS services. *Marine Policy*, *93*, 80–92. http://dx.doi.org/10.1016/j.marpol.2018.03.028.

Fang, Z., Yu, H., Ke, R., Shaw, S., & Peng, G. (2019). Automatic identification system-based approach for assessing the near-miss collision risk dynamics of ships in ports. *IEEE Transactions on Intelligent Transportation Systems*, *20*(2), 534–543. http://dx.doi.org/10.1109/TITS.2018.2816122.

Ford, J. H., Peel, D., Kroodsma, D., Hardesty, B. D., Rosebrock, U., & Wilcox, C. (2018). Detecting suspicious activities at sea based on anomalies in Automatic Identification Systems Transmissions. *PLOS ONE*, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6084947/.

Greig, N. C., Hines, E. M., Cope, S., & Liu, X. (2020). Using satellite AIS to analyze vessel speeds off the coast of Washington State, U.S., as a risk analysis for cetacean-vessel collisions. *Frontiers in Marine Science*, *7*, http://dx.doi.org/10.3389/fmars.2020.00109.

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

Housden, T., & Guinto, J. (2023). Cocaine Haul Sparks Hunt for Australian trio rescued at sea. BBC News https://www.bbc.com/news/world-australia-64645184.

IMO (1998). Adoption of New and Amended Performance Standards (RESOLUTION MSC.74(69)). https://www.imorules.com/MSCRES_74.69.html.

IMO (2024). International Maritime Organization: AIS transponders. https://www.imo.org/en/OurWork/Safety/Pages/AIS.aspx.

Karim, Fazle, Majumdar, Somshubra, Darabi, Houshang, & Chen, Shun (2018). LSTM fully convolutional networks for time series classification. *IEEE Access*, *6*, 1662–1669. http://dx.doi.org/10.1109/ACCESS.2017.2779939.

Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization. arXiv:1412.6980.

Krawczyk, B. (2016). Learning from imbalanced data: Open challenges and future directions. *Progress in Artificial Intelligence*, *5*, http://dx.doi.org/10.1007/s13748-016-0094-0.

Lane, R. O., Nevell, D. A., Hayward, S. D., & Beaney, T. W. (2010). Maritime anomaly detection and threat assessment. In *2010 13th international conference on information fusion* (pp. 1–8). http://dx.doi.org/10.1109/ICIF.2010.5711998.

Liu, D., Wang, X., Cai, Y., Liu, Z., & Liu, Z. J. (2020). A Novel Framework of Real-Time Regional Collision Risk Prediction Based on the RNN Approach. *Journal of Marine Science and Engineering*, *8*(3), http://dx.doi.org/10.3390/jmse8030224, URL https://www.mdpi.com/2077-1312/8/3/224.

MadeSmartGroup (2014). Made smart group navigator: The fastest nautical charts. https://www.madesmart.nl/.

Mehlbaum, S., van den Akker, A., Verweij, A., Wester, A., van der Borden, R., & Dekker, M. (2021). *Zuiver op de Graat?: Over de betrokkenheid van de visserij bij maritieme drugssmokkel*. Politie & Wetenschap.

Ministerie van Infrastructuur en Waterstaat (2022). AIS: Verkeersinformatie Scheepvaart. Rijkswaterstaat https://www.rijkswaterstaat.nl/water/scheepvaart.

Nguyen, Duong, Vadaine, Rodolphe, Hajduch, Guillaume, Garello, René, & Fablet, Ronan (2019). GeoTrackNet-a maritime anomaly detector using probabilistic neural network representation of AIS tracks and a contrario detection. *IEEE Transactions on Intelligent Transportation Systems*, *23*(6), 5655–5667.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Singh, S. K., & Heymann, F. (2020). Machine Learning-Assisted Anomaly Detection in Maritime Navigation using AIS Data. In *2020 IEEE/ION position, location and navigation symposium* (pp. 832–838). http://dx.doi.org/10.1109/PLANS46316.2020.9109806.

Sun, Y., Wong, A. K. C., & Kamel, M. S. (2011). Classification of imbalanced data: a review. *International Journal of Pattern Recognition and Artificial Intelligence*, *23*, http://dx.doi.org/10.1142/S0218001409007326.

van de Bijl, E., Klein, J., Pries, J., Bhulai, S., Hoogendoorn, M., & van der Mei, R. (2022). The Dutch draw: Constructing a universal baseline for binary prediction models. arXiv:2203.13084.

van den Goorbergh, R., van Smeden, M., Timmerman, D., & Calster, B. Van (2022). The harm of class imbalance corrections for risk prediction models: illustration and simulation using logistic regression. *Journal of the American Medical Informatics Association*, *29*(9), 1525–1534. http://dx.doi.org/10.1093/jamia/ocac093, arXiv:https://academic.oup.com/jamia/article-pdf/29/9/1525/45444435/ocac093.pdf.

Venskus, J., Treigys, P., & Markevičiūtė, J. (2021). Unsupervised marine vessel trajectory prediction using LSTM network and wild bootstrapping techniques. *Nonlinear Analysis. Modelling and Control*.

Wolsing, K., Roepert, L., Bauer, J., & Wehrle, K. (2022). Anomaly detection in maritime AIS tracks: A review of recent approaches. *Journal of Marine Science and Engineering*, *10*(1), 112. http://dx.doi.org/10.3390/jmse10010112.

Yang, C. H., Lin, G. C., Wu, C. H., Liu, Y. H., Wang, Y. C., & Chen, K. C. (2022). Deep learning for vessel trajectory prediction using clustered AIS data. *Mathematics*, *10*(16), 2936. http://dx.doi.org/10.3390/math10162936.

Yang, Ying, Liu, Yang, Li, Guorong, Zhang, Zekun, & Liu, Yanbin (2024). Harnessing the power of machine learning for AIS data-driven maritime research: A comprehensive review. *Transportation Research Part E: Logistics and Transportation Review*, *183*, Article 103426. http://dx.doi.org/10.1016/j.tre.2024.103426, URL https://www.sciencedirect.com/science/article/pii/S1366554524000164.

Yu, Yong, Si, Xiaosheng, Hu, Changhua, & Zhang, Jianxun (2019). A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, *31*(7), 1235–1270.