Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 20(8): 13921–13946.
DOI: 10.3934/mbe.2023620
Received: 07 March 2023
Revised: 07 May 2023
Accepted: 29 May 2023
Published: 20 June 2023

Research article

A method for detecting abnormal behavior of ships based on multi-dimensional density distance and an abnormal isolation mechanism

Lixiang Zhang[1]**, Yian Zhu**[1,]***, Jie Ren**[1]**, Wei Lu**[2] **and Ye Yao**[1]

[1] School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China
[2] School of Information, Xi'an University of Finance and Economics, Xi'an 710100, China

* **Correspondence:** Email: zhuya@nwpu.edu.cn.

Abstract: Abnormal ship behavior detection is essential for maritime navigation safety. Most existing abnormal ship behavior detection methods only build A ship trajectory position outlier detection model; however, the construction of a ship speed outlier detection model is also significant for maritime navigation safety. In addition, in most existing methods for detecting a ship's abnormal behavior based on abnormal thresholds, one unsuitable threshold leads to the risk of the ship not being minimized as much as possible. In this paper, we proposed an abnormal ship behavior detection method based on distance measurement and an isolation mechanism. First, to address the problem of traditional trajectory compression methods and density clustering methods only using ship position information, the minimum description length principle based on acceleration (AMDL) algorithm and Multi-Dimensional Density Clustering (MDDBSCAN) algorithm is used in this study. These algorithms not only considered the position information of the ship, but also the speed information. Second, regarding the issue of the difficulty in determining the anomaly threshold, one method for determining the anomaly threshold based on the relationship between the velocity weights and noise points of the MDDBSCAN algorithm has been introduced. Finally, due to the randomness issue of the selected segmentation value in iForest, a strategy of selectively constructing isolated trees was proposed, thus further improving the efficiency of abnormal ship behavior detection. The experimental results on the historical automatic identification system data set of Xiamen port prove the practicality and effectiveness of our proposed method. Our experiment results show that the proposed method achieves an improvement of about 10% over the trajectory outlier detection based on the local outlier fraction method, about 14% over the isolation-based online anomalous trajectory method in terms of the accuracy of ship position information anomaly detection, and about 3% over the feature fusion method in terms of the accuracy of ship speed anomaly detection. This method improves algorithm efficiency by about 5% compared to the traditional isolation forest anomaly detection algorithm.

Keywords: AIS data; outlier detection; MDDBSCAN; isolated forest algorithm

1. Introduction

Maritime safety has always been the focus of naval navigation, especially with the rapid growth of marine traffic, so it has become an imperative [1]. To ensure the safety of ships during navigation, we need to monitor the navigation information of ships in real-time, such as position and speed information. At present, the automatic identification system (AIS) [2,3] installed in most ships can record the navigation information of ships in real-time. This navigation information includes the ship's unique identification number, i.e., the Maritime Mobile Service Identity, longitude, latitude, speed, course, etc. Using this information can help us to analyze the navigation state of the ship and detect abnormal behavior of the ship [4].

With the development of big data and artificial intelligence technologies in recent years [5,6], the issue of trajectory outlier detection has been well-studied in trajectory data mining. At the same time, there are many offline trajectory outliers detection methods, such as the density-based method, isolation-based anomalous trajectory detection (iBAT) [7], time-dependent widespread routes-based trajectory outlier detection (TPRO) [8], etc. Meanwhile, there are also some online trajectory outlier detection methods, such as isolation-based online anomalous trajectory detection (iBOAT) [9], time-dependent widespread routes-based real-time trajectory outlier detection (TPRRO) [10], driving behavior-based trajectory outlier detection [11] and gravity vector [12]. In these methods, IBAT and iBOAT are based on abnormal isolation mechanisms. TPRO and TPRRO are based on the time-dependent popular route. The gravity vector is based on the distance measurement. Most of the above methods, whether offline detection, online detection or anything else, only consider the position anomaly information of ship behavior and ignore other anomaly information. At the same time, their abnormality thresholds were shown to be difficult to determine during abnormality detection, which led to inconsistent abnormality detection results. To solve the above problems, a ship outlier detection method based on distance measurement and an isolation mechanism is proposed in this paper. Meanwhile, this method provides us with a reasonable basis for determining the threshold for judging speed as abnormal. And it is suitable for online outlier detection and can also detect the outliers of ship position information and velocity information.

First, the method uses the minimum description length principle based on acceleration (AMDL) algorithm to compress ship trajectories. The reason for choosing this algorithm is that the algorithm is based on the minimum description length algorithm and has strong applicability. Meanwhile, the shape of the trajectory output by the other trajectory compression algorithms, such as the Douglas-Peuker algorithm, depends on the determination of the threshold. However, the state of motion and direction of ships may change at any time, so the efficiency of trajectory compression methods that rely on setting a threshold is low and these methods may not achieve good results under the trajectory compression of a ship.

Secondly, to accurately extract the normal behavior model of ships, it is necessary to preprocess AIS data (i.e., identify trajectory clusters and remove noise points). At the initial stage of AIS data processing, the distribution characteristics of unprocessed AIS data are unknown. Therefore, we used the Multi-Dimensional Density Clustering (MDDBSCAN) algorithm (based on Density-Based Spatial Clustering of Applications with Noise) to identify ship trajectory clusters, removed noise points from the original AIS data and extracted the ship's normal behavior model to detect ship position outliers. The DBSCAN algorithm does not require prior knowledge of the number of ship trajectory clusters to be formed, and it can discover any shape of ship trajectory cluster classes.

Thirdly, the method offers a strategy of selectively building the iTree algorithm to construct the iForest algorithm. This algorithm has high efficiency and is suitable for the online detection of abnormal behavior of ships. At the same time, in the process of extracting the correct speed set for ships and removing speed outliers, the algorithm does not need to consider the distribution of the original data. Finally, establishing the relationship between velocity weights in MDDBSCAN and anomaly thresholds will provide one suitable anomaly threshold for detecting ship speed outliers.

The main contributions of this paper are as follows:

1) Regarding the issue of the MDL algorithm only considering trajectory position information, this paper presents the AMDL algorithm, which preserves not only position information but also speed information, unlike the MDL algorithm. The AMDL algorithm, based on the MDL algorithm, forcibly retains the points where the acceleration changes from positive to negative or the acceleration changes from negative to positive.

2) In response to the problem of traditional density clustering only using ship position information as a similarity measure, the MDDBSCAN algorithm was developed in this study. Compared with the traditional density clustering-based ship anomaly detection algorithm, it takes into account the ship's speed factor in the similarity measure of the trajectory cluster, so the ship behavior modeling in the trajectory cluster is more accurate, thus improving the detection of abnormal ship behavior.

3) Due to the randomness issue of the selected segmentation values in iForest, we propose a strategy of selectively constructing isolated trees, which improves the detection efficiency of the isolation forest algorithm for abnormal data compared with the traditional isolation forest algorithm. The strategy can maximize the difference between the number of nodes in the left sub-tree and the right sub-tree to improve the convergence speed for the iForest algorithm.

4) In response to the difficulty in determining the anomaly threshold, by analyzing the relationship between the velocity weights and noise points of the MDDBSCAN algorithm, we have established the connection between velocity weights and anomaly thresholds, which provides a reasonable basis for determining anomaly thresholds. Compared with using grid search or determining the anomaly threshold by experience, this method is more efficient, and the anomaly threshold selection is more explanatory.

The remainder of the paper is organized as follows. Section 2 discusses related work. Section 3 describes basic concepts about the sub-trajectories similarity measurement and trajectory compression. Section 4 presents the abnormal ship behavior detection algorithm. Section 5 discusses the experimental setup and result. Section 6 concludes the article and gives future work.

2. Related works

2.1. Definition of abnormal ship behavior

In this section, this paper first introduces the general definition of abnormal ship behavior, the current methods of abnormal ship behavior detection, followed by the main contributions of this paper. Nowadays, there are many definitions of abnormal behavior of ships, but the main problem is that there is no unified definition. Martineau and Roy [13] defined and classified abnormal ship behavior earlier, but this method only divides ship behavior into two categories: motion anomaly and position anomaly. Portnoy et al. [14] defined abnormal behavior according to the difference between the mathematical model of normal ship behavior and the ship's data to be detected. Zhang and Tang [15] defined the abnormal behavior of a ship as the ship's motion that did not conform to the normal navigation activity law. Lane et al. [16] classified abnormal ship behavior into five categories according to AIS data: deviation from the normal route, abnormal activity of the ship AIS, abnormal arrival of the ship, abnormal distance among ships and an abnormal navigation zone. Laxhammar [17] defined abnormal behavior of ships as the abnormal deviation of ships from the channel and course, sudden acceleration, sudden deceleration, and appearance in areas that should not be entered. It could be seen from the above that different experts had different emphases for the definition of abnormal ship behavior.

The detection of abnormal behavior of ships is the detection of the abnormal trajectory and speed of ships, and the trajectory of ships is composed of the trajectory points of ships. Therefore, we detected the abnormal behavior of ships by detecting the position and speed of the ship trajectory points. Combined with the above definition and analysis, we have defined abnormal ship behavior as the occurrence of a position outlier or velocity outlier in the trajectory points of ships. The position outlier refers to the deviation of the ship trajectory from the historical channel, while the speed outlier refers to the ship entering some particular area that does not conform to the general speed of ships in the area.

2.2. Research on detection methods for abnormal behavior of ships

In recent years, there has been much research on abnormal ship behavior detection, including collaborative computing and distributed methods, deep learning methods, statistical methods, distance measurement methods, outlier isolation methods, knowledge-based and data-driven integrating approaches and so on [18]. In the method based on distance measurement, some trajectories that are far away from most normal trajectories are regarded as outliers. Aiming at the problem of the possible skewness in the distribution of raw AIS data, Bao and Du [12] extracted the mathematical model from its trajectory clusters based on density clustering (DBSCAN) to detect the abnormal behavior of ships. Aiming at the problem that the traditional Trajectory Outlier Detection (TRAOD) algorithm cannot detect outliers from locally dense trajectories, Luan et al. [19] combined the Local Outlier Factor algorithm with the traditional TRAOD algorithm to detect trajectory anomalies. Due to a lack of serious studies on outlier detection for trajectory data, Liang et al. [20] used the trajectory outlier detection based on the local outlier fraction (TODLOF) algorithm to detect outliers in the trajectory dataset. Using an approach based on deep learning, Belhadi et al. [21] and others compared the traditional deep learning methods with data mining, machine learning and other methods, and they have proved the advantages of using traditional deep learning the Convolutional Neural Network algorithm and the Region Convolutional Neural Network algorithm for outlier detection. In the method based on statistical or collaborative computing and distributed methods, Szarmach and Czarnowski [22] proposed a method of using a wavelet transform to detect incorrect AIS data. Chen et al. [23] adopted spark technology to improve the detection efficiency of outliers. Using a method based on isolating outliers, to avoid tricky parameters in their trajectory outlier detection model, Hu et al. [24] used the idea of an isolated forest to isolate outliers. In other related research on ship anomaly detection, Belhadi et al. [25] compared the current outlier detection methods and deeply analyzed various trajectory outlier detection methods consequently, various trajectory outlier detection methods could be well understood. Riveiro et al. [26] provided an overview of the state-of-the-art research about maritime anomaly detection from the perspective of data, methods, systems and user aspects.

Regarding the above methods of abnormal ship behavior detection, they were generally for offline detection. That is, they could not detect abnormal ship behavior in realtime. Although the effect of outlier detection was good in the experiment, it could not be applied to practical engineering. The methods based on distance measurement and mathematical modeling have been widely used for online abnormal ship behavior detection methods. Both judge whether the object detected is an outlier by measuring the distance between the object to be detected and the correct object. However, the problem is that the distance threshold selection significantly impacts the judgment of whether the object to be detected is an outlier. At the same time, the method of abnormal ship behavior detection based on mathematical modeling has poor scalability. At the same time, most other online abnormal ship behavior detection methods only mined the position outlier, not the speed outlier. And then, the anomaly threshold is challenging to determine in these methods, leading to unstable anomaly detection. Regarding the approaches based on deep learning, these methods lack explanatory power for detecting abnormal behavior of ships.

All in all, the above traditional methods for detecting abnormal behavior of ships have certain problems, such as being unable to perform online detection, relying heavily on the selection of thresholds for detection results, low scalability, only mining abnormal position information of ship trajectories, lack of interpretability, etc.

2.3. Advantage of the method based on distance measurement and an isolation mechanism

Based on the above problems, we propose an abnormal ship behavior detection method based on distance measurement and an isolation mechanism, which can not only detect the position outliers of ship trajectory points in realtime, but it can also detect the speed outliers in realtime. Meanwhile, this method improves the MDL algorithm to obtain more accurate compressed ship trajectories , and it provides a reasonable basis for the determination of abnormal speed judgment thresholds. Finally, a strategy of selectively constructing isolated trees is proposed to improve the efficiency of detecting abnormal behavior in ships.

For the outlier detection of the ship position information, the AIS data are first processed and compressed, leading to the minimum length description criterion [27] algorithm based on acceleration (AMDL), which reflects the real navigation information of ships with less AIS data as much as possible. Then the ship position information model is extracted from the trajectory cluster after multi-dimensional density clustering (MDDBSCAN) [28,29]. By comparing the differences between ship trajectory points and the ship position information model, the position outliers of the ship can be detected in realtime.

For the outlier detection of the speed information, this method uses an isolation forest algorithm [30]. First, a functional relationship between the speed weights and abnormal speed judgment threshold is established. Second, detect and eliminate the speed outliers by implementing the isolation forest algorithm to obtain the correct ship speed set in some areas. Finally, the goal is to add the speed to be detected to the correct ship speed set, and then to calculate the score of the speed to be detected. The score can be used to judge whether the speed value is an outlier. The advantage of using an isolation forest algorithm is that it has good efficiency and can meet the needs of online detection. Meanwhile, the algorithm has strong expansibility for outlier detection for ship behavior. On this basis, the efficiency of the traditional isolation forest algorithm is effectively improved by selectively constructing isolated trees, resulting in faster detection of abnormal ship speeds. The method proposed is not only applicable to online anomalous behavior detection for ships, but it can also provide a theoretical reference basis for the establishment of anomalous behavior detection models of other moving targets.

3. The definition of basic concepts

In this section, some related terms and formal expressions are defined first, which mainly include the relevant definitions of sub-trajectory similarity measurement [31] and trajectory compression.

3.1. Sub-trajectory similarity measurement method

There are three types of distances between trajectory segments: vertical distance ($d_{\perp}$), parallel distance ($d_{||}$), and angular distance (d_{θ}). These three types of distances are used to measure the similarity of trajectory segments. Figure 1 shows these three distances via a formal method.

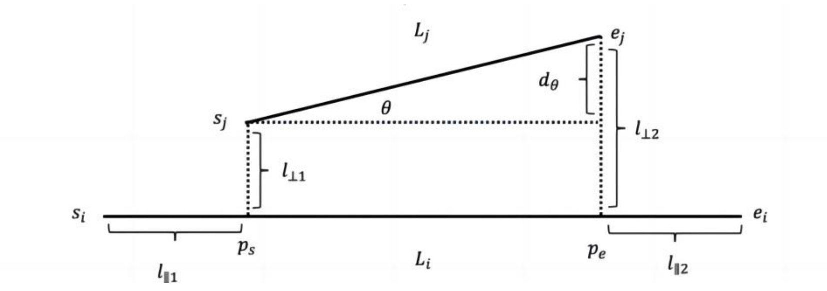


Figure 1. Trajectory segment distance.

It is assumed that there are two trajectory segments in space, namely $L_j = s_j e_j$ and $L_i = s_i e_i$, where s_i and e_i respectively represent the two endpoints of the segment L_i .Then, e_j and s_j respectively represent the two endpoints of the segment L_j. Here, it is assumed that the length of the segment L_j is shorter than L_j.

The vertical distance of L_i and L_j is defined as Formula (1), where the two endpoints (s_j and e_j) of segment L_j are projected as p_s and p_e on the segment L_i. At the same time, the Euclidean distance from the point s_j to p_s is $l_{\perp 1}$, and the Euclidean distance from the point e_j to p_e is $l_{\perp 2}$.

$$d_{\perp}(l_i, l_j) = \frac{l_{\perp 1}^2 + l_{\perp 2}^2}{l_{\perp 1} + l_{\perp 2}} \tag{1}$$

The parallel distance of L_i and L_j is defined as Formulas (2)–(4), where the two endpoints (s_j and e_j) of segment L_j are projected as p_s and p_e on the segment L_i. At the same time, the Euclidean distance from the point s_j to p_s is $l_{\perp 1}$, and the Euclidean distance from the point e_j to p_e is $l_{\perp 2}$.

$$d_{||}(l_i, l_j) = min(\, l_{||1}, l_{||2}) \tag{2}$$

$$l_{||1} = min(\, d(s_i, p_s), d(e_i, p_s)) \tag{3}$$

$$l_{||2} = min(\, d(e_i, p_e), d(s_i, p_e)) \tag{4}$$

The angular distance is defined as Formula (5). The angle of L_i and L_j is θ（$0 \le \theta \le \pi$）. Generally, angle θ selects the smaller angle between L_i and L_j. $|l_j|$ represents the length of the line segment L_j.

$$d_\theta = \begin{cases} |l_j| \times sin\,\theta & 0 \le \theta \le \frac{\pi}{2} \\ |l_j| & \frac{\pi}{2} \le \theta \le \pi \end{cases} \tag{5}$$

The angular distance is usually used for the trajectory segment with direction. When dealing with the trajectory segment without direction, the angular distance can be simply defined as $|l_j| \times sin\,\theta$.

3.2. Trajectory compression technique

The purpose of trajectory compression is to describe the characteristics of a trajectory with as few points as possible. This work adopts the AMDL principle, which is based on the MDL algorithm. At the same time, by using the MDL algorithm, it forcibly retains the points where the acceleration changes from positive to negative or the acceleration changes from negative to positive. The reason for choosing this algorithm is that the algorithm has strong applicability and the shape of the trajectory output does not depend on the determination of the threshold.

The AMDL segmentation cost is shown in Formulas (6) and (7), and $AMDL_{par} = L(H) + L(D|H)$. In Formula (6), p_i represents the trajectory point and p_{c_i} represents the point selected by the AMDL algorithm; additionally, $len(p_i, p_j)$ represents the Euclidean distance between two trajectory points.

$$L(H) = \sum_{i=1}^{par_i - 1} log_2(\, len(p_{c_i}, p_{c_{i+1}})) \tag{6}$$

$$L(D|H) = \sum_{i=1}^{par_i - 1} \sum_{k=c_i}^{c_{i+1}-1} \{ log_2^{(d_{\perp}(p_{c_i} p_{c_{i+1}}, p_k p_{k+1}))} + log_2^{(d_{\theta}(p_{c_i} p_{c_{i+1}}, p_k p_{k+1}))} \} \tag{7}$$

No segmentation cost , i.e., $AMDL_{nopar}$ is the total length of the trajectory from point p_i to point p_j, and the formula is given by Formula (8).

$$AMDL_{nopar} = \sum_{i=1}^{j-1} len(p_i, p_{i+1}) \quad (8)$$

4. Abnormal ship behavior detection method

This paper presents an abnormal ship behavior detection method based on multi-dimensional density clustering and an abnormal isolation mechanism. In this method, massive AIS data must be compressed, so this study adopted an AMDL algorithm. Second, the MDDBSCAN algorithm must be carried out on the compressed data. At the same time, the trajectory cluster is divided into 10 grids. Then, a position information model of the ship trajectory is extracted on each grid. By measuring the distance difference between the point to be detected and the correct model, the method can judge whether the ship's position is abnormal. Thirdly, in each grid, the isolation forest algorithm based on selectively constructing isolated trees is used to remove the abnormal speed points of ships to extract the correct speed set. By calculating the abnormal score value of the speed to be detected in the speed set, the method can judge whether the ship speed is abnormal. During the process, the connection between the velocity weights in MDDBSCAN and anomaly thresholds are established, providing a reasonable basis for determining speed anomaly thresholds. The detection flow chart is shown in Figure 2.

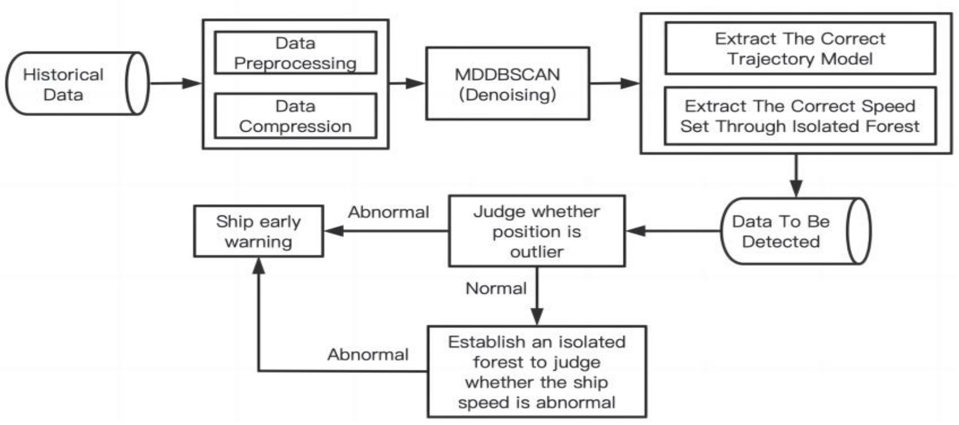

Historical Data
Data Preprocessing
Data Compression
MDDBSCAN (Denoising)
Extract The Correct Trajectory Model
Extract The Correct Speed Set Through Isolated Forest
Data To Be Detected
Judge whether position is outlier
Abnormal
Ship early warning
Normal
Establish an isolated forest to judge whether the ship speed is abnormal
Abnormal

Figure 2. Flow chart of abnormal ship behavior detection.

4.1. Ship trajectory compression

This paper's data compression method is based on the AMDL algorithm. The core idea of the AMDL algorithm is to extract feature points from a trajectory. At the same time, the trajectory compression by this method has two ideal properties: accuracy and simplicity. Accuracy refers to the trajectory after segmentation and the trajectory before segmentation having the same characteristics as much as possible. At the same time, simplicity means that the feature points to be extracted from the original trajectory should be as few as possible. Therefore, the AMDL algorithm process mainly includes two parts: judging whether there are trajectory points with the positive and negative transformations of front and rear acceleration in the segmented trajectory point set and judging whether to segment the trajectory. In the AMDL algorithm, p_{i} represents one trajectory point, $p_{\mathrm{v_i}}$ represents trajectory points of positive or negative transformation of front and rear acceleration, and $p_{c_{\mathrm{i}}}$ represents characteristic points of the ship trajectory selected by the AMDL algorithm.

Algorithm 1 AMDL

```
Input：One trajectory $TR(p_1, p_2 \dots p_n)$
Output：All feature points of the trajectory $CP\left(p_{c_1}, p_{c_2} \dots p_{Par_i}\right)$
1：Add P1 into the set CP；  /*the start point*/
2：startIndex = 1, length = 1;
3：while  startIndex + length ≤ n  do
4：  currIndex = startIndex + length;
5：  Add all points from startIndex to currindex to the set temp
6：  Check if $p_{\mathrm{v_i}}$ exits in the temp set
7：  if non-existent then
8：    costpar = AMDLpar(PstartIndex,PcurrIndex);
9：    costnopar = AMDLnopar(PstartIndex,PcurrIndex);
10：    if costpar>costnopar then
11：      Add the Pcurrindex-1 point to the set CP;
12：      startIndex = currIndex-1 , length = 1;
13：    else
14：      length = length + 1;
15：  else
16：    mark the point as PcurrIndex
17：    Add the pcurrIndex point to the set CP;
18：    startIndex=currIndex;
19：    length = 1;
20：Add Pn to the set CP
```

In Algorithm 1, the first two lines are the initialization operations of the algorithm. The process from the third line to the end aims to find characteristic points of the ship trajectory. During the process, the AMDL algorithm calculates the partition cost (costpar) and the no partition cost (costnopar) for each trajectory point. If costpar is greater than costnopar, then the previous point of that point will be selected as a characteristic point. Meanwhile, $p_{\mathrm{v_i}}$ will also be selected as the characteristic point.

4.2. Position outlier detection for ship trajectory points

For the detection of ship trajectory position point outliers, it is necessary to carry out multi-dimensional density clustering on the compressed AIS data. The MDDBSCAN algorithm is proposed to solve the above problem. At the same time, the trajectory cluster needs to be meshed, and then a correct model of ship position information is extracted on each grid.

The traditional DBSCAN algorithm only considers the Euclidean distance between points. The clustering object of the MDDBSCAN algorithm in this paper is the sub-trajectory, and the clustering process adopts the idea of DBSCAN [28,29]. In this process, the clustering objects are sub-trajectories, that is $Sub_i = \left(p_{c_i} p_{c_j}\right)$, and the sub-trajectory velocity can be represented by $V_{Sub_i} = \frac{1}{2}\left(V_{p_{c_i}} + V_{p_{c_j}}\right)$. At the same time, the similarity distance of sub-trajectories will be calculated by $Dist(Sub_i, Sub_j) = \omega_{\perp} d_{\perp}(Sub_i, Sub_j) + \omega_{||} d_{||}(Sub_i, Sub_j) + \omega_{\theta} d_{\theta}(Sub_i, Sub_j) + \omega_v V_{Sub_i}$.

The algorithm flow of MDDBSCAN is as follows.

Algorithm 2　MDDBSCAN

```
Input：(1) Sub-trajectory set $D = \{\mathrm{Sub}_1,\mathrm{Sub}_2 \ldots \mathrm{Sub}_n\}$
       (2) Neighborhood radius $\varepsilon$， Minimum number of entities($MinSubs$)
Output：Clusters set $S = \{s_1, s_2 \ldots s_n\}$
/*STEP 1*/
clusterID = 0;   /*one initial id*/
Mark all sub-trajectories as unclassified
for each ($Sub_i \in D$) do
  if $Sub_i$is not classified then
    Compute $N_\varepsilon(Sub_i)$ /*find sub-trajectory $Sub_i$ neighborhood*/
    if $|N_\varepsilon(Sub_i)| \geq MinSubs$ then
      allocate $clusterID$ to $\forall Sub_i \epsilon N_\varepsilon(Sub_i)$);
      put $N_\varepsilon(Sub_i) - Sub_i$ into queue $Q$;
  /* STEP 2 */
      ExpandCluster ($Q, clusterID, \varepsilon, MinSubs$)
      clusterID = clusterID + 1;
    else
      mark $Sub_i$ as noised sub-trajectory;
/*STEP 3*/
for each ($s_i \in S$) do
  if $|s_i| <$ minSubs then
    remove $s_i$ from $S$;
/*STEP 2 find density connection set*/
ExpandCluster ($Q, clusterID, \varepsilon, MinSubs$) {
  while $Q \neq \varphi$ do
    Define $M$ as the first sub-trajectory to be checked in the $Q$;
    Compute $N_\varepsilon(M)$ ;
    if  $|N_\varepsilon(M)| \geq MinSubs$ then
      for each ($X \in N_\varepsilon(M)$)do
        if $X$ is not classified or $X$ is noised then
          allocate $clusterID$ to $X$;
        if $X$ is not classified then
          put $X$ into queue $Q$;
    remove $M$ from queue $Q$;
}
```

In Algorithm 2, STEP 1 includes two parts: algorithm initialization and searching for the neighborhood of sub-trajectories. STEP 2 aims to find the density connection set and STEP 3 aims to form clusters and remove noise data.

After completing the multi-dimensional density clustering of sub-trajectories, it is necessary to establish a correct model of ship position information in each cluster. First, the trajectory cluster needs to be meshed. In this work, the sub-trajectory cluster is divided into 10 grids, that is, 10 detection models are generated. Then, the center vector is established in each grid, and the center vector will be used as the detection benchmark to judge whether the position information is abnormal. The center vector is defined as follows: $CV = (avgX, avgY, mediumD)$.

$avgX$ denotes the average X coordinate of all trajectory points in some grid. $avgY$ denotes the average Y coordinate of all trajectory points in some grid. $mediumD$ denotes the median distance. Let all sub-trajectories in the grid be represented as a set a space{sub_1,sub_2 ... sub_n}, and set $\{p_{c_1}, p_{c_2} \dots p_{c_n}\}$ denotes all trajectory points. Then the calculation formula for the components of the CV is as follows.

1) average X coordinate: $avgX = \frac{\sum_{i=1}^{n} p_{c_i}.x}{n}$;

2) average Y coordinate: $avgY = \frac{\sum_{i=1}^{n} p_{c_i}.y}{n}$;

3) medium distance: $mediumD = \frac{\sum_{i=1}^{n} len(p_{c_i},(avgX,avgY))}{n}$; and len denotes the Euclidean distance between two points.

After the center vector is determined in the grid of each sub-trajectory, the abnormal position of ship trajectory points can be judged. The detection idea is to measure the relative distance between the point to be detected and the center vector. If the relative distance exceeds the threshold range, it is considered that the position of the trajectory point is abnormal. If the relative distance is within the threshold, the position of the current point is considered normal. The formula of the relative distance between the point to be detected and the center vector is given by Formula (9).

$$CRD(p, CV) = \frac{len(p,(CV.avgX,CV.avgY))}{CV.mediumD} \tag{9}$$

In Formula (9), p is the point to be detected. When CRD > 1, the distance from the point to be detected to the center vector is greater than the average distance from all points in the grid area to the center vector. When CRD = 1, it is explained that the distance from the point to be detected to the center vector is equal to the average of the distance from all points in the grid area to the center vector. When CRD < 1, the distance from the point to be detected to the center vector is less than the average distance from all points in the grid area to the center vector.

In determining the threshold (*CRD(P,CV)*), we assume that the distance from all points in the sub-trajectory grid to the center vector approximately satisfies the normal distribution then, the three standard deviations criterion is used to determine the threshold, and the formula is given by Formula (10). When $y_{p_i} = 0$, it indicates that the position of the point to be detected is normal. When $y_{p_i} = 1$, it indicates that the position of the point to be detected is abnormal.

$$y_{p_i} = \begin{cases} 0, \overline{CRD} - 3\sigma < CRD_i < \overline{CRD} + 3\sigma \\ 1, \quad else \end{cases} \tag{10}$$

In the MDDBSCAN algorithm, the method takes into account the ship's speed factor in the similarity measure of the trajectory cluster. So, the extracted ship behavior modeling in the sub-trajectory cluster is more accurate, and the iForest algorithm which is used to detect abnormal ship speed can achieve better results after removing noise data that considers the speed factor.

4.3. Speed outlier detection for ship trajectory points

For speed outlier detection for ships, the isolation forest algorithm based on selectively constructing isolated trees is used to extract the correct ship speed set. The algorithm has high efficiency and is suitable for the online detection of abnormal behavior of ships. At the same time, in the process of extracting the correct speed set for ships and removing speed outliers, the algorithm does not need to consider the distribution of the original data. The isolation forest algorithm is more suitable for fewer data sets [32]. Ten grids have been divided for the trajectory cluster, as described in the previous section. Still, to help the isolation forest algorithm achieve better results, each grid in the previous section is divided into four grids again. Through this method, we can remove the abnormal speed of ships in each grid, to obtain the correct speed set for ships in each grid. When it is necessary to detect whether the speed of the ship is abnormal, the method will add the speed of the ship to be detected to the correct set of ship speed in the grid at the corresponding position, as well as determine whether the speed is abnormal by calculating the abnormal score value of the speed to be detected.

Algorithm 3 Selective construction of isolated trees -- iTree(x,e,l)

```
Input：X, e, l.
Output：an iTree
if $e \geq l$ or $|X| \leq 1$ then
    return exNode {$|X|$};    /*return Size of data set*/
else
    Choose any value p between the maximum and minimum values of X;
    $X_l \leftarrow filter(X, X_i < p)$
    $X_r \leftarrow filter(X, X_i > p)$
end if
if |X_l |/|X_r | ⩾ Ratio && the first division then
    view the tree as a bad isolated tree, so not to build;
end if
return InNode{ $Left \leftarrow iTree(X_l, e+1, h)$  $Right \leftarrow iTree(X_r, e+1, h)$, $splitValue \leftarrow p$ };  /*non-leaf node*/
```

iForest is similar to a decision tree and random forest. iForest is composed of isolated trees (iTree). iTree uses random binary trees. Each node connects two child nodes or directly connects a leaf node. Randomly sampling partial data to construct an isolated tree can ensure a difference between different trees. To build an isolated tree, we need to select a feature (speed is selected here) and randomly select a segmentation value to recursively segment the data set until the maximum height limit of the tree is met or the number of samples of the tree nodes is only one. The maximum height limit (*h*) of the tree is related to the number of sub-samples(φ), $h = ceiling(log_2(\phi))$.

Generally, when dividing left and right sub-trees, the isolation forest algorithm randomly selects a number between the minimum and maximum values from the data set as the segmentation value. Samples smaller than the segmentation value will be divided into the left sub-tree, and samples larger than the segmentation value will be divided into the right sub-tree. Due to the randomness of the selected segmentation value, there will be differences in the ability of each isolated tree to distinguish outliers. For the identification of abnormal data, it is expected that the segmentation value can maximize the difference between the number of nodes in the left sub-tree and the right sub-tree, to improve the convergence speed of the algorithm. Therefore, we propose an algorithm for selectively building an isolated tree. The algorithm flow for constructing the isolated tree of ship speed is as shown in Algorithm 3. In Algorithm 3, the terms Ratio represents the ratio of the number of samples divided into the left (right) sub-tree to the number of samples divided into the right (left) sub-tree during the first division. X represents the data set to enter. e denotes the current height of the tree. l denotes the maximum height limit of the tree.

It is usually necessary to build 100 such isolated trees to construct an isolated forest (iForest). The judgment of outliers in iForest is based on the average height of the outlier on 100 trees (i.e., path length). The average height of outliers in iForest is usually low. For iForest, given a data set containing *n* samples, the average path length of the tree is as given by Formula (11).

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \tag{11}$$

H(*i*) is a harmonic number, which can be estimated as ln(i) + 0.5772156649. *c*(*n*) is the average value of path length for a given number of samples n, which is used to standardize the path length *h*(*x*) of sample X. The path length *h*(*x*) of sample point *x* is the number of edges from the root node to the leaf node of iTree. The algorithm flow for calculating *h*(*x*) is as follows.

Algorithm 4 Calculate the length of the sample in the tree -- $\boldsymbol{PathLength(x, T, e)}$

Input：*x* is a sample. *T* denotes an iTree. e denotes the current height of *x* on iTree.（The initial value is 0）

Output：the height of *x* on iTree.

```
1：if  $T$ is a leaf node then
2：    return $e + c(T.size)$
3：end if
4：if $x < T.splitValue$ then
5：    return $PathLength(x, T.left, e + 1)$
6：else
7：    return $PathLength(x, T.\text{right}, e + 1)$;
8：end if
```

Outlier detection using iForest is performed by calculating the score of sample X. The score of sample *x* is defined as Formula (12).

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{12}$$

E (*H* (*x*)) represents the average path length of sample *x* on all iTrees in the isolated forest. *c*(*n*) is the average value of path length for a given number of samples *n*, which is used to standardize the path length *H*(*x*) of sample X. The relationship between the score *s* and *E* (*H* (*x*)) is shown in Figure 3.

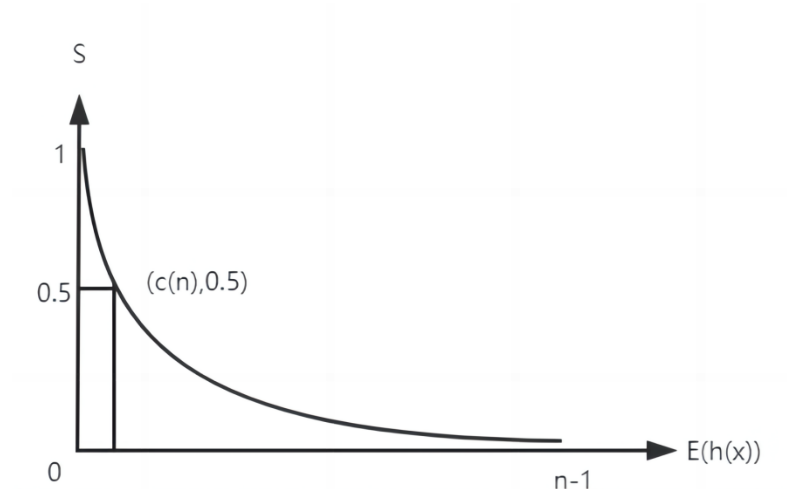


Figure 3. the relationship between the score *s* and *E* (*H* (*x*))

It can be seen from Figure 3 that when *E* (*H* (*x*)) → *C* (*n*), *s* → 0.5, that is, when the average path length of sample *x* is close to the average path length of the iTrees, it is difficult to distinguish whether it is an outlier or not. When *E* (*H* (*x*)) → 0, *s* → 1, that is, when the score of X is close to 1, it is determined to be abnormal. When *E* (*H* (*x*)) → *n*-1, *s* → 0, it is determined to be normal.

Indeed, determining whether the data point is an outlier depends on the threshold value to determine whether the data point is abnormal. If the threshold of deciding an outlier is too high, the speed outlier cannot be detected in the data set as much as possible. If the threshold is too low, misjudging the normal data as abnormal is possible. Here, the threshold can be determined by the relationship between the speed weight and noise points in multi-dimensional density clustering.

Usually, multi-dimensional density clustering is mainly used to measure the position differences of ship sub-trajectories, and its speed factor has little effect on the positions of trajectory points. Therefore, the value of its velocity weight should generally not exceed the reciprocal of the number of dimensions. If the number of noise points decreases obviously with the increase of speed weight, it indicates that the ship speed in the data set is relatively average. So the speed weight should be taken as a smaller value, and the threshold needed for determining the speed as an outlier should not be too large. If the ship speed changes significantly in the data set, the speed weight can be appropriately increased, but it should not exceed the reciprocal number of dimensions of multi-dimensional density clustering. After selecting the appropriate speed weight, the relationship between the threshold and the speed weight is defined as shown in Formula (13).

$$score(\omega) = \frac{k}{1+e^{-\omega}} \tag{13}$$

w is the velocity weight and k is the harmonic number. When the score of ship speed at a certain moment is greater than *score*(w), the ship speed is abnormal; otherwise, it is normal.

$$y_{v_i} = \begin{cases} 1, \ score(v_i) \geq score(\omega) \\ 0, \ else \end{cases} \tag{14}$$

When $y_{v_i} = 1$, the speed is abnormal. When $y_{v_i} = 0$, the speed is normal. After removing the abnormal speed points of ships and extracting the correct speed setting in each grid, by calculating the abnormal score value of the speed to be detected in the correct speed set, Formula (14) can judge whether the ship speed is abnormal.

All in all, by Formula (13), the method will provide us with one reasonable basis for using the threshold value to judge speed as abnormal. Meanwhile, the strategy of selectively constructing iTree will accelerate the detection of abnormal ship speed.

5. Experiments and results analysis

During experiments, first, the original AIS data were preprocessed (improvement of data quality and data compression, etc), and then multiple detection methods for abnormal ship behavior (ship's position outliers and speed outliers) were compared from four perspectives (recall, precision, F1 score and accuracy). Finally, we carried out ablation experiments. The experimental hardware environment in this study was Intel® Core™ i7-8700 octa-core CPU (3.20 GHz), 8 GB RAM; the software experimental environment was Windows 10, Python 3.8 and JDK 1.8.

5.1. Data Preparation

The data [33] selected in the experiment were the AIS data of a passenger ship near Xiamen port (the whole journey is about 18 km), a total of 40011 data point. The spatial area has a longitude of 117.77 to 118.63, latitude of 24.09 to 24.69 and time range from November 29, 2018 to January 3, 2019. However, the unprocessed raw AIS data may have data quality issues, which can affect the construction of abnormal ship behavior detection models [34]. Therefore, for the issue of raw AIS data quality, we conducted relevant research and processing, such as the interpolation of trajectory breakpoints using the cubic spline method and identification and removal of abnormal AIS data (abnormal stop points, abnormal acceleration points, abnormal drift points, abnormal turning points) to enhance the continuity and integrity of AIS data and improve the quality of AIS data [3]. Figure 4 shows the number of ship trajectory points after data preprocessing, MDL compression and AMDL compression.

As seen in Figure 4, the number of data points before and after data preprocessing varies greatly. If the abnormal data caused by AIS equipment abnormality is not removed, the ship's abnormal behavior analysis will be significantly affected. The AMDL compression algorithm is an improvement of the MDL algorithm. Based on the MDL algorithm, it forcibly retains the points where there is a positive or negative acceleration transformation. Therefore, the AMDL algorithm can better reflect the real characteristics of ship motion. The comparison of a trajectory point before and after compression is shown in Figure 5. It can be seen that the compressed trajectory points have a good balance of accuracy and simplicity.

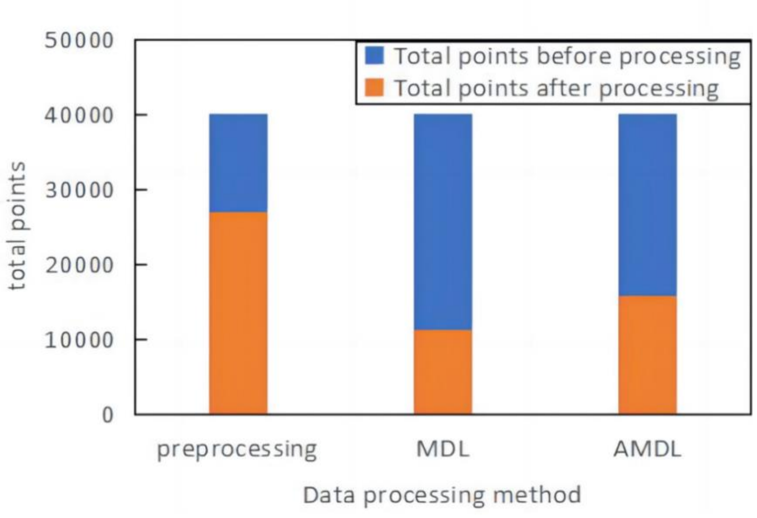
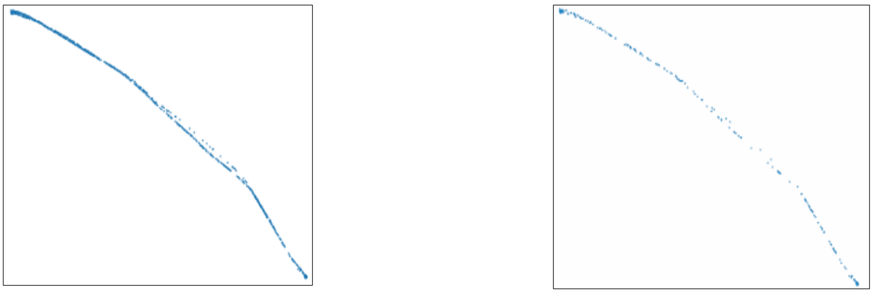


Figure 4. Comparison of the number of points before and after data processing.



(a) Before compression

(b) After compression

Figure 5. the comparison of result before and after trajectory compression.

5.2. Detection and analysis of abnormal ship behavior

For the analysis of ship behavior anomaly detection, we have analyzed it from the perspective of recall, precision, F1 score and accuracy. Since AIS has no official standard data set, to label the data set correctly as much as possible, the noise points of multi-dimensional density clustering were marked as abnormal trajectory points, and the rest were marked as normal trajectory points. There were 935 abnormal trajectory points and 14798 normal trajectory points. Here, the trajectory outlier detection method (MDDBSCAN) in this paper is compared with the TODLOF trajectory outlier detection method [20] and the isolation-based trajectory outlier detection algorithm (IBTOD) trajectory outlier detection method [24] in terms of detection rate and false alarm rate. The confusion matrices for the detection results for normal and abnormal trajectory points are shown in Figures 6–8. Meanwhile, the MDDBSCAN method was compared with TODLOF, IBTOD, graph attention network [35], Long Short-Term Memory, and feature fusion methods [36], and the results are shown in Figure 9 and Table 1.

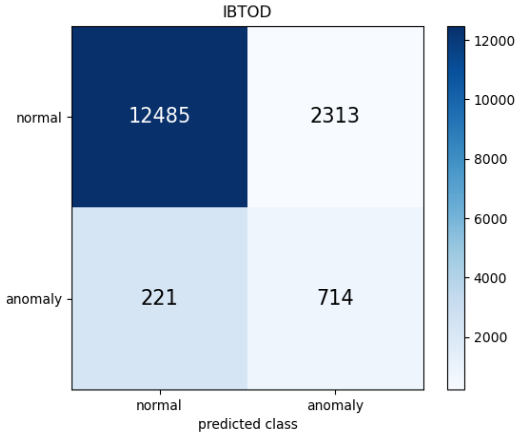


Figure 6. The result of IBTOD algorithm.

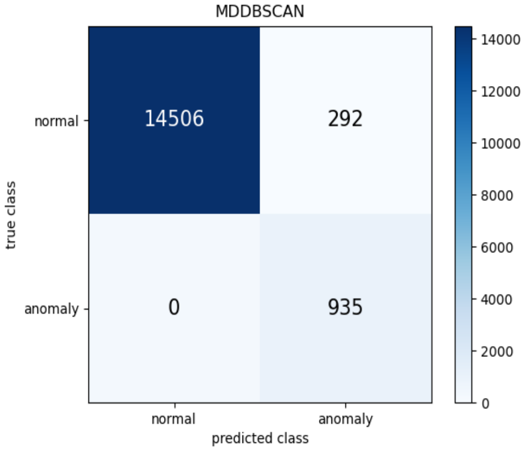


Figure 7. The result of MDDBSCAN algorithm.

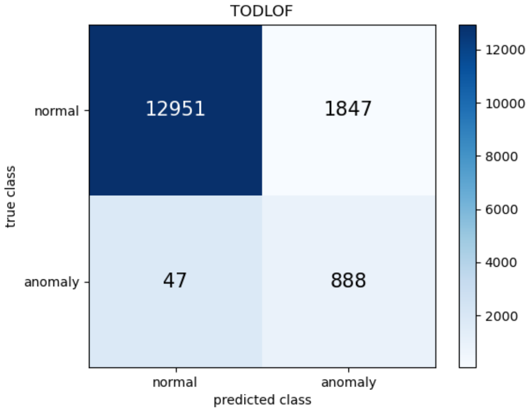


Figure 8. The result of TODLOF algorithm.

Table 1. The ship position anomaly detection results for different methods.

Methods	Recall	Precision	F1	Accuracy
MDDBSCAN	0.9803	1	0.9901	0.9814
TODLOF	0.8752	0.9964	0.9319	0.8796
IBTOD	0.8437	0.9826	0.9078	0.8389
GAT	0.9135	0.8906	0.9013	0.9022
LSTM	0.7214	0.8049	0.7131	0.7651
Feature Fusion	0.9700	0.9650	0.9600	0.9600

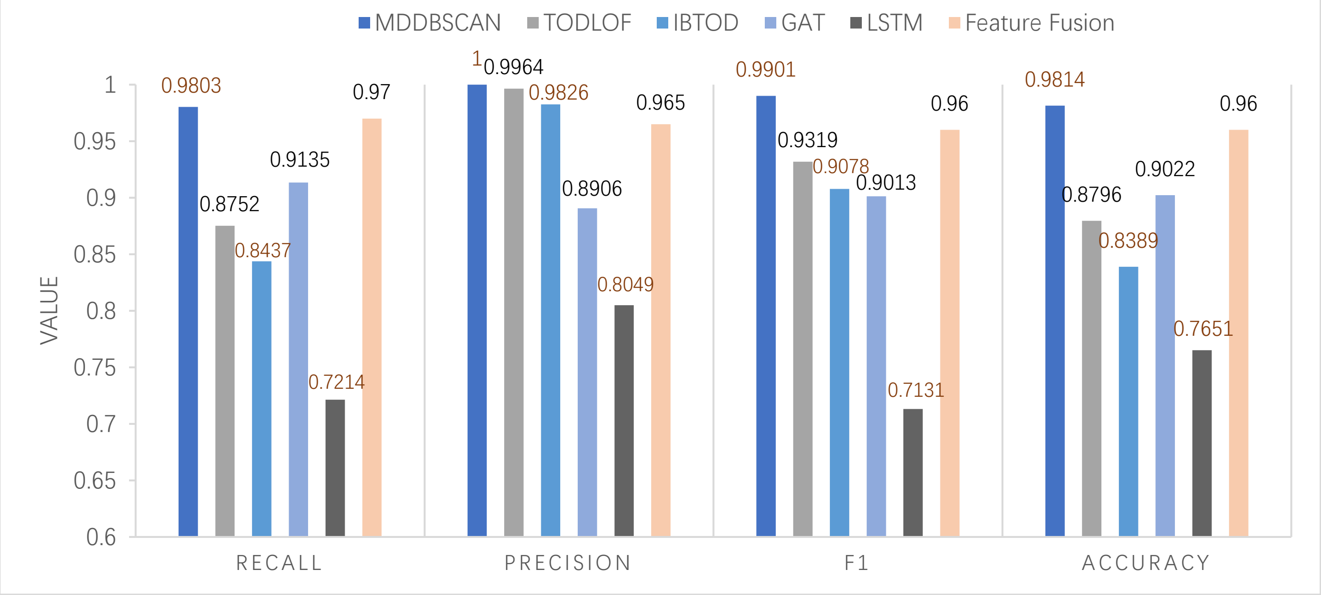
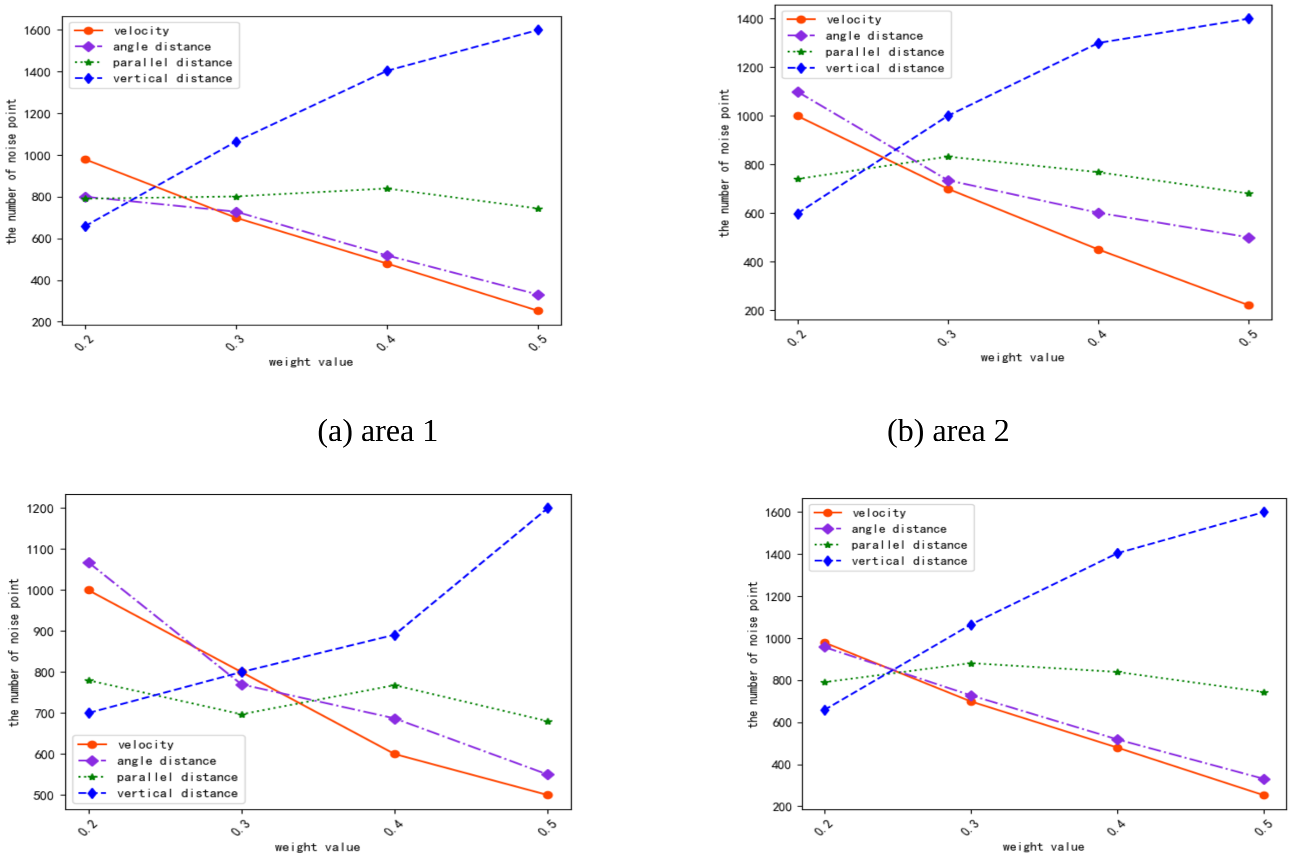


Figure 9. Ship position anomaly detection results for different methods.

Table 1 and Figure 9 show that the MDDBSCAN method outperforms other methods. Analyzing the reasons, the core idea of TODLOF is based on the local outlier factor algorithm, which requires that the detected data must have an obvious density difference. However, for a ship trajectory with a fixed round-trip destination it is difficult to always ensure the obvious density difference, which limits the application scenario of the algorithm. The core idea of the IBTOD is based on the isolation forest algorithm. Still, this algorithm often requires a small data set, and a large number of samples will reduce the ability of isolated forest outliers because normal samples will interfere with the isolation process and reduce the ability to isolate outliers. At the same time, the algorithm assumes that the number of abnormal samples in the overall model is tiny, so the application scenario of the algorithm is also relatively limited. The GAT, LSTM and feature fusion methods are all based on deep learning. Their detection capability depends on the quality of the training data set and the appropriate hyper-parameters, and their detection effect is unstable.

For the detection and analysis of ship speed outliers, 246 ship speed values in a grid area were selected in this study. Five of the values were marked as outliers, and the rest were marked as normal values. Figure 10(a)–(d) all describe the variation of the number of noise points with four different weight values. For example, in the experiment in Figure 10(a), when the velocity weight was set to 0.2, the other three weights were set to one-third of 0.8 and when the velocity weight was set to 0.25, the other three weights were set to one-third of 0.75.



(a) area 1

(b) area 2

(c) area 3

(d) area 4

Figure 10. The number of noise points under different weights.

Four speed weight values were selected to measure the relationship between speed weight and noise points. As seen from Figure 10 above, compared with the increase of the weight of the other three dimensions, the number of noise points has an undeniable downward trend with the increase of speed weight. This reflects that the speed difference of ships in the data is relatively average, so the value of speed weight cannot exceed the weight value of the other three dimensions. Next, a different threshold for determining the speed as abnormal was calculated according to different speed weight values. The confusion matrices for the detection results are shown in Figures 11–13.

It can be seen that the score ($w \leq 0.25$) can provide a more appropriate anomaly threshold for judging whether the speed is abnormal from Figures 11–13. According to Figures 11–13, the recall, precision, accuracy and F1 values of the model can be calculated under different anomaly thresholds, and the method was also compared with feature fusion; the results are shown in Figure 14 and Table 2.

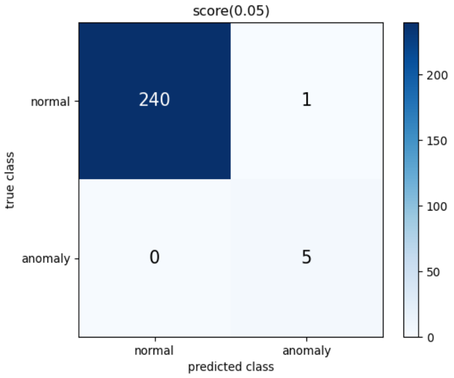


Figure 11. The result of score(0.05).

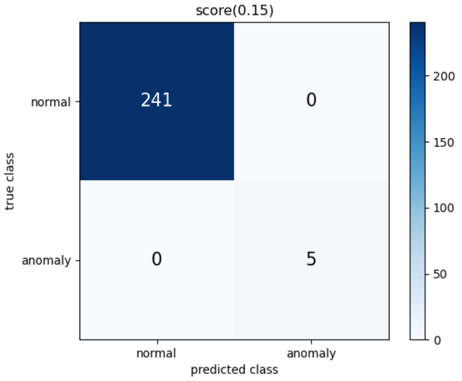


Figure 12. The result of score(0.15).

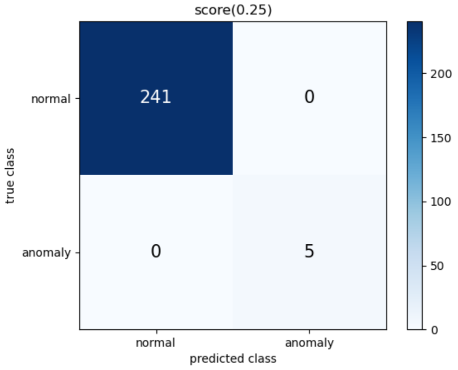


Figure 13. The result of score(0.25).

Table 2. The ship position anomaly detection results for different methods.

Threshold/Method	Recall	Precision	F1	Accuracy
score(0.05)	0.9959	1	0.9979	0.9959
score(0.15)	1	1	1	1
score(0.25)	1	1	1	1
0.75	1	0.9877	0.9938	0.9879
0.8	1	0.9797	0.9897	0.9797
Feature Fusion	0.9600	0.9700	0.9600	0.9600

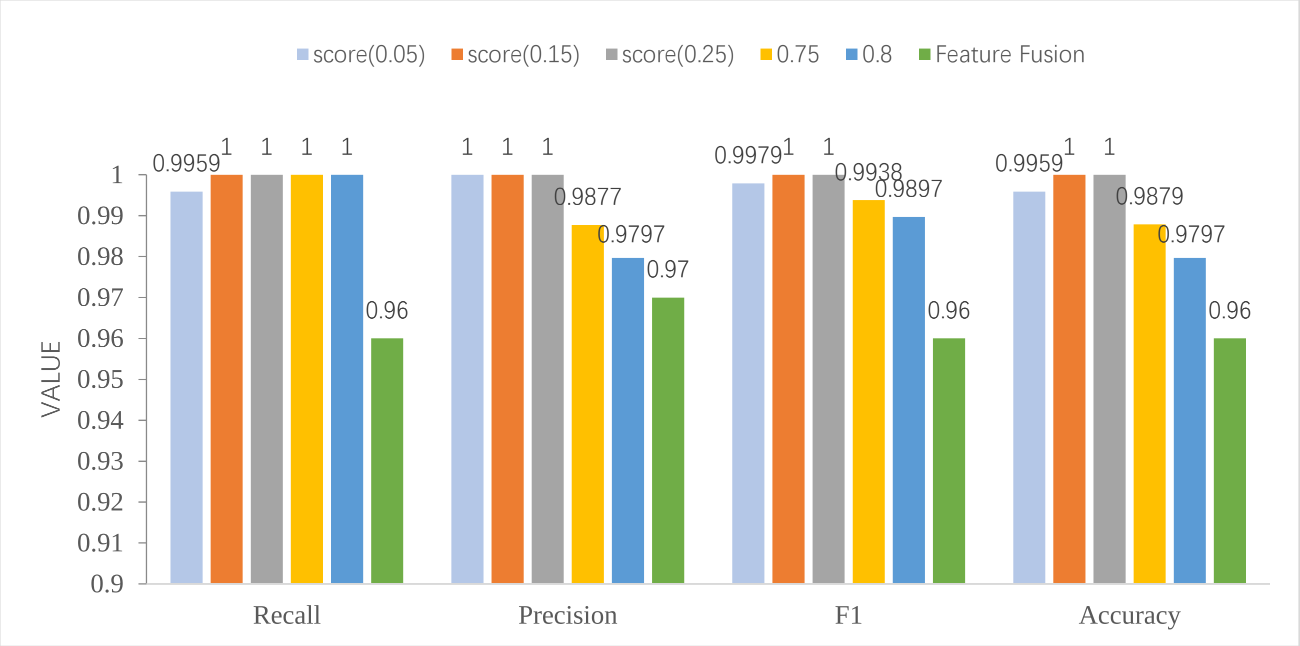


Figure 14. Ship position anomaly detection results for different methods.

Table 2 and Figure 14 show that when *w* is less than 0.25, the model can be guaranteed to detect abnormal ship speed to a greater extent by using the *score*(*w*) formula. It can be seen that *score*(*w*) can be used to determine a more appropriate anomaly threshold for judging whether the speed is abnormal. When the anomaly threshold > 0.75, the detection capability begins to deteriorate, because the speed of ships at sea is relatively average. The high anomaly threshold is difficult to apply to accurately identify the anomaly data in a data set with a low degree of dispersion. The feature fusion method is based on a deep learning algorithm, and its detection capability depends on the appropriate super parameters and the quality of the training set. Its detection capability is not stable enough. Next, comparing the improved iForest algorithm with the traditional iForest algorithm (which adopts the strategy of selectively constructing isolated trees) from the perspective of algorithm efficiency, the results are shown in Table 3.

Table 3. The comparison of the efficiency of different methods of detection.

Method	The number of data points detected	Running time (ms)	Average detection time for single data points (ms)
IForest	246	1750	7.11
Improved iForest	246	1662	6.76

As can be seen from Table 3, the iForest algorithm takes 7.11 ms to detect a single data points, while the improved iForest algorithm takes 6.76 ms. In terms of algorithmic efficiency, the improved iForest method improves efficiency by about 5% over the traditional iForest algorithm. By analyzing the reason, it can be seen that the improved iForest algorithm adopts the strategy of selectively constructing isolated trees; when the ratio of the number of samples divided into the left sub-tree and the number of samples divided into the right sub-tree is not large, it chooses the strategy of stopping construction, so its efficiency will be better than that of the iForest algorithm.

Finally, we have carried out ablation experiments to verify the high accuracy of the iForest algorithm in detecting the abnormal speed of ships after noise removal by the MDDBSCAN algorithm. The results are shown in Figures 15 and 16 and Table 4. Meanwhile, in this experiment, the threshold used to detect whether the ship speed is abnormal was set to *score* (0.15). It can be seen from Figures 15 and 16 and Table 4 that the detection capability of MDDBSCAN- Improved iForest is better than that of Improved iForest. By analyzing the reasons, the MDDBSCAN algorithm provides a global anomaly detection scenario for the iForest algorithm, which has high accuracy on such data sets.

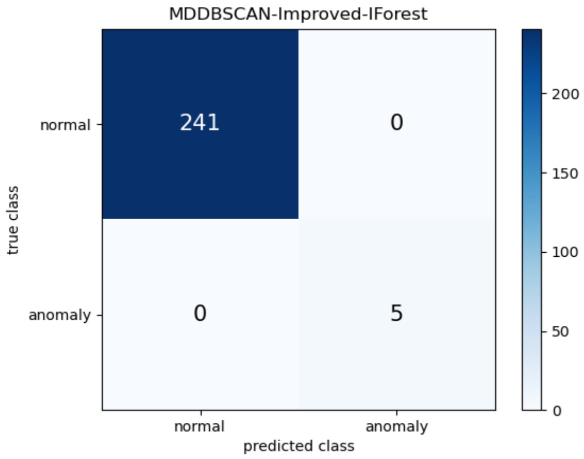


Figure 15. The result of MDDBSCAN-improved iForest.

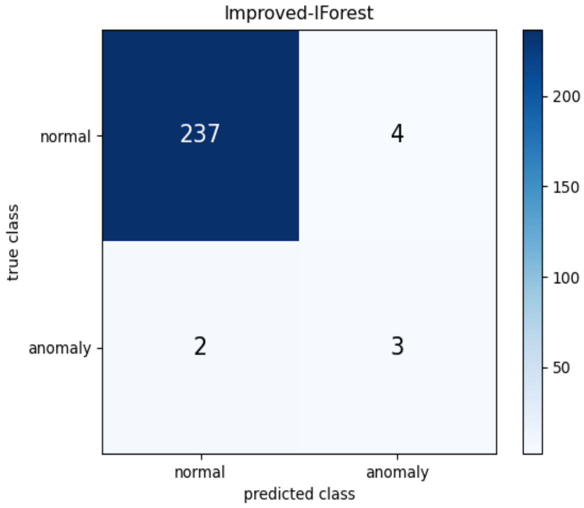


Figure 16. The result of Improved iForest.

Table 4. The comparison of Improved iForest and MDDBSCAN-improved iForest.

Method	Recall	Precision	F1	Accuracy
Improved iForest	0.9834	0.9916	0.9875	0.9756
MDDBSCAN-improved iForest	1	1	1	1

6. Conclusions

This method separates the detection of ship behavior outliers into three steps. The first part is data preprocessing and data compression, which achieves the accuracy and simplicity of describing ship trajectories. Second, the position information modeling scheme detects the ship position outliers. By comparing the five trajectory outlier detection methods, the method in this paper had a better detection effect. Finally, the isolation forest algorithm is used to detect the ship's speed outliers, and the functional relationship between the speed weight of multi-dimensional density clustering and the threshold for determining the speed as abnormal has been established. Experiments showed that the threshold selected by $score(w)$ had a good result for detecting ship speed outliers. This paper's abnormal ship behavior detection method is suitable for online detection and can also mine more abnormal ship information besides speed, such as ship acceleration, heading, etc. Meanwhile, due to the evolving computing power techniques, establishing a more efficient and accurate abnormal ship behavior detection model will also have a promising possibility.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research was funded by the Key Research and Development Program of China, grant number 2021YFC2802503; Key Research and Development Program of Shaanxi Province, grant number 2021ZDLGY05-05 and 2019ZDLGY12G07.

Conflict of interest

The authors declare no conflict of interest.

References

1. C. Claramunt, C. Ray, E. Camossi, A. Jousselme, M. Hadzagic, G. Andrienko, et al., Maritime data integration and analysis: recent progress and research challenges, in *20th International Conference on Extending Database Technology*, 2017.
2. T. Lv, C. He, J. Zhang, Z. Song, Massive AIS data storage and query based on Hadoop platform, *J. Phys. Conf. Ser.*, **1948** (2021), 012016. https://doi.org/10.1088/1742-6596/1948/1/012016
3. L. Zhang，Y. Zhu, W. Lu, J. Wen, A detection and restoration approach for vessel trajectory anomalies based on AIS, *J. Northwest. Polytech. Univ.*, **39** (2021), 119–125. https://doi.org/10.1051/jnwpu/20213910119
4. K. Wolsing, L. Roepert, J. Bauer, K. Wehrle, Anomaly detection in maritime AIS tracks: A Re-view of Recent Approaches, *J. Mar. Sci. Eng.*, **10** (2022), 112. https://doi.org/10.3390/jmse10 010112

5. C. Tian, Y. Yuan, S. Zhang, C. Lin, W. Zuo, D. Zhang, Image super-resolution with an enhanced group convolutional neural network, **153** (2022), 373–385. https://doi.org/10.1016/j.neunet.2022.06.009
6. C. Tian, Y. Zhang, W. Zuo, C. Lin, D. Zhang, Y. Yuan, A heterogeneous group CNN for image super-resolution, *IEEE Trans. Neural Netw. Learn. Syst.*, **13** (2022). https://doi: 10.1109/TNNLS.2022.3210433.
7. D. Zhang, L. Nan, Z. Zhou, C. Chen, L. Sun, S. Li, iBAT: Detecting anomalous taxi trajectories from GPS traces, in *UbiComp 2011: Ubiquitous Computing, 13th International Conference*, (2011), 99–108. https://doi.org/10.1145/2030112.2030127
8. J. Zhu, W. Jiang, A. Liu, G. Liu, L. Zhao, Time-dependent popular routes based trajectory outlier detection, in *International Conference on Web Information Systems Engineering*, **9418** (2015). https://doi.org/10.1007/978-3-319-26190-4_2
9. C. Chen, D. Zhang, P. Castro, N. Li, L. Sun, S. Li, et al., iBOAT: isolation-based online anomalous trajectory detection, *IEEE Trans. Intell. Trans. Syst.*, **14** (2013), 806–818. https://doi.org/10.1109/TITS.2013.2238531
10. J. Zhu, W. Jiang, A. Liu, G. Liu, L. Zhao, Effective and efficient trajectory outlier detection based on time-dependent popular route, *World Wide Web*, **20** (2017), 111–134. https://doi.org/10.1007/s11280-016-0400-6
11. W. Hao, W. Sun, B. Zheng, A fast trajectory outlier detection approach via driving behavior modeling, in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, (2017), 837–846. https://doi.org/10.1145/3132847.3132933
12. L.Bao, M. Du, A distance-based trajectory outlier detection method on maritime traffic data, in *2018 4th International Conference on Control, Automation and Robotics (ICCAR)*, 2018. https://doi.org/10.1109/ICCAR.2018.8384697
13. E. Martineau, J. Roy, Maritime anomaly detection: domain introduction and review of selected literature, *Defense Res. Develop. Canada*, 2011.
14. L. Portnoy, E. Eskin, S. Stolfo, Intrusion detection with unlabeled data using clustering, *ACM Workshop Data Mining Appl.*, 2001.
15. S. Zhang，Q. Tang, Abnormal vessel behavior detection based on AIS Data, *Artif. Intell. Rob. Res.*, **04** (2015), 23–31. https://doi.org/10.12677/airr.2015.44004
16. R. Lane, D. Nevell, S. Hayward, T. W. Beaney, Maritime anomaly detection and threat assessment, in *2010 13th International Conference on Information Fusion*, (2010), 1–8. https://doi.org/10.1109/ICIF.2010.5711998
17. R. Laxhammar, Anomaly detection for sea surveillance, in *International Conference on Information Fusion*, (2008), 1–8.
18. Y. Wang, J. Liu, R. Liu, Y. Liu, Z. Yuan, Data-driven methods for detection of abnormal ship behavior: Progress and trends, *Ocean Eng.*, **271** (2023). https://doi.org/10.1016/j.oceaneng.2023.113673
19. F. Luan, Y. Zhang, K. Cao, Q. Li, Based local density trajectory outlier detection with partition-and-detect framework, in *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, (2017), 1708–1714. https://doi.org/10.1109/FSKD.2017.8393023

20. B. Liang, S. Wu, W. Chen, Z. Zhu, Trajectory outlier detection based on partition-and-detection framework, in *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, 2017. https://doi.org/10.1109/FSKD.2017.8393071
21. A. Belhadi, Y. Djenouri, D. Djenouri, T. Michalak, J. C. Lin, Deep learning versus traditional solutions for group trajectory outliers, *IEEE Trans. Cybernetics,* **6** (2020), 1–12. https://doi.org/10.1109/TCYB.2020.3029338
22. M. Szarmach, I. Czarnowski, Multi-Label classification for AIS data anomaly detection using wavelet transform, *IEEE Access*, **10** (2022), 109119–109131. https://doi.org/10.1109/ACCESS.2022.3214217
23. Y. Chen, J. Yu, G. Yong, Detecting trajectory outliers based on spark, in *2017 25th International Conference on Geoinformatics*, (2017), 1–5. https://doi.org/10.1109/GEOINFORMATICS.2017.8090919
24. K. Hu, P. Duan, B. Hu, Q. Duan, IBTOD: An isolation-based method to detect outlying sub-trajectories on multi-factors, in *IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference*, 2018. https://doi.org/10.1109/IMCEC.2018.8469416
25. A. Belhadi, Y. Djenouri, C. Lin, Comparative study on trajectory outlier detection algorithms, in *2019 International Conference on Data Mining Workshops (ICDMW)*, (2019), 415–423. https://doi.org/10.1109/ICDMW.2019.00067
26. R. Maria, P. Giuliana, V. Michele, Maritime anomaly detection: A review, *Wiley Interdiscip. Rev. Data Mining Knowl. Discovery,* **8** (2018), 8. https://doi.org/10.1002/widm.1266
27. S. Papadimitriou, H. Kitagawa, P. Gibbons, C. Faloutsos, LOCI: fast outlier detection using the local correlation integral, in *Proceedings 19th International Conference on Data Engineering*, 2003, 315–326. https://doi.org/10.1109/ICDE.2003.1260802
28. G.Pallotta, M.Vespe, K.Bryan, Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction, *Entropy*, **15** (2013), 2218–2245. https://doi.org/10.3390/e15062218
29. W.Dai, C.Zhang, X.Su, S. Cao, Trajectory Outlier Detection Based on DBSCAN and Velocity Entropy, in *2020 International Conferences on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics),* (2020), 550–557. https://doi.org/10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics50389.2020.00097
30. Z. Cheng, C. Zou, J. Dong, Outlier detection using isolation forest and local outlier factor, in *Proceedings of the Conference on Research in Adaptive and Convergent Systems*, (2019), 161–168. https://doi.org/10.1145/3338840.3355641
31. F. Luan, Y. Zhang, K. Cao, Q. Li., Based local density trajectory outlier detection with partition-and-detect framework, in *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, (2017), 1708–1714. https://doi.org/10.1109/FSKD.2017.8393023
32. T. Fei, M. Kai, Z. Zhou, Isolation forest, in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, (2008), 413–422. https://doi.org/10.1109/ICDM.2008.17
33. Historical AIS Data Services (accessed on 10 December 2018). Available from: http://www.vtexplorer.com/

34. C. Iphar, C. Ray, A. Napoli, Data integrity assessment for maritime anomaly detection, *Expert Syst. Appl.*, **147** (2020), 3. https://doi.org/10.1016/j.eswa.2020.113219

35. H. Liu, Y. Liu, Z. Zong, Research on ship abnormal behavior detection method based on graph neural network, in *2022 IEEE International Conference on Mechatronics and Automation (ICMA)*, (2022), 834–838. https://doi.org/10.1109/ICMA54519.2022.9856198

36. G. Huang, S. Lai, C. Ye, H. Zhou, Ship trajectory anomaly detection based on multi-feature fusion, in *2021 IEEE International Conference on Smart Data Services (SMDS)*, (2021), 72–81. https://doi.org/10.1109/SMDS53860.2021.00020