POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

SEAquence: a Transformer-based IUU Fishing Detection from Maritime AIS Data

Laurea Magistrale in Mobility Engineering - Ingegneria della Mobilità

Author: Damiano Masuino

Advisor: Prof. Mark James Carman

Academic year: 2022-2023

1. Introduction

The sea has always been a source of opportunity, for everyone. It is simultaneously the habitat that preserves an immense multitude of plants, the home to countless fish, and the surface that has greatly facilitated the development of trade and human exploration. However, when discussing opportunities, it does not refer solely to the positive ones: its vast distances have often contributed to the emergence and evolution of various forms of illegal activities as well.
In this context, this thesis aims to investigate new methods for the classification of human marine activities, with the goal of creating a model that serves as an ally in identifying potential illegal fishing activities. The approach used was to test various strategies based on the use of state-of-the-art deep learning Transformer architectures (models known for their high performance in the field of sequential data, such as those related to language), comparing different alternatives and achieving promising results.
To classify the type of naval activity, the trajectory traced by the boat has been studied. In particular, data from AIS (Automatic Identification System) have been used to analyze the variation in the position of the vessel under study over the temporal range of analysis.

2. Related works

Transformers-based models are now widely prevalent in the field of Natural Language Processing (NLP), and there is a rich available literature. This kind of architecture was initially introduced in [7] as a model capable of revolutionizing textual analysis. However, it quickly proved to be highly versatile and capable of processing data in many different forms, such as images ([1]). This adaptability of Transformer algorithms has been effectively leveraged in this thesis.

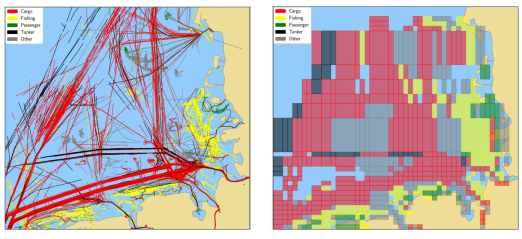

Figure 1: Trajectories of different vessel types (left image) and the result of trajectory clustering (right image), *image from [3].*

In the past, several attempts have been made to classify naval activities, using different data sources and strategies. [3], is an example where the trajectory itself is not considered as a geometric entity, but rather the focus is placed on

the spatial arrangement of the vessels (see *Figure 1*).
Geographical areas have been associated with the presence of a particular type of vessel. The classification is done by spatializing the vessel and incorporating additional information. For example, if a large-sized vessel (additional information required) is located in the area associated with high tanker traffic (the black region on the map), then the model classifies it as a tanker. Although the results are encouraging, this model requires detailed data such as the tonnage of the vessel.
By searching for ideas outside of the maritime domain, [2] proposes an innovative method (based on [4], [5], and [6]) for trajectory classification that is not based on the sequence of coordinates but on the creation of representative images, which are then used as input to a CNN model. The trajectories are spatially discretized, transforming them into pixels of a 2-dimensional image (see *Figure 2*).

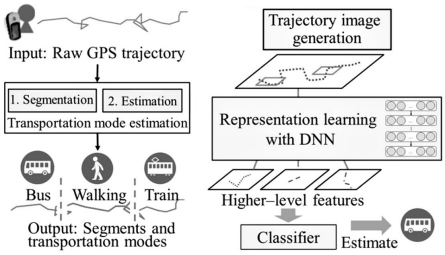

Figure 2: The scheme of the process used in the paper *image from [2].*

The study analyzes trajectories found in a city and aims to classify them based on the mode of transportation (such as bicycles, cars, subways, etc.). This paper can be seen as the missing link among all the others. So, why not use Vision Transformers (ViT) to study naval trajectories in a similar way to what is presented in this paper? This question, along with the points listed above, serves as the starting point for this study.

3. Research questions

After researching the state of the art, it became clear that there are no studies focused on the development of Transformer-based classification models. The purpose of this thesis is to shed light on these still poorly explored fields, finding in particular an answer to the following main research questions:

What is the effectiveness and potential of using ViT models for the detection of IUU activities?

ViT (Vision Transformer) models excel at capturing spatial relationships and long-range dependencies in images. This potential can be harnessed for detecting fishing activities, especially illegal, unreported, and unregulated (IUU) fishing, by analyzing AIS satellite data. ViT models can learn to recognize patterns, vessel types, or behaviors associated with IUU activities, enhancing monitoring and enforcement efforts.

What are the most effective strategies?

What emerges as the optimal strategy for studying trajectories? Should textual models be employed, treating trajectories as text sequences, or should images of the trajectories be utilized instead? Moreover, what information is advisable to incorporate in the inputs (e.g., the amount of information to be depicted on the images)?

Is the pre-training dataset always so important?

The need for enormous datasets to train the model often translates to high costs and limited practical usability. However, the question arises: How far is it possible to push the versatility of these models?

4. Data

The primary data source for this thesis is the AIS (Automatic Identification System), a system known for its uniqueness in data exchange among various entities, each with distinct purposes and modes of communication. AIS data includes information about a vessel's position and direction, transmitted via VHF systems and interpreted by AIS transponders. Different classes of AIS instruments exist, such as Class A, Class B, and Class C, each with varying functionalities and transmission frequencies. The relevant data for this study revolves around ship identification, type, position, and transmission time. This focus ensures that the model relies

on readily available information, minimizing the reliance on specific data that may be incomplete or absent.
AIS data is employed by multiple users, including vessels themselves for enhancing situational awareness, data providers for commercial purposes, and port authorities for managing and monitoring coastal traffic. However, access to AIS data via satellite equipment is typically restricted to private data providers due to the high costs involved. For this thesis, data obtained from the eastern coast of the United States, a region known for its dense fishing activity, has been used.
This area was selected due to its relevance in fisheries, as compared to other less active maritime zones. The dataset comprises both static and dynamic information, and it exhibits a slight imbalance, with approximately 22.9% of the training data and 18.7% of the testing data related to fishing activities.
The table below summarises the characteristics of the data used in this thesis, from the geographic and temporal perspectives.

Detail	Value
Longitude min	34.18621
Longitude max	43.40141
Latitude min	-75.94461
Latitude max	-65.05574
TRAIN Data Time Range	from 2019-09-03 to 2019-12-07
TEST Data Time Range	from 2018-12-31 to 2019-01-17

Table 1: Dataset description.

5. Models & Experiments

This section will provide a description of all the methodologies used in this thesis. It is divided into 3 main sections: the pre-processing phase, where raw AIS data was converted into useful trajectories; the description of the first test done using a classic Transformer architecture and studying maritime trajectories as sequences of characters; the description of the application of ViT models to classify naval activities.

5.1. Pre-processing

This initial phase is common to both approaches used for the actual classification. AIS data appears as long sequences of rows, where each row represents an AIS signal. It is indeed possible to represent the initial AIS data as a set of spatially scattered points. Each point is associated with a signal from a vessel, but the trajectories and behaviors of individual ships are not clear at this stage. After removing any items with NaN values in the features related to position, identification, and type, the first step was to simplify the vast amount of data as much as possible.
It is easy to notice how the human eye can almost instantly identify different trajectories among the cluster of points. However, it is a completely different challenge for an algorithm to achieve the same task. To tackle this, a decision was made to leverage the study of various anchorage areas to create a robust method that autonomously identifies different trajectories.
Vessels within an anchorage area may have varying degrees of movement, leading to irregular patterns in the clustered data. Some ships may remain relatively stationary with slight drift, while others might experience more significant shifts due to currents or other factors.
To study this phenomenon, the Minimal Enclosing Circle (MEC) algorithm was chosen as the primary tool. The Minimum Enclosing Circle (MEC) algorithm is a geometric algorithm used to find the smallest circle that encloses a given set of points in a two-dimensional plane. The algorithm's basic idea revolves around finding the center and radius of the circle that encloses the points while ensuring no point lies outside the circle. For each point, the following five points were considered, and the Minimum Enclosing Circle (MEC) was calculated.

Figure 3: Cluster of points related to an anchoring phase in dark blue.

A threshold of 75 meters was used to determine

whether that point was related to a navigation phase or if the ship was anchored at that point. In other words, if a ship emits a series of AIS signals close to each other (such that the circle enclosing five consecutive signals has a radius smaller than the 75-meter limit), it is assumed that the ship is anchored and making minimal movements (see *Figure 3*).
Within the sub-datasets, the points were then divided between those related to navigation phases and those related to anchoring. For further simplification, consecutive points related to anchoring were condensed into a single point whose position was the average of the positions of the AIS signals where the ship was anchored. After defining the anchoring points, defining the trajectories becomes straightforward. The trajectories are the lines described by all consecutive points that lie between two anchoring points. The algorithm identifies these trajectories and assigns a unique identifier to each of them.

5.2. Trajectory as a sentence

As previously mentioned, Transformer architectures are considered state-of-the-art when it comes to natural language processing. As the first methodology to use in identifying fishing activities, it was decided to apply them in their "natural habitat" by transforming trajectories into sentences.
Of course, it is not possible to do this using commonly used words, so spatial discretization has been exploited to describe trajectories not as a sequence of spatial coordinates but rather as a sequence of simple numerical characters.

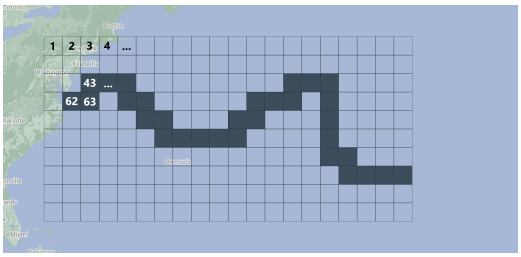

Figure 4: The output is simply the sequence of cell IDs.

Each trajectory was considered by discretizing the space it extends over. Each portion of discretized space was associated with a unique number. Therefore, the trajectory was described by a sequence of unique numbers referring to the sections of space it passes through (see *Figure 4*).
Regarding the model related to trajectory classification as textual sequences, it can be analyzed and described by breaking it down into 7 main blocks.

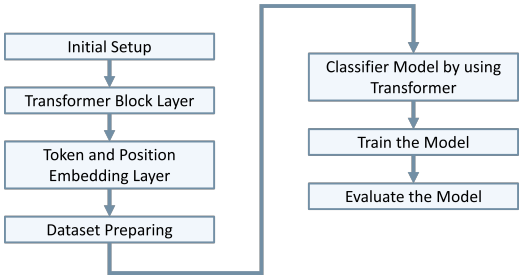

Figure 5: Sequence classification model main steps.

In the initial setup section, the code imports all the necessary libraries, including Keras and TensorFlow. Then, the definition of the transformer block (named "*TransformerBlock*") is performed. At this stage, a custom Keras layer is defined, with the goal of representing the transformer architecture. The third step in the code is the implementation of the "*TokenAndPositionEmbedding*" class, which serves as a custom Keras layer for embedding the input text data (i.e., the trajectories) as tokens and their corresponding positions. The output of the layer is the sum of token and position embeddings, effectively incorporating both the spatial information (positions) and semantic information (tokens) into the continuous vector representations of the input text data. In the fourth step, the dataset is loaded and the fifth stage is where the model architecture is defined. The input sequences are passed through the custom *TokenAndPositionEmbedding* prepared in the second step. Then, the data is passed through the *TransformerBlock* layer. The final output from the *TransformerBlock* is averaged across all time steps using a particular pooling layer (specifically called "*GlobalAveragePooling1D*"), followed by a couple of Dense layers to classify the text into two classes (positive or negative sentiment). The two classes are referred to as fishing or not-fishing maritime activities. The fifth and sixth points are where the actual model defined in the previous points is used. The model is compiled with an optimizer, loss

function, and evaluation metric. The code proceeds to train the model on the training data and evaluate its performance on the validation data. The training is done for 10 epochs with a batch size of 32, but further details will be discussed in the following chapter.

5.3. Trajectory as a picture

The research direction shifted towards this approach due to the less-than-ideal results obtained in the first methodology. It was hypothesized that leveraging the Transformer's ability to capture shape features in trajectory images could potentially improve the classification performance. In contrast to the first approach, in this case, the study explored various trajectory image creation techniques. Another difference is that a pre-trained version of ViT was here used (instead of a non-pre-trained Transformer). The decision to use a pre-trained model offered a practical and efficient solution to achieve favorable results and it was driven by two primary reasons:

1. The time and resources required to train a Vision transformer model from scratch were prohibitive (several times the data required to train a vanilla transformer).
2. By utilizing a pre-trained model that had been trained on diverse datasets, potentially dissimilar from the data used in this particular study, researchers had the opportunity to assess the adaptability and versatility of these models.

Four different approaches were used to create images of the trajectories from AIS data. The first approach was to leverage the spatial discretization described earlier, which was used to create textual sequences, for generating images as well.

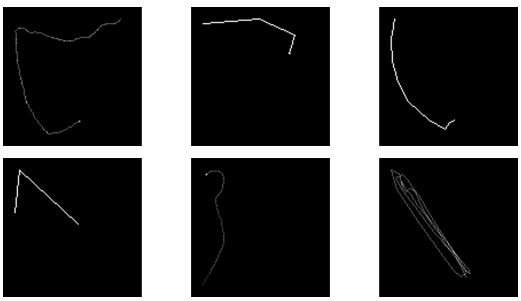

Figure 6: First image synthesis approach.

In essence, the groundwork was already in place: instead of extrapolating a sequence of unique numbers, the approach involved saving an image depicting the spatial sections through which the trajectory passed.
After observing a significant improvement in the results (as discussed in the subsequent chapter), the decision was made to test additional variations of trajectory images.

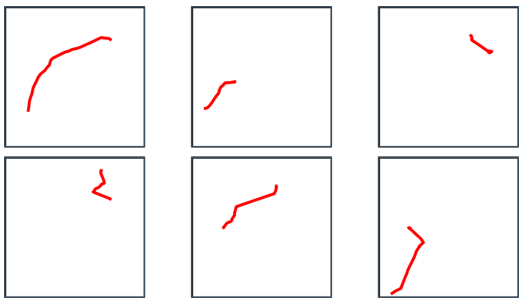

Figure 7: Second image synthesis approach.

The first evolution involved a change in the zoom perspective. Instead of focusing on the trajectory's shape, the attention shifted toward the ship's geographic position. In the second graphical approach, images were created with a constant scale and geographic center for all trajectories. From this point onward, no spatial discretization is used. By maintaining a consistent scale and center across all images, the shape of the trajectory was no longer emphasized, particularly for shorter trajectories. However, this approach provided valuable information about the geographic location of the trajectory.
The third set of images created and used is, in fact, the same as described above in the second point. The trajectories are the same, and the methodology is identical, with the only difference being the addition of a map as the background.

Figure 8: Third image synthesis approach.

The fourth approach represents a further evo-

lution in terms of the amount of information included. Multiple indicators have been added for both the starting and ending points of the trajectory. Additionally, the trajectory's color representation is no longer constant; instead, it is depicted using a range of tones based on the normalized velocity of the ship.

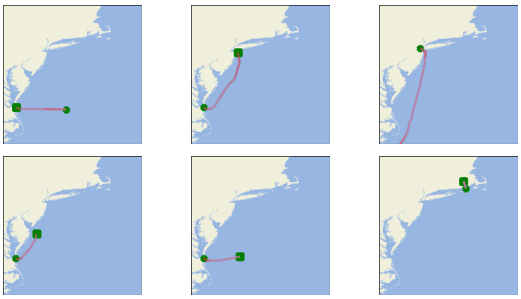

Figure 9: Fourth image synthesis approach.

Following a similar process to what was done for the analysis of the model used for sequence classification, the model used here has been divided into steps (see *Figure 10*).
Similarly to the previous approach, the first step involves initializing the actual model, and all the required libraries for the model are loaded. In the second step, the data used for image classification is loaded. During this stage, the different datasets created were loaded one by one to evaluate their performance from a classification perspective. In the third stage, the script defines image transformations using "*transforms.Compose*" from *torchvision* library. The transformations include converting images to tensors, resizing them to (224, 224) pixels, and normalizing the pixel values. This step is required to use the images as input for the transformer layers.

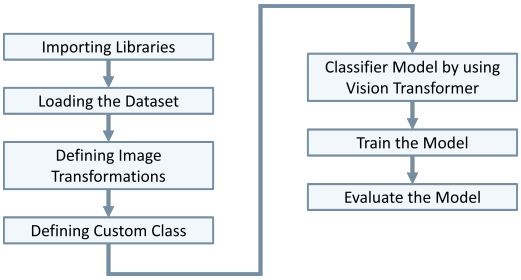

Figure 10: Image classification model main steps.

In the fourth step, the script defines a custom dataset class named *ImageDataset*, which is tailored for the ViT model. It inherits from *torch.utils.data.Dataset* and provides methods to access images and their corresponding labels. In the fifth step, the code defines a custom PyTorch module for the Vision Transformer (ViT) model. The custom module is called *ViT*, and it extends the functionality of the *ViTModel* class provided by the Hugging Face's transformers library. The *ViT* class combines the Vision Transformer model with a linear classifier for multi-class classification tasks. The last steps are quite similar to the ones referred to in the sequences analysis. In them, the code loads the model, loss function (cross-entropy), and optimizer (Adam). Training data is then loaded using the *ImageDataset* class and then fed into the model in batches. The model is trained for a specified number of epochs (10), and the loss and accuracy are printed at the end of each epoch.

6. Results

The results of the classification of trajectories as text sequences are the following:

Metric	Value
Accuracy	0.8124
PPV	0.0037
TPR	0.3590
F1 Score	0.0073

Table 2: Metrics from 1st approach.

The results from the image classification strategy (four different images of trajectories) are the following:

Metric	Value
Accuracy 1	0.7440
PPV 1	0.3301
TPR 1	0.3207
F1 Score 1	0.3253
Accuracy 2	0.9520
PPV 2	0.8089
TPR 2	0.9252
F1 Score 2	0.8632

Accuracy 3	0.9550
PPV 3	0.8200
TPR 3	0.9312
F1 Score 3	0.8721
Accuracy 4	0.9538
PPV 4	0.8679
TPR 4	0.8831
F1 Score 4	0.8755

Table 3: Metrics from 2nd approach.

7. Conclusions

Transformer architectures have demonstrated remarkable capabilities in natural language processing tasks, image recognition, and various other domains. The researchers hypothesized that these architectures could be adapted and fine-tuned for marine activity classification using Automatic Identification System (AIS) data.

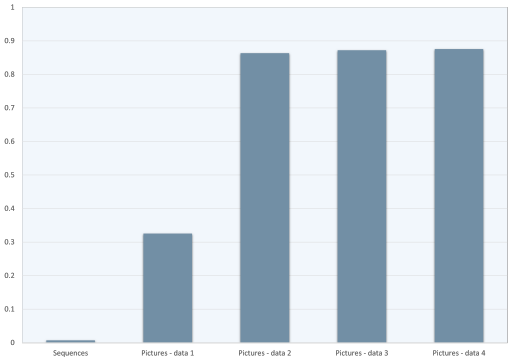

Figure 11: F1 scores comparison.

In the graph above, it is posible to see a comparison of F1 scores for different models used. The first one on the left refers to the classifier model utilizing sequences, while the others are related to trajectory classifications using the produced images (with their four corresponding strategies).
Evaluating the different scores, it is evident that studying trajectories as images yields significantly better results compared to studying naval routes as sequences. Another clear conclusion is that the classification of fixed-scale and zoomed images (focusing on geographical dispersion and not just the shape of the trajectory) promises better results than other alternatives. The quantity of information contained in each image does not seem to significantly impact the performance. The Transformer architectures, particularly the Vision Transformer (ViT), demonstrated versatility. Excellent classification results were achieved even when starting from pre-trained models on different images. This highlights the immense potential of an ideal model trained solely on a massive number of naval scenarios.
In summary, using images to study trajectories and leveraging the power of Transformer architectures, especially ViT, holds great promise for improving the classification of maritime activities.

7.1. Further Developments

From a future perspective, several further advancements can be made in this area.

- A more in-depth study of the type of information to include in the images could lead to new insights. While this study did not find information that significantly improved classification, it does not mean that such information does not exist.
- There is potential for studying how this model could be used for classifying other types of maritime activities or diving into more detailed classifications, such as different types of fishing practices.
- Developing a system that integrates real-time maritime activity on a single dashboard, similar to the live map available on some websites, along with indications of activities classified as suspicious by the model (false negatives), would be highly beneficial.

References

[1] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR 2021*, 2020.

[2] Yuki Endo. Classifying spatial trajectories using representation learning. *Int J Data Sci Anal 2016*, 2016.

[3] Paul Kraus. Ship classification based on trajectory data with machine-learning methods. *The 19th International Radar Symposium IRS 2018*, 2018.

[4] Hongbin Liu. End-to-end trajectory trans-

portation mode classification using bi-lstm recurrent neural network. *12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*, 2017.

[5] Hongbin Liu. Spatio-temporal gru for trajectory classification. *2019 IEEE International Conference on Data Mining (ICDM)*, 2018.

[6] Asif Navaz. Convolutional lstm based transportation mode learning from raw gps trajectories. *IET Intelligent Transport Systems*, 2020.

[7] Ashish Vaswani. Attention is all you need. *31st Conference on Neural Information Processing Systems*, 2017.