A Heuristic ETL Process to Dynamically Separate and Compress AIS Data

Atefe Sedaghat*, Masood Jafari Kang*, Maryam Hamidi*

Abstract— **Massive vessel trajectory data can be obtained from marine Automatic Identification Systems (AIS) to extract information about water traffic. To efficiently collect and process such a huge amount of data special methods are needed. This study designs a new system for collecting and processing AIS data in a real-time manner. The proposed system not only compresses vessel data while keeping useful information but also adds more attributes to raw trajectory data. The additional attributes include trip id, trip origin/destination, traffic density, and traffic flow. At first, this study presents a dynamic Extract, Transform, and Load (ETL) pipeline that collects AIS messages from vessels, processes those raw data, and loads the processed data in a central database. An optimized algorithm is developed that can process millions of records as fast as possible and send the processed data to production. Next, a user interface is developed to quantify traffic conditions and visualize them in graphs and maps. Finally, Gulf Intercoastal Waterway (GIWW) is considered as study area, where historical and real-time AIS data located in GIWW were collected to test the functionality of the method.**

I. Introduction

The Intracoastal Waterway refers to a series of inland waterways that run along the eastern and southeastern coasts of the United States, connecting numerous bays, estuaries, and rivers [1]. The waterway serves as a vital transportation route for commercial vessels. Due to its complex and often narrow channels, navigating the waterway can be challenging which requires a high level of skill and experience. However, the use of the Automatic Identification System (AIS) has become increasingly common among vessel operators to improve safety and efficiency in these waters. AIS is an automated tracking system that uses transponders on vessels to broadcast their location, speed, and other information to other ships and to shore-based authorities. This information can be used to help prevent collisions, optimize vessel routes, and monitor vessel traffic in real time.

Despite the benefits of AIS, the sheer volume of data generated by vessels can be overwhelming and difficult to process, making it challenging to extract useful insights. Traditional methods of data processing are often time-consuming and require manual effort, which is not feasible for real-time monitoring of vessel traffic. Moreover, raw AIS data often contains noise and false data that needs to be filtered out before it can be used effectively. As a result, there is a need for an efficient algorithm that can dynamically process AIS data and provide useful insights into vessel traffic. To address this need, we propose an online monitoring vessel traffic system that utilizes an Extract, Transform, and Load (ETL) pipeline to dynamically extract the stream data of AIS messages and process them into meaningful information. Our algorithm processes information such as online traffic volume, origin/destination, vessel trips, trip direction, and vessel routing segments. The processed data is then stored in a database and made available for visualization in a user interface tool. This allows users to query the database and retrieve information on the online trip of a specific vessel in a day, displayed on a real-time map.

In this paper, we focus on applying our algorithm to the Intracoastal Waterway region, however, the algorithm is applicable to other regions in the maritime network. We obtained historical AIS data for this region for two years and simulated an API to generate a stream of data for testing purposes. Our results demonstrate the effectiveness of our algorithm in processing AIS data in real time and providing useful insights into vessel traffic in the region.

The paper is organized as follows. Processing based on historical AIS data (offline mode) and both historical and current AIS data (online mode) are discussed in Section II. The methodology of the paper including ETL process is presented in Section III. In Section IV, the result using a dashboard is visualized and finally, the conclusion and future works are presented in Section V.

II. Literature review

In this section, we aim to review the various papers that have utilized historical AIS data for vessel tracking and trajectory analysis. Our concentration is on the implementation of online monitoring and intelligent frameworks that dynamically obtain and compare real-time AIS data with historical data to analyze vessel movements and identify trajectories.

A. Processing based on historical AIS data (offline mode)

Li et al discussed a multi-step algorithm for clustering AIS trajectories, which combines Dynamic Time Warping (DTW), Principal Component Analysis (PCA), and an improved center clustering algorithm. They extracted the characteristics of AIS data by trajectory clustering to reduce the dimension of AIS data, then found customary routes and discern the abnormal trajectories [2]. Zhang et al proposed an approach using data-driven algorithms to infer the route, starting with a simplification algorithm to compress redundant information in the original trajectories for efficiency. Turning points are then

**Department of Industrial and Systems Engineering, Lamar University, Beaumont, Texas 77710*

identified based on changes in the direction of the simplified trajectory, and clustered using the DBSCAN algorithm according to spatial proximity and turn similarity. ACO (Ant Colony Optimization) is then used to find the optimal path from the starting turning node to the ending turning node, based on the linkage between turning nodes [3]. Ren et al proposed a network based on a multi-clustering algorithm that combined k-means, DBSCAN, and affinity propagation (AP) clustering methods. The multi-clustering algorithm is designed to partition large-scale datasets into several sub-clusters, which are then merged to generate high-dimensional trajectories. The authors also proposed a distance metric to measure the similarity between trajectories [4]. Eljabu et al highlight the importance of developing automatic methods for extracting traffic routes from AIS data, given the increasing volume of data that is becoming available. The proposed method demonstrates the potential of density-based clustering algorithms for this task and highlights the importance of considering both spatial and temporal proximity when clustering AIS data [5].

Kang et al analyzed AIS data from the Houston Ship Channel for two years to study vessel congestion in narrow waterways. It focuses on a high-traffic, narrow section and examines vessel traffic patterns, speed, and factors contributing to congestion like vessel size and arrival timing [6]. Kabir et al developed a methodological framework and solution algorithms to capture significant directional changes of a ship's trajectory. By doing so, they were able to reduce the number of data points needed for vessel's U-turn analysis needed for maritime traffic management and channel development [7]. Zohoori et al presented a vectorized algorithm for analyzing waterway traffic characteristics based on AIS data. The algorithm included a nested loop algorithm to segment the waterway and separate vessel trips into inbound, outbound, and stop status. The vectorized algorithm was developed to extract traffic features such as travel speed, traffic density, traffic flow, trip attraction, trip generation, and origin-destination matrices. The results show that the vectorized algorithm significantly decreases processing time compared to loop-based methods [8].

Wu et al proposed an AIS-based method to identify hot spots in a waterway that experienced high frequencies of vessel conflicts between large vessels and those carrying hazardous materials. The paper also examined the impact of time-of-day on the frequency of vessel conflicts at each hot spot [9]. In another research by this author, his paper proposes a method to study the travel behavior of vessels passing through hotspots in narrow waterways using AIS data. The method involves building trip information for all trips in a hotspot, including inbound, outbound, and moored trips, which have stopped in a hotspot. By excluding the moored trips, the proposed method can reveal more accurate behavior patterns, especially in terms of speed distributions. The study investigates the flow speeds and densities of inbound and outbound trips passing through these hotspots, taking into account the different behaviors of various types of vessels [10]. Zohoori et al proposed an algorithm for modeling and quantifying delays caused by beam restrictions in narrow waterways. The authors also proposed procedures for determining parameters such as destination docks, vessel arrival and departure times. The algorithm is applied to three sections of the Houston Ship Channel, showing the number of impacted vessels and delays for each section. The proposed model and algorithm can be useful in various studies, including vessel scheduling, vessel ordering optimization, and expansion projects [11].

B. Processing based on historical and current AIS data (online mode)

Evmides et al proposed an intelligent framework for vessel traffic monitoring that incorporates data analytics, machine learning, and visualization techniques. The framework consists of several modules, including data acquisition, data processing and analysis, prediction and decision-making, and visualization and reporting [12]. Chi et al proposed a framework that offers a promising approach to monitoring vessel efficiency in real-time using AIS data [13]. The framework has the potential to assist in reducing greenhouse gas emissions and optimizing vessel operations, leading to cost savings and environmental benefits. Zhang and Li proposed a methodology that involves four steps: online cleaning, compression, partition, and clustering of AIS data. The data is first cleaned to remove noise and errors, then compressed to reduce the size of the dataset. Next, the data is partitioned into smaller subsets, and the clustering algorithms are applied to identify traffic patterns and anomalies [14]. Kontopoulos et al offers a promising approach for detecting intentional AIS switch-off in real-time in maritime transportation. The approach has the potential to improve safety by identifying vessels that may be attempting to avoid detection. The approach involves comparing the current AIS data to historical data to identify sudden changes or gaps in the data that may indicate an intentional switch-off. Machine learning algorithms are applied to the data to improve the accuracy of anomaly detection [15]. Mobtahej et al worked on some research in anomaly detection using deep learning algorithms [16]. Gao and Shai proposed a ship spatiotemporal key feature point (KFP) online extraction algorithm for AIS trajectory data. The algorithm uses a modified sliding window approach to consider the navigation angle deviation, position deviation, and spatiotemporal characteristics of AIS data. The proposed algorithm's performance is compared with the Douglas-Peucker (DP) algorithm in terms of feature extraction accuracy and operational efficiency. The results show that the proposed algorithm can quickly and easily extract KFPs from AIS trajectory data, benefiting ship traffic flow and navigational behavior learning [17].

After conducting a comprehensive review of the existing literature on historical analysis of AIS data and online vessel monitoring, we identified several gaps that require further research and we aimed to address them in our paper. Specifically, we found there is a noticeable lack of studies and papers proposing a replacement system for AIS data or suggesting enhancements to add more features to the AIS data. There is also a general lack of research on online traffic data analyzing and monitoring. Furthermore, we have observed that the existing papers that attempt to reduce the size of data, especially using clustering methods, are not efficient, as they

take too much time to process. Additionally, many of the papers are based on case studies and provide case-based features, rather than a general methodology that can be applied more broadly. Finally, we found no studies or methods dealing with missing data when either the transporter is off or there are noises.

III. Methodology

Extract, Transform, Load (ETL) pipeline is a series of processes that are used to extract data from various sources, transform the data into a format that is suitable for analysis, and load the transformed data into a target database or data warehouse. The ETL process typically begins with the extraction of raw data from multiple sources, such as databases, APIs, flat files, and spreadsheets. The extracted data is then transformed and cleaned to ensure that it is accurate, complete, and consistent. This includes removing duplicates, dealing with missing values, and correcting formatting issues. The transformed data is then loaded into a target database or data warehouse, where it can be queried and analyzed using various reporting and visualization tools [18].

In this paper, an ETL pipeline is developed to receive a stream of AIS messages dynamically and filter the data for the area of interest (AoI) as shown in fig. 1. The filtered stream of AIS data is then sent to the Transform component of the pipeline. In this component, various processes are conducted to determine the vessel's direction, trip segments, trip number, and tour number. The processed data is then transformed and loaded into a local database.

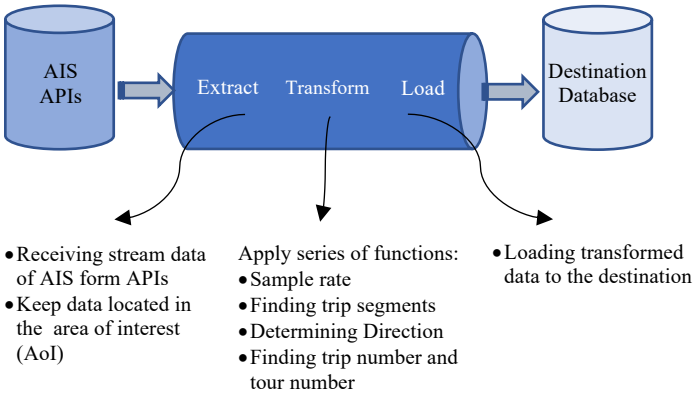

Fig 1. AIS ETL pipeline

Fig. 2 elaborates more on the Transform component and all processing steps which are applied to raw AIS record

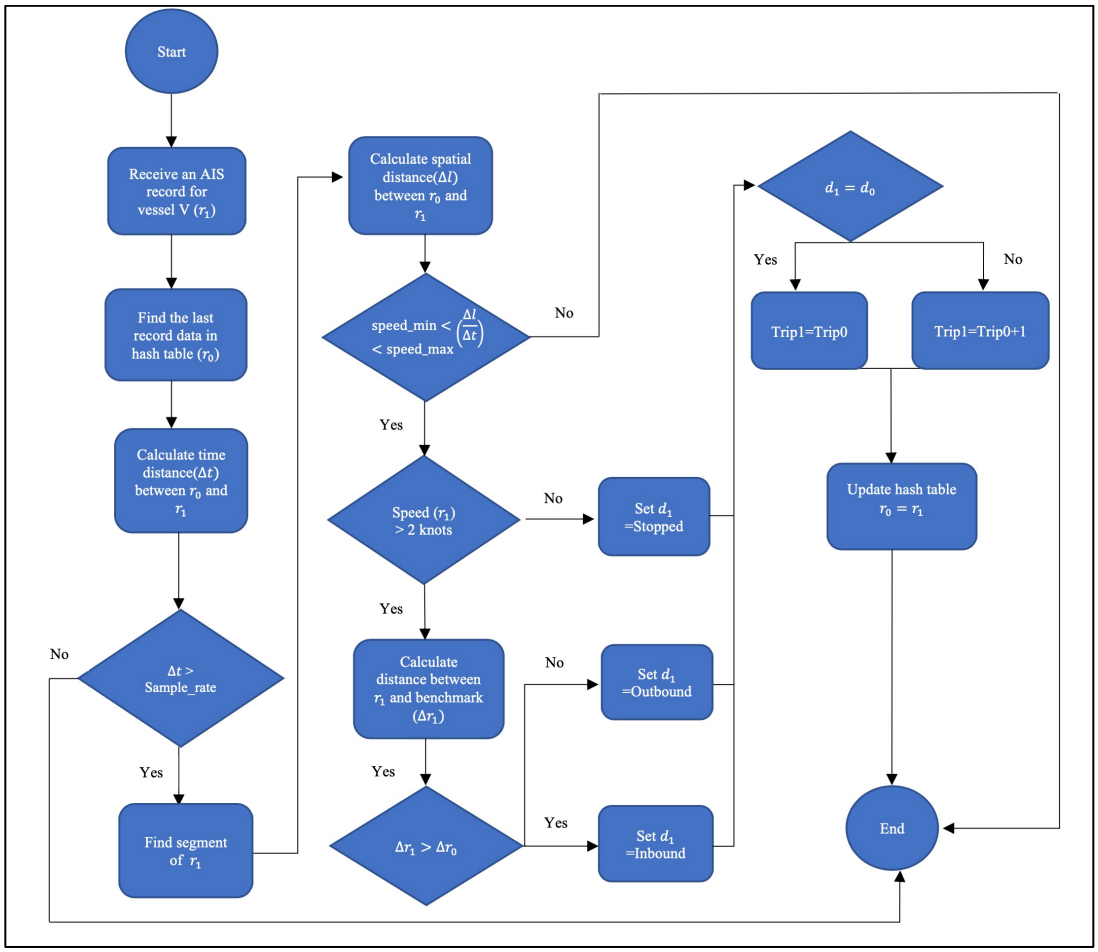

Fig 2. Flow chart of the Transform part of ETL pipeline

The proposed algorithm is based on comparing each record with its previous message, and thus, we utilize a hash table to store the last AIS records of each vessel upon receiving the stream of data. First, the algorithm checks if it has the last record of a given vessel. If there is no record for that vessel, the process is stopped, and the record will be added to the most recent AIS records. Otherwise, the time difference between the current record and the previous record is calculated, and if this difference is less than a pre-defined sample rate, the process is stopped, otherwise the record is processed further. The spatial distance between the new and previous records is then calculated to determine vessel's speed. This metric would help to find noises. If the vessel speed is greater than regular vessel speed, it can be assumed that there is a jump in trajectory data, and therefore, filtering out that record. If the speed is less than two knots, the vessel direction is considered to be "Stopped". Otherwise, the pipeline checks the distance between the new record and a pre-defined benchmark location (it is a fixed location which is considered far from the AoI to be used for defining vessel directions). A function is defined to calculate the distance between the new record and the benchmark (Δr_1) and the distance between the previous record and the benchmark (Δr_0). If $\Delta r_1 > \Delta r_0$, the vessel's direction is considered "Inbound", otherwise the vessel's direction is considered "Outbound." Knowing each record direction, we can compare current record direction with previous record to find trip numbers. If the directions are the same, the trip number does not change. Otherwise, one trip is added to the previous vessel trip number. The processed data is then loaded into a local database, including static tables for segments, vessel direction, vessel status, and vessel type, as well as dynamic tables for trips and vessel profiles.

IV. Application

In this study, we utilize the Gulf Intracoastal Waterway (GIWW), which encompasses the states along the Gulf from Texas to Florida, as the scope for our analysis. However, it is important to note that the algorithm developed for this study is applicable to any scope within the maritime network. To obtain historical data for the GIWW region, we used AIS data for a period of two years. As for the stream of data, we opt to simulate an API to generate it, rather than purchasing it from a provider. We execute the proposed ETL pipeline to process and load the data into the database. As shown in Fig 3, the pipeline successfully processed and loaded a significant number of records into the database.

To visualize our results, we implement it on an online dashboard that allowed us to analyze and explore the data further dynamically. The implementation of the dashboard provides a user-friendly interface that facilitated our analysis of the maritime data, enabling us to draw meaningful conclusions and gain deeper insights into vessel movement tracking and behavior within the GIWW region. As illustrated in Fig 4, we are able to narrow down our analysis to a specific vessel on a particular day, which enables us to reach to the vessel's information movement in a real time manner, extract valuable insights such as the vessel's trip, origin and destination, as well as the inbound and outbound direction. In Fig 5, we can customize the date, vessel type and direction to figure out the traffic flow and finally in Fig 6, we know how many trips have been generated by a specific type of vessel.

```
08:57:23: loading data from API 2018-01-02T00:00:00 to 2018-01-02T00:05:00 ...
08:57:24: 26935 records are loaded from API.
08:57:24: getting data in AoI...
08:57:24: 3522 records are located in AoI.
08:57:24: cleaning data...
08:57:24: 3390 records are filtered as cleaned data.
08:57:24: transforming data...
08:57:32: 583 records are processed.
08:57:32: Inserting new vessel data...
08:57:32: 0 vessels were added to the database successfully.
08:57:32: Inserting new trip data...
08:57:33: 583 records were added to the database successfully.
08:57:33: Inserting last records...
08:57:34: 1166 records were added to the database successfully.
08:58:34: loading data from API 2018-01-02T00:05:00 to 2018-01-02T00:10:00 ...
08:58:34: 23892 records are loaded from API.
08:58:34: getting data in AoI...
08:58:34: 3212 records are located in AoI.
08:58:34: cleaning data...
08:58:34: 3093 records are filtered as cleaned data.
08:58:34: transforming data...
08:58:43: 543 records are processed.
08:58:43: Inserting new vessel data...
08:58:43: 0 vessels were added to the database successfully.
08:58:43: Inserting new trip data...
08:58:44: 543 records added to the database successfully.
08:58:44: Inserting last records...
08:58:44: 1086 records added to the database successfully.
```

Fig 3. ETL pipeline log messages for a couple of hours

The proposed algorithm is highly efficient, processing data at a rate of 0.000001 seconds per record. For example, if we want to process one month of data (10 million records), it will take only 10 seconds using our algorithm. This speed is a significant improvement over traditional methods of data processing, which enables real-time monitoring of vessel traffic.

V. Conclusion

We present a new system for collecting and processing AIS data in real-time, using a dynamic ETL pipeline and an optimized algorithm that can process millions of records as fast as possible. Our system not only compresses vessel data while retaining useful information, but also adds additional features such as trip ID, trip origin/destination, traffic density, and traffic flow. Therefore, we propose a new collection system along with a new database schema which can be used based on AIS data. We demonstrated the effectiveness of our algorithm by applying it to historical and simulated AIS APIs for GIWW region. It is worth mentioning that our algorithm is applicable to any scope in maritime network. The algorithm is fast meaning that it can process millions of data in a few seconds This speed is a significant improvement over traditional methods of data processing and enables real-time monitoring of vessel traffic. In addition, we visualized the vessel movements on a real-time map using a user interface tool. The user interface provides real-time information on traffic conditions, such as online traffic density and traffic flow, through graphs and maps. This information is especially valuable for port authorities, who can use it to manage vessel traffic and optimize port operations. The ability to visualize data in real-time can significantly benefit other stakeholders in the maritime industry as well. It allows them to monitor vessel traffic and quickly respond to any issues that may arise, such as congestion or accidents.

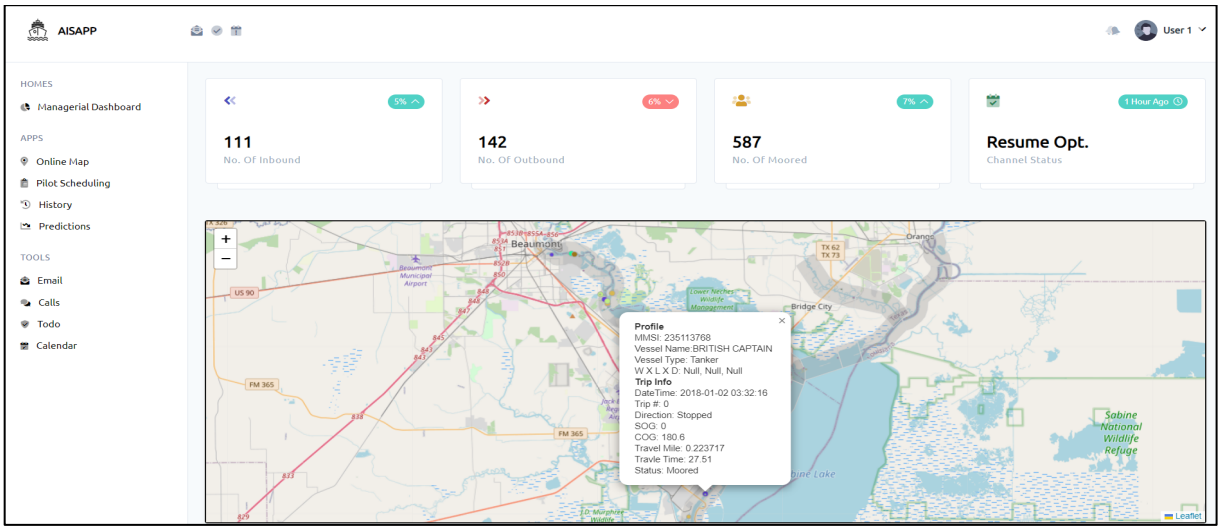

Fig 4. Live map

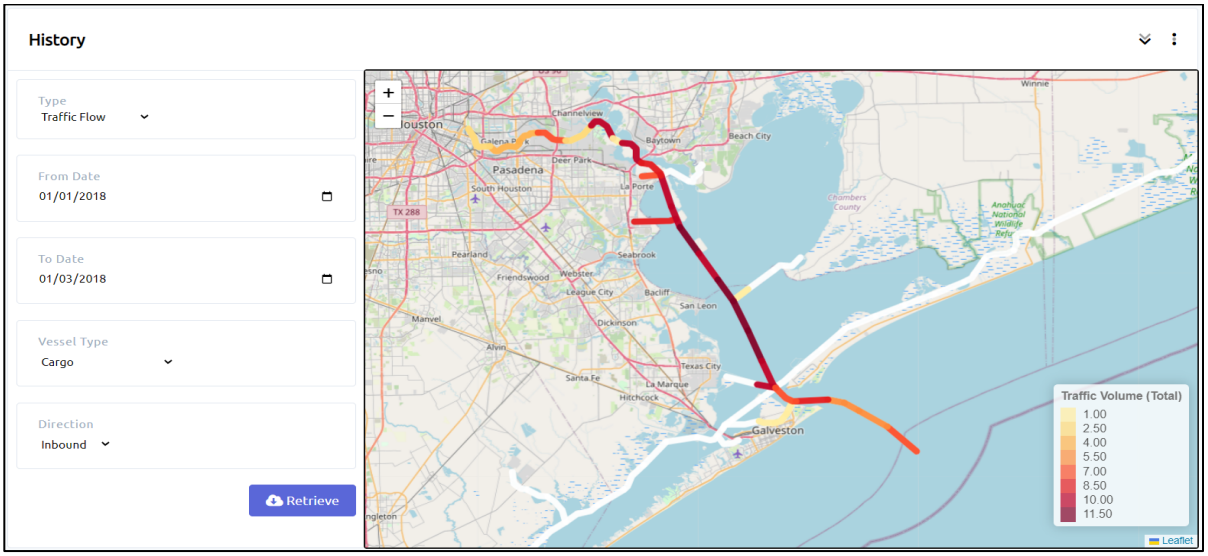

Fig 5. Traffic volume

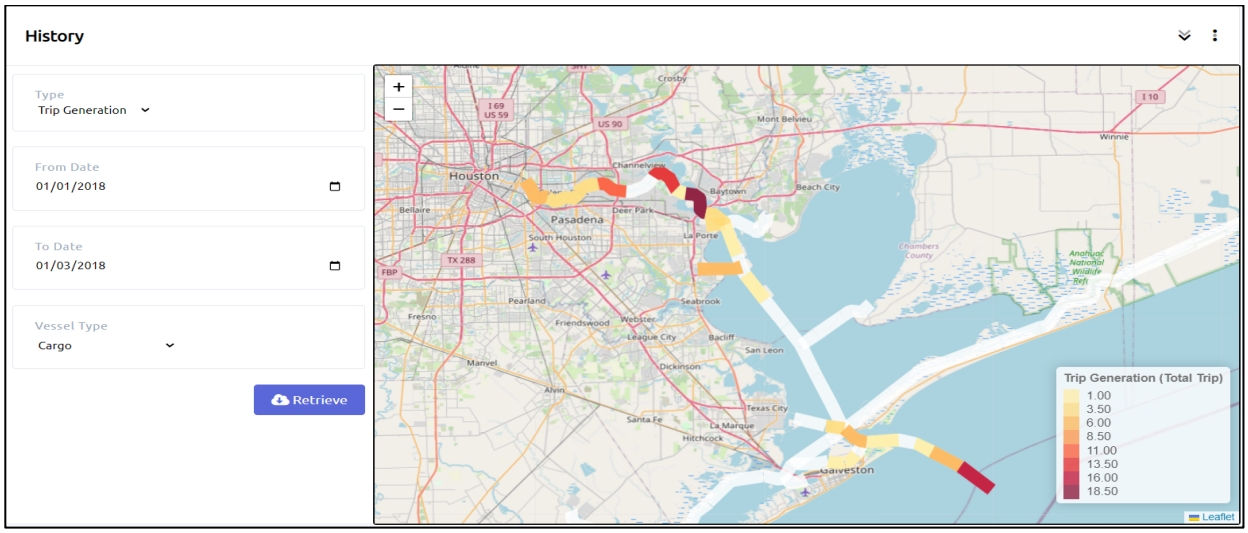

Fig 6. Trip generation

REFERENCES

[1] https://www.britannica.com/topic/Intracoastal-Waterway

[2] H. Li, J. Liu, R.W. Liu, N. Xiong, K. Wu, and T.H. Kim, "A dimensionality reduction-based multi-step clustering method for robust vessel trajectory analysis," *Sensors*, vol. 17, no. 8, pp. 1792, 2017.

[3] S.K. Zhang, G.Y. Shi, Z.J. Liu, Z.W. Zhao, and Z.L. Wu, "Data-driven based automatic maritime routing from massive AIS trajectories in the face of disparity," *Ocean Engineering*, vol. 155, pp. 240-250, 2018.

[4] F. Ren, Y. Han, S. Wang, and H. Jiang, "A novel high-dimensional trajectories construction network based on multi-clustering algorithm," *EURASIP Journal on Wireless Communications and Networking*, vol. 2022, no. 1, pp. 1-18, 2022.

[5] L. Eljabu, M. Etemad, and S. Matwin, "Spatial Clustering Method of Historical AIS Data for Maritime Traffic Routes Extraction," *in 2022 IEEE International Conference on Big Data (Big Data)*, Osaka, Japan, 2022, pp. 893-902.

[6] M.J. Kang, S. Zohoori, M. Hamidi, and X. Wu, "Study of narrow waterways congestion based on automatic identification system (AIS) data: A case study of Houston Ship Channel," *Journal of Ocean Engineering and Science*, vol. 7, no. 6, pp. 578-595, 2022.

[7] M. Kabir, M.J. Kang, X. Wu, and M. Hamidi, "Study on U-turn behavior of vessels in narrow waterways based on AIS data," Ocean Engineering, vol. 246, p. 110608, 2022.

[8] S. Zohoori, M.J. Kang, M. Hamidi, and B. Craig, "A vectorized algorithm for waterway traffic analysis using AIS data," *Journal of Ocean Technology*, vol. 16, no. 4, 2021.

[9] X. Wu, A.L. Mehta, V.A. Zaloom, and B.N. Craig, "Analysis of waterway transportation in Southeast Texas waterway based on AIS data," *Ocean Engineering*, vol. 121, pp. 196-209, 2016.

[10] X. Wu, A. Rahman, and V. A. Zaloom, "Study of travel behavior of vessels in narrow waterways using AIS data–A case study in Sabine-Neches Waterways," *Ocean Engineering*, vol. 147, pp. 399-413, 2018.

[11] S. Zohoori, U. Roy, M. Hamidi, and X. Wu, "Quantifying wide-body vessel navigation delay in narrow waterways: a case study at the Houston ship channel," *Journal of Waterway, Port, Coastal, and Ocean Engineering*, vol. 148, no. 4, p. 04022010, 2022.

[12] N. Evmides, L. Odysseos, M. P. Michaelides, and H. Herodotou, "an intelligent framework for vessel traffic monitoring using ais data," *in 2022 23rd IEEE International Conference on Mobile Data Management (MDM)*, Paphos, Cyprus, 2022, pp. 413-418.

[13] H. Chi, G. Pedrielli, T. Kister, S. H. Ng, and S. Bressan, "An AIS-based framework for real-time monitoring of vessel efficiency," *in 2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Singapore, 2015, pp. 1218-1222.

[14] Y. Zhang and W. Li, "Dynamic maritime traffic pattern recognition with online cleaning, compression, partition, and clustering of ais data," *Sensors*, vol. 22, no. 16, pp. 6307, 2022.

[15] I. Kontopoulos, K. Chatzikokolakis, D. Zissis, K. Tserpes, and G. Spiliopoulos, "Real-time maritime anomaly detection: detecting intentional AIS switch-off," *International Journal of Big Data Intelligence,* vol. 7, no. 2, pp. 85-96, 2020.

[16] P. Mobtahej, X. Zhang, M. Hamidi, and J. Zhang, "Deep learning-based anomaly detection for compressors using audio data," *in 2021 Annual Reliability and Maintainability Symposium (RAMS)*, May 2021, pp. 1-7.

[17] M. Gao, and G.Y. Shi , "Ship spatiotemporal key feature point online extraction based on AIS multi-sensor data using an improved sliding window algorithm," *Sensors*, vol.19, no.12, pp.2706, 2019.

[18] https://www.ibm.com/topics/etl