A Cloud Based Architecture Capable Of Perceiving and Predicting Multiple Vessel Behavior

Dimitrios Zissis, Elias K. Xidias & Dimitrios Lekkas

Department of Product & Systems Design Engineering,
University of the Aegean,
Ermoupolis, Syros, 84100

Abstract

Progressively huge amounts of data, tracking vessels during their voyages across the seas, are becoming available, mostly due to the Automatic Identification System (AIS) that vessels of specific categories are required to carry. These datasets provide detailed insights into the patterns vessels follow, while safely navigating across the globe, under various conditions. In this paper, we develop an Artificial Neural Network (ANN) capable of predicting a vessels future behavior (position, speed & course), based on events that occur in a predictable pattern, across large map areas. The main concept of this study, is to determine if an ANN is capable of inferring the unique behavioral patterns that each vessel follows and successively use this as a means for predicting multiple vessel behavior into a future point in time. We design, train and implement a proof of concept ANN, as a cloud based web application, with the ability of overlaying predicted short and long term vessel behavior on an interactive map. Our proposed approach could potentially assist in busy port scheduling, vessel route planning, anomaly detection and increasing overall Maritime Domain Awareness.

1. Introduction

Safety of navigation can be defined as "such conditions of conducting the ships at sea which ensure that ships are not endangered by collisions, stranding or storm damage" [1]. Nowadays, Information & Communication systems are been widely deployed, in an attempt to ensure such conditions or at least provide aids to those navigating dangerous waters. These systems make use of radio or satellite communications to transmit vessel and environment related information, such as a vessel's navigational properties and traffic conditions, but also safety and distress warnings. As of December 2004, the International Maritime Organization (IMO) requires all vessels over 299GT to carry an AIS transponder on board, which transmits their position, speed and course, amongst other static information (such as vessel's name, dimensions and voyage details). AIS transponders include a GPS (Global Positioning System) receiver, which collects position and movement details, and a VHF transmitter, which transmits periodically this information on two VHF channels (frequencies 161.975 MHz and 162.025 MHz) making this data available to the public domain. Much of this data is filtered through to the general public via the Internet. Authorities are interested in using this data to gain a better understanding of

conditions at seas and thus uncover threats to security, but also operate as an aid in search and rescue operations, accident investigation and navigation.

While in the past, surveillance had suffered from a lack of data, current tracking technology has transformed the problem into one of an overabundance of information, leading to a need for automated analysis. The major challenge faced today, is developing the ability to identify patterns emerging within huge amounts of data, fused from a variety of sources and generated from monitoring large numbers of vessels on a day-to-day basis. The extraction of implicit and often unknown information from large datasets belongs to the field of data mining. This is a multidisciplinary field, which draws input from a variety of domains including machine learning, statistics, database technologies, artificial intelligence and others. Specifically, machine learning provides the technical basis for data mining; it is used to extract useful information from raw data and to infer underlying structures [2].

In this manuscript we investigate the use of an Artificial Neural Network (ANN), in order to develop a model of short and long term multiple vessel behavior prediction, with the objective of increasing overall Maritime Domain Awareness (MDA). The main concept of this study is to determine if an ANN is capable of inferring the unique behavioral patterns each vessel follows on its passage across the Aegean and successively use this as a means for predicting their behavior into a future point in time. The ANN is tasked with acquiring knowledge of vessel patterns from real world observations, regarding a selection of passenger vessels navigating across the Aegean Sea during the summer months of 2013. Due to the size of the dataset and with the intent of building a scalable infrastructure that can potentially deal with larger datasets, we employ a cloud computing infrastructure to train the ANN in a timely fashion. We design, train and implement a proof of concept neural network, as a cloud based web application, with the ability to overlay predicted short and long term vessel behavior on an interactive map.

This manuscript is organized as follows: Section 1 provides an introduction to the problem space, Section 2, provides related research literature and discusses our proposed approach to the problem, Section 3, describes the tasks of data collection and processing before introducing the model architecture, while Section 4 presents implementation details and evaluation results of the trained ANN. In Section 5, we describe our approach to implementing the neural network as a cloud service using the Model-View-Controller pattern, while in the successive sections we discuss our conclusions and provide related literature references.

2. Approach

2.1. Data Fusion and MDA

Maritime Domain Awareness (MDA) is the effective understanding of activities, events and threats in the maritime environment that could impact global safety, security, economic activity or the environment [3]. The primary goals of MDA include "enhancing transparency in the maritime domain to detect, deter and defeat threats" and "enable accurate, dynamic, and confident decisions and responses to the full spectrum of maritime threats"[4]. Maritime threats include nation-wide, terrorist and piracy threats, but also environmental and social threats. The collection, fusion, analysis and dissemination of maritime intelligence and information are the fundamental building blocks of MDA. Data fusion combines multi origin information to determine relationships among data; thus improving the understanding of a current complex environment, but also attempting to predict its future state. As such, data fusion can enable improved estimation of situations and, therefore, improved responses to situations [5]. The JDL Data Fusion Group defines data fusion as

****
****

> *"a process dealing with the association, correlation, and combination of data and information from single and multiple sources to achieve refined position and identity estimates, and complete and timely assessments of situations and threats, and their significance. The process is characterized by continuous refinements of its estimates and assessments, and the evaluation of the need for additional sources, or modification of the process itself, to achieve improved results."*

The JDL proposed a number of possible ways of differentiating among types of data fusion processes, named fusion levels, that were later improved by Steinberg, Bowman & White, [5] "to more accurately representing logically different types of problems, which are generally (though not necessarily) solved by different techniques". These levels were defined as [5]

- **Level 0**. *Sub-Object*: Assessment estimation and prediction of signal or object observable states on the basis of pixel/signal-level data association and characterization
- **Level 1**. *Object Assessment*: estimation and prediction of entity states on the basis of observation-to-track association, continuous state estimation (e.g. kinematics) and discrete state estimation (e.g. target type and ID)
- **Level 2**. *Situation Assessment*: estimation and prediction of relations among entities, to include force structure and cross force relations, communications and perceptual influences, physical context)
- **Level 3**. *Impact Assessment*: estimation and prediction of effects on situations of planned or estimated/predicted actions by the participants
- **Level 4**. *Process Refinement*: adaptive data acquisition and processing to support mission objectives.

Higher-level fusion aims to enhance situational awareness and assessment [6,7]. Currently large data sets of vessel activities and behaviours are becoming available. Data of sufficient quantity and quality is believed to provide a fair representation of the current state of reality; if the data covers all aspects of a situation in a timely manner, one can then say that complete and continuous situational awareness has been achieved [8]. Most current approaches to vessel monitoring using AIS data are assessed at Level 0 & Level 1 fusion levels, object assessment, ultimately providing an estimation and prediction of vessel states on the basis of observation to track association and continuous state estimation (Vessel ID, kinematics and vessel type). In the aspect of improving MDA, it is now becoming a necessity that related Information Systems (IS) progress to higher levels of data fusion. This requires that related IS's develop the capabilities to accurately capture contextual information related to vessel routes, estimation and prediction of vessel behaviour; thus progressing to level 2 and 3 of data fusion.

A pattern is composed of events that repeat in a specific timeframe and predictable manner. Vessel behaviour can be defined as the sum of all characteristics defining vessels movement, such as vessel position, course, heading and speed, observed over a given period of time. Vessel behaviour monitored over time, provides insights into the navigational patterns followed by each vessel on specific routes. Machine learning techniques, can be employed to "learn" vessel behaviour and be used for predicting vessels behaviours. Such an approach can assist in busy port scheduling, collision avoidance, vessel route planning, but also anomaly detection and thus increasing overall Maritime Domain Awareness.

The development of global ship tracking systems opens possibilities of advancing maritime security far beyond the simple collision prevention. Anomaly detection has been identified as a critical component in order to achieve Situation Awareness in the context of information fusion and maritime surveillance [6,8–14].

Laxhammar, Falkman & Sviestins define anomaly detection as a method that supports the situation assessment process at JDL level 2 by indicating objects and situations that, in some sense, deviate from the expected, known or "normal" behavior and thus may be of interest for further investigation [10]. Generally, the methods which are used in the context of anomaly detection are based on statistical/probabilistic models [15–18], such as the Gaussian Mixture Model (GMM) and the adaptive Kernel Density Estimator (KDE) [10,19], Bayesian networks [13,14,20,21], but also neural networks [6,9,22]and hybrid approaches [23].

2.2. Neural Networks

An ANN is a machine learning information processing paradigm inspired by biological nervous systems. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons), which work in unison to solve specific problems [24]. In general, a neural network is a parallel system, capable of resolving problems that linear-computing cannot [25]. The unique characteristics of ANNs, adaptability, nonlinearity, and arbitrary function mapping, make them quite suitable and useful for forecasting tasks. According to Karlaftis & Vlahogianni, ANNs have been mainly used as data analytic methods because of their ability to work with massive amounts of multi-dimensional data, their modeling flexibility, their learning and generalization ability, their adaptability and their good predictive ability [26].

Forecasting has long been considered as the research filed of linear statistics. Traditional approaches, such as Box-Jenkins or ARIMA method [28], assume that real world observations are generated from linear processes [29]. However, they may be totally inappropriate, if the underlying mechanism is nonlinear. It is unreasonable to assume *a priori* that a particular realization of a given time series is generated by a linear process [29]. In fact, real world systems are most often nonlinear [30]. When modeling complex datasets with possible nonlinearities or missing data, ANN's are often regarded as more flexible compared to statistical models. ANNs are less sensitive to error term assumptions and can tolerate noise, chaotic components, and heavy tails better than most other methods [31]. Throughout academic literature, neural networks have been employed to predict various evolving patterns based on real world observations such as traffic flow prediction, bankruptcy prediction, stock market fluctuations, power electric demand, congestion, airline passenger traffic and many more.

In the maritime domain ANNs have been employed for tasks such as forecasting traffic flow at the Suez canal [32], predicting wave influence on the yaw motion of a ship [33] and vessel classification [34]. Lagerweij, Vries, & Someren, analyze vessel trajectories and classify them into three categories based on AIS data. In their work they perform the tasks of clustering, classification and outlier detection with the goal of identifying irregular vessel behaviors. Less work however has been conducted in the field of vessel movement prediction using ANN, mostly due to the lack of sufficient data. Ebada developed a system, capable of predicting accurately the turning tracks of ships [35]. The physical and operational data of a ship are described and used as inputs into the system in order to predict the turning maneuvers. Closely related to this work is Simsir and Ertugrul [36] study, with the aim of predicting the future coordinates of a manually controlled vessel using a trained ANN in the Bosporus Straits. The ANN was trained on position and speed data collected from vessels which navigated manually through the Strait. They were able of accurately predicting vessel positions in a three minute ahead window. Rhodes Bomberger, Seibert and Waxman developed a Fuzzy ARTMAP classification neural network architecture where normal vessel speeds for different regions in a port area were learned by clustering [6]. New data that is was not recognized by the network during online operation was labelled anomalous. The same

research group has also proposed and implemented a learning system of motion patterns for anomaly detection, where associative neural networks learn to predict future vessel locations within the boundaries of a specific port [9,11].

In our work, we improve on previous research in the field by attempting to train an Artificial Neural Network to predict multiple vessel's future behavior (position, speed & course) in a short and long term time frame, based on events that occur in a predictable pattern, across large map areas. We employ an ANN to learn vessel patterns as they travel, on safe voyages across a specific map segment and across the coverage of a number of receiving antennas. Specifically we focus on vessels departing from a specific port (in this case Piraeus and the close by smaller port of Rafena) and we attempt to determine if an ANN is capable of inferring the unique behavioral patterns each vessel follows on its passage across the Aegean. For such a journey information needs to be collected from as many as 30 coastal receiving stations to cover the vessels full journey back to port. We design, train and implement a proof of concept neural network as a cloud based web application, with the ability to overlay predicted short and long term vessel behavior on an interactive map.

2.3. Data Collection

Neural networks are only as good as the data they are given and the questions that are asked of them [37]. One of the major constrains of applying machine learning techniques to vessel position prediction has been the lack of data necessary for training the ANN. For this project we use data collected from MarineTraffic.com (MT). MT is part of an open, community-based project that provides real-time information to the public, about ship movements and port traffic for many countries around the world.

The basic principle of data fusion and dissemination is depicted in Figure 1 below. All involved parties are depicted here: The network of AIS receivers covering different or overlapping areas, fusing data into the system; the central premises hosting the database, the data collection software and the web applications; other VTS networks mutually exchanging data; other applications receiving data feed in various formats; and finally the end-users consuming the services through the web or through mobile devices.

Figure 1. Generic Topology of MarineTraffic data fusion system and peripheral equipment and applications

AIS transmissions can be defined as spatial time series datasets [9], describing the movements of vessels. Data points in geospatial data sets are typically multi-dimensional; including coordinates, a timestamp and descriptive information. Accordingly, an AIS message contains the vessel's Maritime Mobile Service Identity (MMSI) – a unique nine digit identification number; Navigation status; Rate of turn ; Speed over ground; Positional accuracy; Course over ground; True heading; True bearing at own position; UTC Seconds. Additionally messages may contain Radio call sign, Vessel Name, Vessel Type; Vessel Dimensions; Vessel Draught, Vessel Destination and vessels estimated time of arrival.

Table 1 In the table an instance of the dataset provided by MT is presented. In this table we do not use the original MMSI vessel information. AIS messages do not include time related information. Messages are time stamped when received by MT to precisely identify messages in a time series sequence.

The test dataset used for this project includes three passenger vessels travelling across the Aegean Sea during the summer months of 2013. Passenger vessels follow a highly repetitive pattern; as their daily routes start and conclude at their departure port (Piraeus or Rafena) after visiting a number of islands within a 12-24 hours' time frame. They each follow a distinct route between islands; Vessel 1 visits Piraeus, Syros, Paros, Naxos Amorgos (Figure 2), Vessel 2 follows the Route Rafena, Andros, Tinos, Mykonos, while Vessel 3 follows the route Piraeus, Chios, Mytilene. These routes are within good coverage of MarineTraffic giving us data of sufficient quality and quantity, providing a fair representation of reality.

Figure 2. Screen capture depicting the route followed by Vessel 1 travelling from Piraeus to Syros, Paros, Naxos Amorgos and back to Piraeus

3. Data Processing and Model Construction

3.1. Data Preparation

Neural networks require properly formatted training data. This involves cleaning the data: removing errors, preprocessing and post processing [37]. A number of preprocessing steps have to be completed to convert the dataset used in this research project into a time-series data set suitable for ANN predictions. AIS data is received as a series of messages following a nonstandard pattern of irregular time intervals. Data is received as an asynchronous data stream, often containing duplicate or missing entries received from multiple base stations at varying time intervals

Table 2 Data Pre-processing

In order to produce a valid and useful training input for the neural network, data had to be clustered and down sampled. Original position reports are broadcasted by ships at time intervals that vary between 3 seconds and more than 30 seconds, depending on their speed and the type of their AIS transponder. These reports are used for the real-time map display, but they are only archived every 2 minutes for each vessel (as this interval is enough for tracking purposes and the applications are not used for real time navigation purposes). For the purpose of the ANN training we experimented with an even smaller frequency of position reports. An interval of 15 minutes would be enough to detect an anomaly in vessel tracks, while at the same time we ensure that the time series will

be uninterrupted even for areas where the AIS signal reception is poor and the collected data is not as frequent as in fully covered areas. The following SQL query is used to achieve the described data down-sampling:

```sql
SELECT A.MMSI, A.LON, A.LAT, A.SPEED, A.COURSE, A.TIMESTAMP
FROM POSITIONS A
INNER JOIN
(SELECT MIN(TIMESTAMP) AS TIMESTAMP, MMSI
FROM POSITIONS
WHERE MMSI IN (123456789,234567890,345678901)
GROUP BY DATEPART(YEAR, TIMESTAMP, DATEPART(MONTH, TIMESTAMP), DATEPART(DAY,
TIMESTAMP), DATEPART(HOUR, TIMESTAMP), DATEPART(MINUTE, TIMESTAMP)/15, MMSI) B
ON A.MMSI=B.MMSI AND A.TIMESTAMP=B.TIMESTAMP
ORDER BY A.MMSI, A.TIMESTAMP
```

Input data had to be additionally clustered in specific geographical locations. Although the observed ships follow specific predetermined routes, small deviations from their predefined courses are expected to be normal. In order to produce a dataset that will lead to increased accuracy, we decided to use multiple vessel tracks following a specific route and to cluster them into specific geographical 'bounding boxes'. In order to achieve this, we are rounding each geographical coordinate (latitude and longitude stored in decimal degrees) to the second decimal digit and shifting both values by 0.005. This gives an interval of 0.01 degrees between each possible successive coordinate, which is equivalent to a distance of 0.6 nautical miles (approx. 1111 meters) measured on Y-axis and equivalent to about one pixel on Google maps at zoom level 7. The method is depicted in Figure 3 below. Following this, our training set contained approximately 2,500 entries regarding three passenger vessels as recorded over the three month period. In respect to building an architecture capable of scaling (big data) and high interoperability we adopted the principles of Service Oriented Architectures exposing Simple Object Access Methods (SOAP) or Representational State Transfer (REST) interfaces. We built a C# web service capable of returning a dataset for the required vessel or vessels, automatically processed as described above.

Figure 3. Position Clustering: Two slightly different tracks clustered into bounding boxes. Blue circles are the points actually used in the training time-series

As discussed in [6] we follow the approach of velocity discretization. Four speed bins were defined as "stopped", "slow", "medium" and "fast moving" vessel. Vessels are considered "stopped" when vessel speed is reported below five knots, "slow moving" when below 8.5 knots, "medium" moving up to 17 knots, and "fast moving" at speeds up to 26 knots. We transform the vessel heading from 0-360 degrees into four cardinal directions (North, East, South, West) and four intercardinal directions [NorthEast (NE), SouthEast (SE), SouthWest (SW), and NorthWest (NW)]. The data series are then normalized in the range [-1, 1] before training the neural network. Inputs and outputs to our neural network are also normalized to the original data scale before submitted to the NN. This provided for much smaller training sizes and greater accuracy.

Table 3. Data Normalization

3.2. Neural Network Architecture

The structure of the neural network needs to be able to hold the complexity of the at state problem. Careful analysis and design are required for the selection of the critical elements of an ANN, which include the size of the input layer, the output layer, the number of hidden neurons and the number of hidden layers. Although there has been extensive research on the design of optimal neural networks structure, it is still largely an art and a matter of experimentation as each problem presents a distinct challenge[29].

3.2.1. ANN Structure

Multilayer feedforward networks are a class of universal approximators. According to Hornik, Stinchcombe and White , multilayer feedforward networks are capable of approximating any Borel measurable function from one finite dimensional space to another to any desired degree of accuracy, provided sufficient hidden units are available[38]. feedforward neural network consists of a number of interconnected processing units, organized in layers. All connections have a different weight and ultimately these connections encode the knowledge of the network. Data enters the input layer and is outputted at the output layer as in Figure 4. A critical decision when designing any ANN is the selection of the ideal architecture to fit the problem.

Figure 4. A depiction of a feedforward neural network.

3.2.2. Input

The number of input nodes corresponds to the number of variables in the input vector used to forecast future values. For neural networks implemented to predict future values in a time series dataset, the number of input neurons corresponds to the number of lagged observations required to discover the underlying pattern [29]. It is important to consider the rate of change between entries in the dataset when selecting the number of lagged observations.

Figure 5. For neural networks to predict future values in a time series dataset, the number of input neurons corresponds to the number of lagged observations required to discover the underlying pattern

Our forecasting ANN is required to predict a vessels Latitude, Longitude, Speed and Course at a future point in time (Figure 5). Neural networks can be trained to predict one-step-ahead (based on the time interval data is collected in the

dataset prediction will occur for the following value) or multi-step-ahead forecasting. For multistep ahead forecasting, iterative or direct methods of forecasting can be used. In the iterative approaches, forecast values are used as inputs for the preceding calculation, while direct methods requires the neural network to have several output nodes to directly forecast each step into the future [29].

For our given dataset we experimented with a backward window size of three; thus requiring a structure which could handle 15 input neurons, for the short term prediction and ten lagged observations for long term prediction; thus requiring a structure that could handle 50 input neurons. For short term prediction (15 minutes) we trained the neural network to output a single prediction while for long term prediction [5 steps ahead (5*15 minutes) and 10 steps ahead (10*15 minutes)], we experimented with iterative and direct prediction approaches.

3.2.3. Output

The number of output nodes corresponds to the forecasting prediction. According to the selected prediction (short or long term) and approach, we differ our output neuron number. For short term predictions, the output neuron count is 4 (we do not include the vessel id in the prediction as this is inferred), giving us a prediction for the vessels future Latitude, Longitude, Heading and Speed. For long term predictions we increase the output neuron layer accordingly to the future prediction window size (e.g. 5 steps ahead requires 20 output neurons).

An important element of the neural networks structure is their net inputs by using a scalar-to-scalar function called "the activation function or threshold function or transfer function", output a result value called the "unit's activation" [39]. In general, the activation function introduces a degree of nonlinearity that is valuable for most ANN applications. The predicted output of our ANN is in the range [-1, 1]; thus we selected the hyperbolic tangent function as an activation function for the hidden and output layers [40].

3.2.4. Hidden

Identifying the number of hidden neurons and layers is a matter of finding the correct balance between giving the network enough power to model the complexity of the problem and learning the data. If the network is too small then the network will not be able to learn the required pattern (under-fitting) while if the network is oversized, training will be slowed proportionally and the network will learn the dataset too well thus reducing its ability to generalize. The most common way in determining the number of hidden nodes is via experimentation or by trial-and-error [29].

In our given approach we make use of incremental pruning. This enables the ANN to autonomously select the optimal hidden layer structure, based on its capacity to learn best. For such an approach we predetermine the number of input and output layers while providing a range of minimum to maximum numbers of hidden neurons and layers. The algorithm will incrementally increase the size of the neural network and retrain at each increment until it reaches the maximum limits. When reaching the maximum limit, the configuration that trained best is considered the optimal network configuration. Incremental pruning led us to proposed structures of 2 hidden layers with 77 and 12 hidden

neurons, 87 and 11 hidden neurons, 88 and 14 hidden neurons and 60 and 8 hidden neurons. We then tested each of these sets accordingly leading us to a selection of 87/11 neurons and two hidden layers for short term prediction and 88/14 for long term prediction.

3.3. Neural Network Training

Training is the means by which neural network weights are adjusted to give desirable outputs. The propagation training algorithm will go through a series of iterations that will most likely improve the neural network's error rate by some degree [41]. The error rate is the percentage difference between the actual output from the neural network and the ideal output provided by the training data [42]. The Mean Square Error (MSE) is an error calculation method used in describing how well a machine learning method, typically a regression model, represents the data being modelled [41]. This process is repeated until the error for each training pattern drops under a certain accepted level. The rate of learning can be rather slow. We set the target MSE to 0.01 for short term prediction while 0.04 for long term prediction. For the purposes of this project, we experimented with Manhattan Propagation, Levenber Marquardt Training, Quick Propagation, Manhattan Propagation, Back Propagation and Resilient Propagation with the latter two achieving best MSE (0.01-0.04) and training rates. The Resilient Propagation approach was adopted as was able to outperform Back Propagation on most instances [41].

4. Network Training and Evaluation

We implement our ANN in C# using the Encog3 machine learning library [42]. Encog is an open source advanced machine learning framework that supports a variety of algorithms, as well as methods for normalizing and processing the data. Our selection to use open source libraries was reached in light of greater research reproducibility. Most Encog training algorithms are multi-threaded and scale well to multicore infrastructure (such as large cloud computing instances). As the AIS data is a high volume dataset, we employ a cloud infrastructure to speed up the training process, but also provide the final solution as a cloud Software as a Service (SAS). A design choice was to build the solution around a service based model and follow the principles of Service Oriented Architectures exposing SOAP or REST (web services) interfaces.

Cloud computing offers the benefits of high solution scalability, resource pooling, resiliency and availability. Cloud Infrastructure Okeanos, provided by GRNET, for the Greek Research and Academic Community, is an *Infrastructure as a Service* level cloud service [43]. For the purposes of this project we employed large virtual machines (instances) capable of intense computing in the cloud. These instances offer High CPU capabilities and large memory capacity.

The training process was performed in the cloud and completed in a number of minutes. After training was completed, we evaluated the trained ANN by feeding it data that was excluded from the original dataset. Often before an ANN is given a specific dataset for training, a portion of it is kept aside for evaluating the networks performance; usually a random 70% is used for training purposes and 30% for testing. In our case, as evaluation data we used a number of real world observations acquired the days following those used for training the ANN. The selected vessels continued on their predefined paths (as in the training patterns) for several successive days. The training data set is used

exclusively for model development and the test sample is used only to assess the trained network. This data was pre and post-processed in the same method with the training data.

Overall the results predicted using the ANN compared well with the actual observations of vessel movements. Most importantly the ANN was able to infer each separate vessel pattern and each separate route performed by the passenger vessels. The trained ANN was able to accurately predict future positions of the three vessels during their voyage (Figures 6 & 7). The model was able to recognize the overall pattern of "Course" deviations, but made some erroneous predictions when rapid course changes were performed by the vessel. It is interesting to note that these rapid changes (spikes in the graph) are mostly due to port manoeuvres executed by the vessel while docking in a port (Figure 8). The trained ANN gives good performance at predicting vessel speed, with the ability to recognize sudden changes in speed, although these at some point are slightly shifted (Figure9). Once again it is interesting to note that these sudden changes in the vessels speed are when the vessel enters a port. The ANN was capable of predicting these sudden spikes with good accuracy.

Figure 6. Chart depicts validation of short term Latitude Prediction for three vessels travelling across the Aegean. Blue line depicts the ideal output while red line depicts the predicted output

Figure 7. Chart depicts validation of short term Longitude Prediction for three vessels travelling across the Aegean. Blue line depicts the ideal output while red line depicts the predicted output

Figure 8. Chart depicts validation of short term Course Prediction for one vessel travelling across the Aegean. Blue dotted line depicts the ideal output while red line depicts the predicted output

Figure 9. Chart depicts validation of short term Speed Prediction for a vessel travelling across the Aegean. It is interesting to note that the trained ANN is capable of learning the vessels pattern, predicting speed drops when entering ports during its voyage. Blue dotted line depicts the ideal output while red line the predicted output.

As previously discussed for multistep ahead predictions, iterative or direct methods of forecasting can be used. In our case, the iterative prediction approach was not capable of giving good predictions over a number of steps, as the prediction error soon escalated. The direct approach (multiple future predictions simultaneously), provided much

more accurate results. The trained ANN for long term prediction has 20 output neurons for 5 step ahead prediction giving predicted Latitude, Longitude, Course and Speed for each step along the way.

Although the long term ANN was capable of inferring the relationship between separate vessels and consecutively learning each vessels pattern, it did not perform as well as the short term prediction ANN. The trained ANN was capable of recognising the overall pattern and predicting future position (Latitude and Longitude) with a small error, but did not perform as well on peaks (figures 10 & 11). It had difficulty in predicting rapid course or speed changes, (Figure2 & 13). This is mostly because using a backward window of 10 and future window of 5, with intervals of 15 minutes, does not give any intention of rapid course or speed changes. While when the ANN is capable of learning the vessels pattern it has difficulty predicting in the long term, sudden peaks or drops in the pattern. On the overall, it is able to accurately predict when a vessel is at open sea going at full speed and stopped when in port.

Figure 10. Chart depicts validation of long term Latitude Prediction for three vessels travelling across the Aegean. Blue line depicts the ideal output while red line the predicted output

Figure 11. Chart depicts validation of long term Longitude Prediction for three vessels travelling across the Aegean. Blue line depicts the ideal output while red line is the predicted output

Figure 12. Validation of long term Course Prediction results for one vessel travelling across the Aegean. Blue dotted line depicts the ideal output while red line depicts the predicted output.

Figure 13. Validation of long term Speed Prediction results for one vessel travelling across the Aegean. Blue dotted line depicts the ideal output while red line depicts the predicted output.

5. Model use as a web application

We designed and developed a prototype web application, which uses the trained short term vessel behaviour prediction to predict future vessel behaviour and overlay it on an interactive map. Due to the nature of our application, the web application was built using the ASP.NET MVC 4 (Model-View-Controller) framework. The ASP.NET MVC 4 is a framework for building scalable, standards-based web applications using well-established design patterns, that places an emphasis on a loosely coupled application architecture and highly maintainable code [44,45]. The Model-View-Controller pattern is an architectural pattern that encourages strict isolation between the individual parts of an

application. This isolation is better known as separation of concerns, or, in more general terms, "loose coupling". The MVC pattern splits an application into three layers: the model, the view, and the controller. The model represents core business logic and data. Models encapsulate the properties and behaviour of a domain entity and expose properties that describe the entity. The proposed solution contains models encapsulating vessel data. The view in MVC is responsible for transforming a model or models into a visual representation. In web applications, this most often means generating HTML to be rendered in the user's browser, although views can manifest in many forms. The models are rendered into HTML forms for data entry, data visualization, but also are used to generate the interactive map. The controller, essentially C# code, controls the application logic and acts as the coordinator between the view and the model. Controllers receive input from users via the view, and then work with the model to perform specific actions, passing the results back to the view.

The trained ANN is used as a controller to predict future vessel positions according to user input. The predictions are overlaid on an interactive map and provided to the public via http://mt-nn.azurewebsites.net/. At present information regarding vessel movements is provided by the user for demonstration purposes, so as not to overload MT databases until the solution has been thoroughly evaluated. Forecasts from the neural network are transformed to the original data scale before positioned on the map. A user can interact with a vessel from the interactive map, selecting to view the future predictions, while the controller responsible retrieves required data from the database. This information is then passed on to the controller responsible for the model prediction and output is calculated. This procedure is described below in more detail.

A user needs to interact with a vessel select future prediction and the previous position detail is retrieved from the DB and forwarded to the input neuron the trained ANN as described below in Figure 14.

Figure 14. MVC web architecture for ANN vessel prediction

- On user request, the ANN controller returns the Map View after loading vessels current positions
- The user is capable of interacting with the map and selecting vessels
- From the loaded map view the user is capable of requesting a prediction for selected vessel/s
- The ANN controller loads the previously trained ANN model and previous vessel/s positions to use as input for the network.
- The trained ANN is used to predict future vessel behaviour which is used to update the map view

6. Discussion

In this paper we discuss preliminary results of on-going research into vessel behaviour prediction and anomaly detection. Examining the results, we are able to claim with confidence, that the trained ANN is capable of inferring

separate vessel patterns within a large dataset and predicting with good accuracy multiple future vessel positions, especially in short term predictions. The ANN shows difficulty in accurately predicting rapid course changes and speed spikes, specifically in regards to long term predictions. This can be considered as expected, as often there are no indicators of these drastic changes in the vessels pattern, especially when performing in port manoeuvres. It is common vessel behaviour to perform different in port manoeuvres according to the environment settings (e.g weather conditions, other vessels in port). It is possible that to increase its predictive capacity in this domain, the ANN would require shorter time intervals between observations and an increase in the backward window; thus giving the ANN the ability to capture the rate of change in the prediction value. On the other side the trained ANN shows a valid capability of understanding the pattern and identifying speed drops when vessels enter a port; thus recognizing behaviour change due to arriving or departing from a port. The models implemented in this approach are limited as to not having the capacity of apprehending additional environmental conditions affecting vessel behaviour, as these are lacking from the dataset. In future works we plan on enhancing our models prediction capacity by enriching our training dataset with information regarding weather conditions and proximities to other vessels or ports in the area.

7. Conclusions

In this paper, we train an ANN to predict future vessel's behaviour (position, speed & course), based on events that occur in a predictable pattern, across large map areas. The implemented models have been trained and evaluated on multiple real world vessel observations based on the AIS. The approach shows great potential in learning and understanding several vessel patterns and is capable of predicting future behaviour successfully. We design and implement a proof of concept ANN as a cloud based web application, with the ability to overlay predicted short and long term vessel behaviour on an interactive map. Our proposed approach, could potentially assist in busy port scheduling, vessel route planning, anomaly detection and increasing overall Maritime Domain Awareness.

8. Acknowledgement

9. List of Abbreviations and Acronyms

AI: Artificial Intelligence

AIS: Automatic Identification System

ANN: Artificial Neural Network

GPS: Global Positioning System

JDL: Joint Directors of Laboratories

MDA: Maritime Domain Awareness

MMSI: Maritime Mobile Service Identity

MSE: Mean Squared Error

MVC: Model View Controller

REST: Representational State Transfer

SOAP: Simple Object Access protocol

VHF: Very High Frequency

10. References

[1]   W.M. and J.U. Z. Kopacz, The Maritime Safety System, its Main Components and Elements, J. Navig. 54 (2001) 199–211.

[2]   I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, Second Edition, Morgan Kaufmann Publishers (Elsevier), 2005.

[3]   B. Santos, K. Lunday, Maritime Domain Awareness, Coast Guard J. Saf. Secur. Sea, Proc. Mar. Saf. Secur. Counc. 66 (2009).

[4]   NSPD 41/HSPD13, National Plan To Achieve Maritime Domain Awareness For The National Strategy For Maritime Security, 2005.

[5]   A.N. Steinberg, A.N. Bowman, C.L. White, F. E., Revisions to the JDL Data Fusion Model, 1999.

[6]   N. Bomberger, B. Rhodes, M. Seibert, A. Waxman, Associative Learning of Vessel Motion Patterns for Maritime Situation Awareness, in: 2006 9th Int. Conf. Inf. Fusion, IEEE, 2006: pp. 1–8. doi:10.1109/ICIF.2006.301661.

[7]   M.R. Endsley, Toward a Theory of Situation Awareness in Dynamic Systems, Hum. Factors J. Hum. Factors Ergon. Soc. 37 (1995) 32–64. doi:10.1518/001872095779049543.

[8]   E. Martineau, J. Roy, Maritime Anomaly Detection: Domain Introduction and Review of Selected Literature, Technical report, DTIC Document, 2011.

[9]   B.J. Rhodes, N.A. Bomberger, M. Zandipour, Probabilistic associative learning of vessel motion patterns at multiple spatial scales for maritime situation awareness, in: 2007 10th Int. Conf. Inf. Fusion, IEEE, 2007: pp. 1–8. doi:10.1109/ICIF.2007.4408127.

[10]   R. Laxhammar, G. Falkman, E. Sviestins, Anomaly detection in sea traffic - A comparison of the Gaussian Mixture Model and the Kernel Density Estimator, (n.d.) 756–763.

[11]   R. Laxhammar, Anomaly detection for sea surveillance, in: 11th Int. Conf. Inf. Fusion, IEEE, 2008: pp. 1–8.

[12]   C. Brax, Anomaly detection in the surveillance domain, Örebro universitet, 2011.

[13]   F. Johansson, G. Falkman, Detection of vessel anomalies - a Bayesian network approach, in: 2007 3rd Int. Conf. Intell. Sensors, Sens. Networks Inf., IEEE, 2007: pp. 395–400. doi:10.1109/ISSNIP.2007.4496876.

[14]   A. Nicholson, F. Cozman, S. Mascaro, A.E. Nicholso, K.B. Korb, Anomaly detection in vessel tracks using Bayesian networks, Int. J. Approx. Reason. 55 (2014) 84–98.

[15]   A. Dahlbom, L. Niklasson, Trajectory clustering for coastal surveillance, (2007) 1–8. doi:10.1109/ICIF.2007.4408114.

[16]   S.C. David Lindsay, Effective Probability Forecasting for Time Series Data Using Standard Machine Learning Techniques, in: S. Singh, M. Singh, C. Apte, P. Perner (Eds.), Pattern Recognit. Data Min., Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. doi:10.1007/11551188.

[17]   G.K.D. de Vries, M. van Someren, Machine learning for vessel trajectories using compression, alignments and domain knowledge, Expert Syst. Appl. 39 (2012) 13426–13439.

[18]   K. Kowalska, L. Peel, Maritime anomaly detection using Gaussian Process active learning, (n.d.) 1164–1171.

[19]   B. Ristic, B. La Scala, M. Morelande, N. Gordon, Statistical analysis of motion patterns in AIS Data: Anomaly detection and motion prediction, (2008) 1–7.

[20]   R.O. Lane, D.A. Nevell, S.D. Hayward, T.W. Beaney, Maritime anomaly detection and threat assessment, (2010) 1–8.

[21]   F. Fooladvandi, C. Brax, P. Gustavsson, M. Fredin, Signature-based activity detection based on Bayesian networks acquired from expert knowledge, (2009) 436–443.

[22]   S.-B.C. Sang-Jun Han, Kyung-Joong Kim, Evolutionary Learning Program's Behavior in Neural Networks for Anomaly Detection, in: N.R. Pal, N. Kasabov, R.K. Mudi, S. Pal, S.K. Parui (Eds.), Neural Inf. Process., Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. doi:10.1007/b103766.

****

[23]   J.B. Kraiman, S.L. Arouh, M.L. Webb, Automated anomaly detection processor, in: A.F. Sisti, D.A. Trevisani (Eds.), Proc. SPIE 4716, Enabling Technol. Simul. Sci. VI, 128, International Society for Optics and Photonics, 2002: pp. 128–137. doi:10.1117/12.474940.

[24]   V. Bevilacqua, Hidden Markov Models for Recognition using Artificial Neural Networks, in: Int. Conf. Intell. Comput. Intell. Comput., Springer, 2006: p. 1331.

[25]   D. Verber, Implementation of Massive Artificial Neural Networks with CUDA, in: C. Volosencu (Ed.), Cut. Edge Res. New Technol., InTech, 2012. doi:10.5772/2431.

[26]   M.G. Karlaftis, E.I. Vlahogianni, Statistical methods versus neural networks in transportation research: Differences, similarities and some insights, Transp. Res. Part C Emerg. Technol. 19 (2011) 387–399.

[27]   G.M. Box, G.E.P., Jenkins, Time Series Analysis: Forecast- networks and robust time series prediction, IEEE Trans. (1976).

[28]   A. Pankratz, Forecasting with Univariate Box-Jenkins Models: Concepts and Cases, John Wiley, New York, 1983.

[29]   G. Zhang, B. Eddy Patuwo, M. Y. Hu, Forecasting with artificial neural networks: The State of the Art, Int. J. Forecast. 14 (1998) 35–62.

[30]   C.W.J. Granger, Strategies for modelling nonlinear time-series relationships, Econ. Rec. 69 (1993) 233–238.

[31]   I. Kaastra, M. Boyd, Designing a neural network for forecasting financial and economic time series, Neurocomputing. 10 (1996) 215–236.

[32]   M.M. Mostafa, Forecasting the Suez Canal traffic: a neural network analysis, Marit. Policy Manag. 31 (2004) 139–156. doi:10.1080/0308883032000174463.

[33]   V. Nicolau, D. Aiordachioaie, R. Popa, Neural network prediction of the wave influence on the yaw motion of a ship, in: 2004 IEEE Int. Jt. Conf. Neural Networks (IEEE Cat. No.04CH37541), IEEE, 2004: pp. 2801–2806. doi:10.1109/IJCNN.2004.1381100.

[34]   R. Lagerweij, G. de Vries, M. van Someren, Learning a Model of Ship Movements., University of Amsterdam, 2009.

[35]   M.A.M. A. Ebada, Prediction of Ship Turning Manoeuvre Using Artificial Neural Networks, in: 8th Numer. Towing Tank Symp., 2005.

[36]   U. Simsir, S. Ertugrul, Prediction of manually controlled vessels' position and course navigating in narrow waterways using Artificial Neural Networks, Appl. Soft Comput. 9 (2009) 1217–1224.

[37]   E.M. Azoff, Neural Network Time Series Forecasting of Financial Markets, John Wiley & Sons, Inc., 1994.

[38]   K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks. 2 (1989) 359–366.

[39]   B. Karlik, A.V. Olgac, Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks, Int. J. Artif. Intell. Expert Syst. 1 (2010) 111.

[40]   G.S. da S. Gomes, T.B. Ludermir, Optimization of the weights and asymmetric activation function family of neural network for time series forecasting, Expert Syst. Appl. 40 (2013) 6438–6446.

[41]   J. Heaton, Programming Neural Networks with Encog3 in C#, 2nd Edition, Heaton Research, Incorporated, 2011.

[42]   J. Heaton, Introduction to Neural Networks for C#, 2nd Edition, Heaton Research, Incorporated, 2008.

[43]   D. Zissis, D. Lekkas, Addressing cloud computing security issues, Futur. Gener. Comput. Syst. (2011). doi:10.1016/j.future.2010.12.006.

[44]   J. Chadwick, T. Snyder, H. Panda, Programming ASP.NET MVC 4: Developing Real-World Web Applications with ASP.NET MVC, O'Reilly Media, 2012.

[45]   J. Galloway, P. Haack, B. Wilson, K.S. Allen, Professional ASP.NET MVC 4 (Wrox Professional Guides), John Wiley & Sons, 2012.