Maritime Transport Research 3 (2022) 100073

Contents lists available at ScienceDirect

Maritime Transport Research

journal homepage: www.sciencedirect.com/journal/maritime-transport-research

Achieving fuel efficiency of harbour craft vessel via combined time-series and classification machine learning model with operational data

Januwar Hadi [a], Dimitrios Konovessis [b], Zhi Yung Tay [a,*]

[a] Engineering Cluster, Singapore Institute of Technology, 10 Dover Drive, 534038, Singapore
[b] Department of Naval Architecture, Ocean and Marine Engineering, University of Strathclyde, 100 Montrose St, Glasgow G4 0LZ, United Kingdom

ARTICLE INFO

Keywords:
Machine learning
Harbour craft vessel
Ship energy efficiency
Time-series and classification LSTM model
Haar wavelet

ABSTRACT

This paper presents work on forecasting the fuel consumption rate of a harbour craft vessel through the combined time-series and classification prediction modelling. This study utilizes the machine learning tool which is trained using the 5-month raw operational data, i.e., fuel rate, vessel position and wind data. The Haar wavelet transform filters the noisy readings in the fuel flow rate data. Wind data are transformed into wind effect (drag), and the vessel speed is acquired through transforming GPS coordinates of vessel location to vessel distance travelled over time. Subsequently, the *k*-means clustering groups the tugboat operational data from the same operations (i.e., cruising and towing) for the training of the classification model. Both the time-series (LSTM network) and classification models are executed in parallel to make prediction results. The comparison of empirical results is made to discuss the effect of different architectures and hyperparameters on the prediction performance. Finally, fuel usage optimization by hypothetical adjustment of vessel speed is presented as one direct application of the methods presented in this paper.

Introduction

In response to global warming concerns, the International Maritime Organization (IMO) has vowed to cut greenhouse gas emissions (GHG) in maritime transportation by at least 40% by 2030 (International Maritime Organisation IMO, 2011) based on revisions to mandatory requirements on energy efficiency criteria for ships. This obligatory shift not only affected international travelling vessels in their search for alternative means of optimising their operation to achieve fuel efficiency but also affected harbour craft vessels in comparable ways. Furthermore, several countries such as Singapore and Norway have conceptualised the development of a maritime decarbonisation blueprint, with one of the focal areas being harbour craft, which plays a significant role in ensuring the ports' functionality (MPA, 2017; Norwegian Government Action Plan, 2019). For instance, harbour craft like tugboat contributes significantly to GHG by burning fuel for piloting and assisting vessels in docking. According to Singapore sea straits CO_2 emission statistics presented by Leong and Singhal, 2015, tugboat operation contributes 993 tonnes/day of CO_2 emissions. This represents similar CO_2 emissions to 78,793 typical passenger cars based on information published by the US Environmental Protection Agency (2018). From these considerations, numerous studies have been conducted to develop machine learning models (Gkerekos et al., 2019; Kim et al.,

* Corresponding author.
E-mail address: zhiyung.tay@singaporetech.edu.sg (Z.Y. Tay).

https://doi.org/10.1016/j.martra.2022.100073
Received 15 February 2022; Received in revised form 22 August 2022; Accepted 7 September 2022
Available online 18 September 2022

Table 1
Vessel specifications.

Main particulars	Value	Unit
Length overall (LOA)	29	m
Displacement	665	tonnes
Maximum speed	12	knots
Main engines	NIIGATA 6L26HLX	
Number of engines	2	
Total BHP	4000	BHP
Type of propulsion	Azimuth pod	
Number of propulsors	2	

2021; Petersen et al., 2012; Uyanık et al., 2020) that can forecast fuel consumption swiftly and accurately, providing advice to operators on how to operate the vessel efficiently and, therefore, reduce GHG emissions.

The study on vessels' fuel consumption model had been ongoing for a few years and several machine learning models developed are utilised for various types of oceans travelling vessels such as container ships, tankers and cruise ships (Fam et al., 2021). For instance, Bialystocki and Konovessis (2016) developed a polynomial regression model for a vehicle's carrier vessel in forecasting fuel consumption while considering several highly correlated factors such as operational variables and environmental condition noon reports recorded at various operational load conditions. A variational autoencoder based on the neural network generative model was incorporated by encompassing crew navigational experience in making prediction of vessel route and fuel consumption (Hadi et al., 2022). (Tay et al., 2021b,a) utilised the *k*-means clustering and Hidden Markov Model (Rabiner, 1989) in the prediction of fuel consumption by deducing the stochastic correlations among the variables collected from a tugboat. In a recent study, Kee et al. (2018) proposed developing a multi-linear regression (MLR) model for a vessel to predict fuel consumption in laden or ballast conditions using several input factors, namely operational and environmental historical data. When the developed model was compared to the artificial neural network (ANN), it was discovered that ANN outperforms MLR in terms of fuel prediction accuracy. In the study by Leifsson et al. (2008), a grey box model for predicting fuel consumption for containerships was developed. The grey box incorporates empirically generated vessel resistance (as a white box model) integrated with operational and environmental historical data used as input parameters into the ANN (as a black-box model). The results demonstrated that the grey box model outperforms the white box model in terms of fuel prediction accuracy. Gkerekos et al. (2019) presented a study comparing multiple machine learning models in predicting vessel fuel consumption using data obtained from onboard sensors. The results indicated that the ANN outperforms traditional regression machine learning models.

The use of ANN as building blocks for a fuel prediction model is proven to produce promising results over the traditional statistical methods (Hu et al., 2019). This is because the ship fuel consumption is affected by many factors, which has caused difficulties to analyse using traditional statistical methods. In 2021, a study using 13,000 TEU class container ship data encourages the use of ANN further to build prediction models. The models could provide valuable information for ship operators to support decision-making to maintain efficient operating conditions, albeit in the context of ocean-going bigger container ships (Kim et al., 2021). Another study from 2021 used a bulk carrier as a subject vessel to investigate the application of ANN for fuel consumption prediction, which also showed promising results (Tran, 2021). These studies motivate this paper to emulate the use of ANN in the context of a harbour craft, specifically a tugboat.

The model developed by several researchers mostly used regression techniques to predict fuel consumption (Bocchetti et al., 2015; Kee et al., 2018; Meng et al., 2016; Uyanık et al., 2020; Wang et al., 2018) with the process simply considering historical fuel consumption occurrences and relating to historical operational and environmental factors. In practice, the historical operational, environmental, and fuel consumption parameters in time series may still have certain influences on the present fuel consumption rate. The time-series of the parameter factors are typically disregarded and difficult to be represented by traditional regression machine learning methods. Furthermore, the majority of the implementations of the machine learning models comprise merely of a single model, which could be further optimized through hybrid (combined) models (Liu et al., 2020, 2019; Wen et al., 2019) for better prediction accuracy.

In this paper, actual operational data from a tugboat operating in the Singapore Sea were collected. The operational data were obtained using several sensors installed onboard to capture parameters such as wind speed and direction, vessel location, and fuel consumption rate. Prior to model training, the raw operational data were cleaned and pre-processed. A long short-term memory (LSTM) time-series model was then developed, capturing historical parameters in time series, and forecasting future fuel consumption rates. In addition, by combining the classification LSTM model with the time-series LSTM model, a hybrid model was developed to further validate its capability in improving fuel consumption prediction compared to the single time-series LSTM model. The hybrid LSTM model equips the vessel operator with informed decision-making as to whether it is worthwhile to adjust the vessel speed with the aim to achieve fuel efficiency. Besides that, the proposed hybrid model used secondary data from classification methodology to associate the neighbouring data points to particular operational activities of the tugboat. This is a novel approach and is significant for tugboats as fuel consumption is not only affected by its speed and wind effect but also its operations whether in tugging or cruising. The ability of the hybrid model to identify the operational activities of the tugboat via cluster information (classes) will help inform the neural network model in producing better predictions. To the knowledge of the authors, there are limited works that utilised the hybrid LSTM model in predicting the fuel consumption of tugboats and thus the results presented here are novel and insightful on the effectiveness of the hybrid model in achieving fuel efficiency.

Fig. 1. Case study vessel, POSH Grace tugboat.

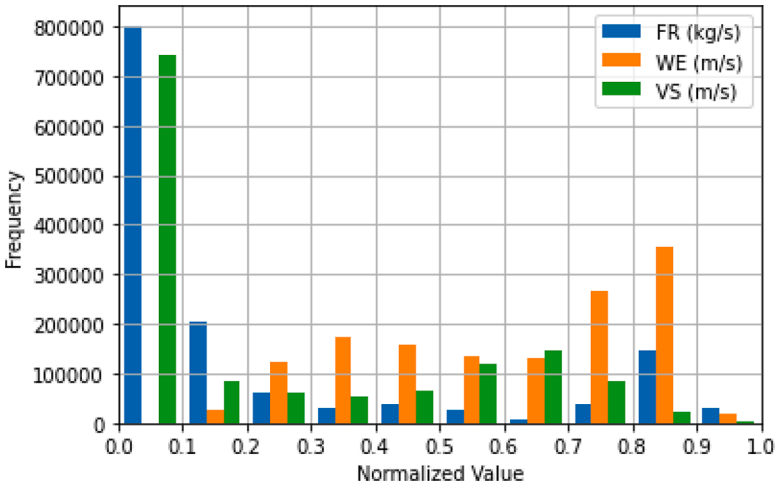


Fig. 2. Histogram of pre-processed and normalized data.

Collection of tugboat operational data

The operational data were collected from the data acquisition system installed onboard the traditional diesel-powered tugboat (vessel specifications given in Table 1) as shown in Fig. 1. The main operational activity of the tugboat consists of anchoring, assisting the large vessels in docking, and piloting around the southern sea of Singapore. The purpose of collecting the operational data was to conduct a research study on predicting the fuel rate to achieve fuel efficiency through data analytics and neural network modelling. To facilitate the collection of the operational data, two Coriolis mass flow meter sensors were installed to record the fuel consumption on both port and starboard main engines. The remaining two auxiliary engine fuel consumptions were not considered in this case study as the consumption rate is significantly smaller compared to the main engines. In addition, the auxiliary engines' fuel consumption is also highly predictable as they are consuming fuel at a constant rate over time. The vessel movement and speed were extracted through an automatic identification system (AIS) from Marine Traffic (www.marinetraffic.com). Lastly, the wind speed and directions were collected from the anemometer sensor onboard.

The operational data acquired over a five-month duration from May 2020 to September 2020 were used for this case study. Overall, the operational data composed of around 1.4 million data points were resampled to one-second intervals. The histogram of the pre-processed and normalized value of the fuel, wind and vessel speed data is shown in Fig. 2. Normalized value allows all parameters to be plotted and observed side by side, that otherwise would be out of one another's scale. The number of counts (frequency) of the tugboat fuel rate (*FR*) and vessel speed (*VS*) are dominated by low *FR* and *VS* values. High frequencies for low *FR* and *VS* values do not necessarily mean they occur at the same time. In contrast, the wind effect (*WE*) frequency has a more even spread being moderately concentrated towards the high side.

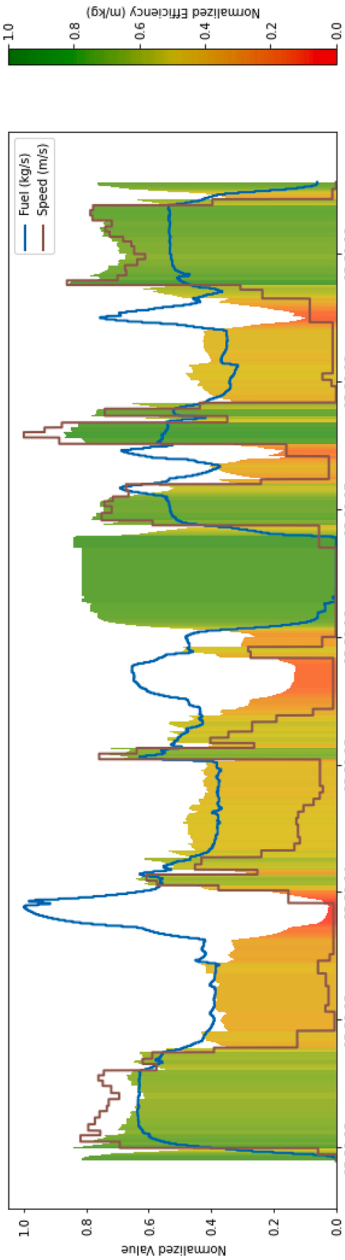


Fig. 3. Distance per fuel analysis. Sample data from year 2020.

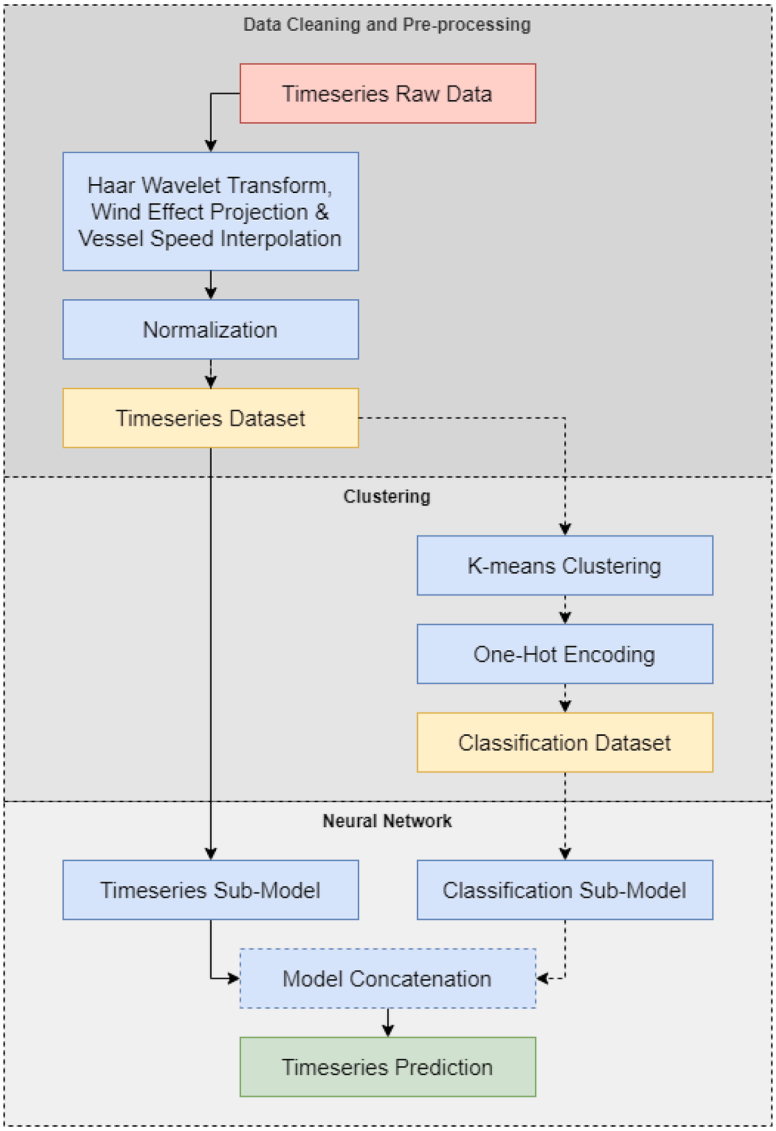


Fig. 4. Flow chart for combined model.

Methodology

In the automotive industry, it is common to use the miles per gallon (MPG) metric as a sole variable in gauging the efficiency of an automobile by measuring the distance travelled over a unit of fuel. However, in the context of the marine industry, it is not as straightforward as in the automotive industry because a marine vessel experiences greater environmental variables with more extreme conditions (i.e., wind and current) as compared to land vehicles. In addition, the operational characteristics of a certain marine vessel i.e., a tugboat, may differ from other vessels such as passenger vessels. E.g., the fuel consumption rate of a passenger vessel is directly influenced by the variation of the travelling speed and number of passengers. The fuel consumption of a tugboat could become high despite travelling at low or zero speed, influenced by the various operational activities such as tugging or towing. Therefore, the collected vessel's parameters have a mixture of signature values for various specific activities or operations. In view of this, the conventional linear regression models which associate the output with a single input are thus not an appropriate machine learning model to use. Instead, this paper uses a tool from computer science – an ANN model from a supervised machine learning toolkit – to create a future prediction based on the most recent history. The aim is to predict future fuel consumption given a limitation of available variables in the most recent history (i.e., data from environmental and vessel conditions). Environmental condition data such as *WE* (drag or push) provides useful information for the solution proposed in the paper. In contrast to the environmental condition, vessel condition such as *VS* is within the vessel operator's control. Therefore, it is possible to gauge the efficiency of the vessel by making a hypothetical prediction of future *FR* if an adjustment to *VS* is made.

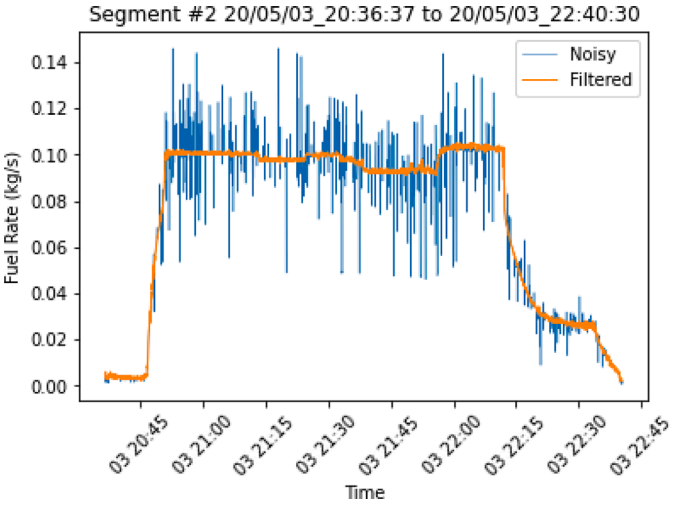


Fig. 5. Unfiltered fuel mass flow rate signal for two main engines.

The aforementioned future *FR* is represented by a new metric introduced here, i.e., the distance (in meters) per kilogram of fuel consumed (DPF). The DPF is similar to miles per gallon (MPG), except that it is in SI unit of meter per kilogram of fuel. Fig. 3 shows the DPF represented as a normalised efficiency value for the tugboat under investigation. Each *i*th DPF value is transformed to a normalized value by using the sigmoid function η_i given as,

$$\eta_i = \frac{1}{1 + e^{-\left(z_i|_{FR} \; - z_i|_{\;VS}\right)}} \tag{1}$$

where e is the Euler's constant. $z_i|_{FR}$ and $z_i|_{VS}$ are the standardized (scaled) values of *VS* and *FR*, respectively.

The equation to calculate the standardized values z_i for *FR* and *VS* is,

$$z_i|_{FR \;or\; VS} = \frac{(x_i - \mu)}{\sigma}\bigg|_{FR \;or\; VS} \tag{2}$$

where x_i is the *i*th-data point in the series (*VS* or *FR*). μ and σ are the mean and standard deviation of the series, respectively.

The normalised efficiency value for DPF shaded in green shows the sections where the tugboat is efficient (low *FR* and high *VS*), and in general, it is when the speed curve is above the fuel curve. On the other hand, the normalised efficiency value in red is the section where the vessel is less efficient (high *FR* and low *VS*). However, using the normalised DPF metric alone is not entirely ideal, as the tugboat's productive time is when it is in tugging/towing operation where the vessel speed is relatively low or zero speed. Hence, the shaded sections in red when fuel is not equal to zero but at high *FR* (shown by vertical black dashed lines in Fig. 3) may have been the times when the tugboat is most productive. Given this, only the towing power alone which uses a considerable amount of fuel is the most important, and both fuel consumption and distance travelled become irrelevant. Despite the limitations of the DPF metric in certain operations where producing a certain towing power is critical and a must (i.e., assisting, bollard pulling), there is still an opportunity for fuel optimization when the tugboat is in motion such as cruising or transporting/towing.

In addition to the ANN model, this paper also uses another tool from computer science, i.e., unsupervised machine learning (clustering), to synthesize secondary data related to the vessel's activity. For example, data points from tugging jobs will be clustered together, while data from cruising will be in a different cluster. Therefore, each data point is associated with a cluster or class label. This process is known as classification which generates secondary data highly related to vessel operation. The time-series data combined with the classification data could be used to train an ANN model.

The block diagram of the methodologies is presented in Fig. 4. Blocks in red and green are the original raw unprocessed data and final prediction respectively. The yellow blocks are the intermediate datasets, while the blue blocks are the processes to consume relevant datasets and generate the final predictions.

Data filtering and pre-processing

The raw data collected from the tugboat are the fuel mass flow rate, wind speed and direction, and travelling distance. The data must be filtered, pre-processed, and transformed into a time-series dataset prior to being utilized in both unsupervised and supervised machine learning models (see Fig. 4). The various methods of data filtering, pre-processing, and transformation are presented hereafter. The purpose of this section is to create a dataset of the aforementioned three parameters that are relatively free from noise and are of uniform period (1-s interval).

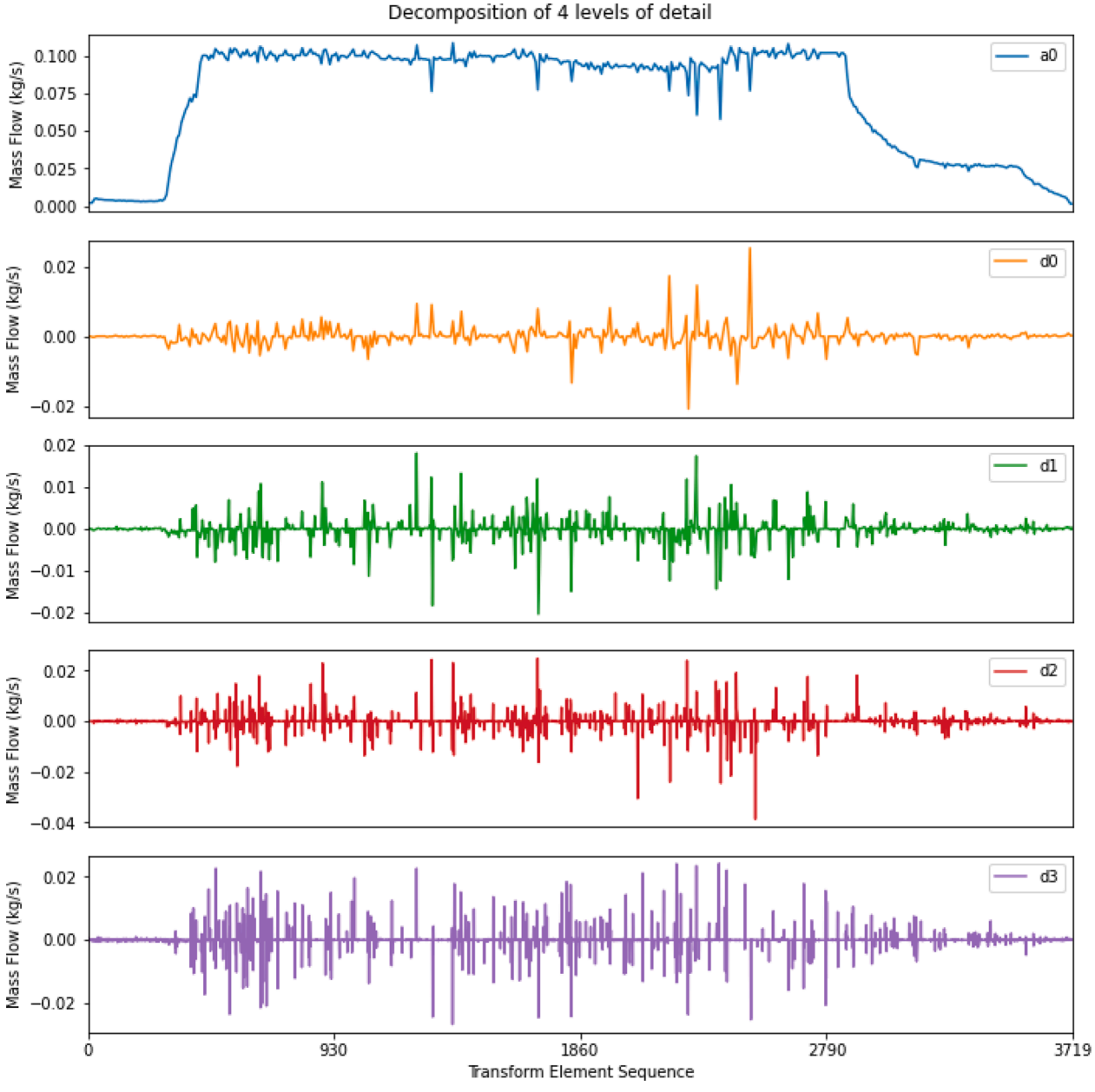


Fig. 6. Decomposition of signal using HWT.

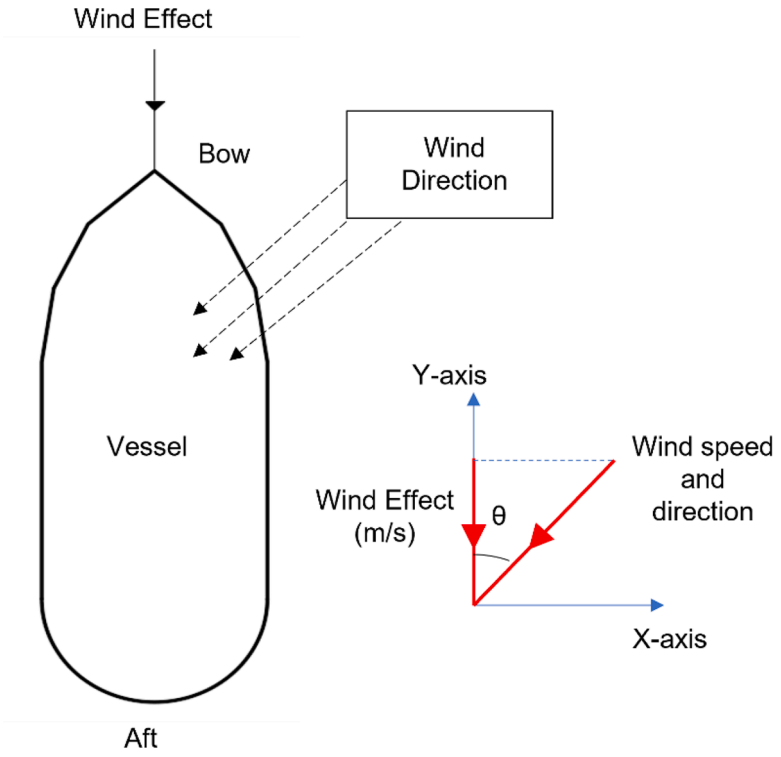


Fig. 7. Encountering wind speed and direction on the vessel.

Fuel mass flow rate filtering

The data acquisition system records the fuel's mass flow rate data measured by the mass flowmeters at the fuel line feeding the two main engines. The unit is in kilogram per second. Fig. 5 shows a sample of both original (noisy) and filtered data from one segment or one tugboat job. There are predominantly noisy data that appear like spikes as shown in Fig. 5. The exact cause of the noise is not part of the scope of this paper. Nonetheless, two possibilities for the occurrence of noise are due to mechanical vibration and electromagnetic interference to mass flowmeter that cause invalid data points. These invalid data points, therefore, must be removed using filtering techniques. This paper uses Haar Wavelet Transform (HWT) due to its simplicity and effectiveness in filtering out noise (Hadi et al. 2022).

Haar wavelet transform. The Haar wavelet transform (HWT) is a discrete wavelet transform that allows the representation of an unfiltered time-series signal in a waveform to be decomposed into multiple levels of details (Chaovalit et al., 2011). The whole process of the HWT is applied through a sliding window. The outcome of the sliding window throughout the unfiltered signal produces filtered data as shown in Fig. 5. The fuel data is also referred to as fuel rate (*FR*) throughout this paper. The unfiltered signals are decomposed as input vectors $\psi(t)$ algorithm using a constant sliding window of size t_w given in Eq. (3) as,

$$\psi(t) = \begin{cases} 1, & 0 \le t < \frac{t_w}{2} \\ -1, & \frac{t_w}{2} \le t < t_w \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

The decomposition produces a transform where the signals are decomposed into multiple levels according to the levels of details as shown in Fig. 6. The deepest level of detail (d3) is suppressed aggressively, and the subsequent shallower levels of details (i.e., d2 to d0) are suppressed less aggressively. The average (a0) is left untouched as it carries the most information. Afterwards, the modified signals are inversed-transformed back to the fuel consumption domain (kg/s).

Wind effect projection

The wind speed and direction collected from the anemometer sensor are resampled to one-second intervals to match the size of the fuel data. The wind data consists of wind speed in meters per second and wind direction ranging from 0 to 360° measured clockwise from the vessel bow (*Y*–axis) shown in Fig. 7. The wind speed is defined as the intensity of the wind's frontal impact on the vessel hull. In this context study, the vessel is always assumed to be travelling forward in *Y*-axis direction. Hence, only the prevailing wind speed and direction along the *Y*-axis, termed the wind effect (*WE*), is considered in the study. The calculation of the *WE* is given as,

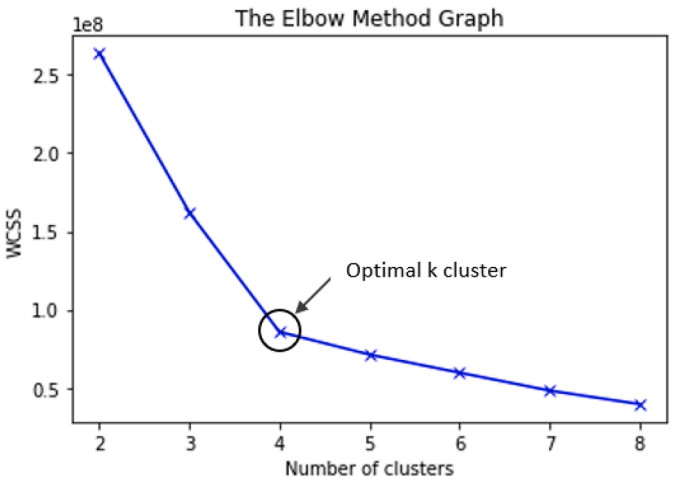


Fig. 8. Optimal cluster selection using elbow method.

$$WE = \sin(wd - 270^{\circ}) \times ws \tag{4}$$

where *wd* is the wind direction and *ws* the wind speed. The wind speed is projected to the bow-aft axis using wind direction. The positive *WE* value indicates a drag effect by headwind, whereas the negative value indicates a push effect by a tailwind. It is to note that the *WE* neglects the aerodynamic and other fluid dynamic factors.

Vessel speed interpolation

The travelling distance is calculated using the latitude and longitude coordinates of the tugboat position obtained from Marine Traffic (www.marinetraffic.com). The tugboat's positional coordinates are useful to measure the vessel speed, albeit in a coarse resolution. The travelling distance at a specific interval between the coordinates may be represented as the vessel speed (*VS*), which is determined based on the distance travelled over time measured at one-second intervals. For coordinate data intervals that are irregular or longer than a one-second interval, the *VS* is upsampled to a one-second interval using linear interpolation.

Normalization

To facilitate the discussion in the Result and Discussion section, the three parameters, i.e., *FR*, *VS*, and *WE* are normalised into a range from 0 to 1. The function to normalise the parameters is given in Eq. (5).

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{5}$$

where x_{scaled} is the scaled single data value, x the unscaled single data value, x_{max} the maximum value in the data and x_{min} the minimum value in the data. The purpose of normalization is to rescale all data points to the same value range while maintaining the ratio of separation/distance from one another within the same parameter. In other words, the maximum value of a parameter is transformed to a value of 1, and the minimum is transformed to 0, while other values are between 0 and 1.

k-means clustering

k-means clustering is arguably the most popular clustering method. Note that it is possible to use other clustering methods instead of *k*-means clustering. The purpose of the clustering step in this paper is to isolate and classify the knowledge base of a certain range of ship parameters. E.g., if the ship is not travelling very fast and experiencing a lot of headwinds (*VS* and *WE* values are low), while the *FR* is high, intuition tells that the ship is trying to travel against strong wind. Scenarios like this are isolated and withheld as a cluster. Scenarios/clusters from this example are separated from other clusters such as the bollard pull operation that causes the ship to travel at a very slow speed regardless of the *WE* with high fuel consumption (*FR* is high, *VS* is low). Hence, the cluster information would give a classification or association to a certain operation.

The normalized *FR*, *WE* and *VS* data are scattered into a three-dimensional cartesian coordinate system, forming points in 3D space. Each data point will be associated with other data points within its vicinity to form clusters. A cluster is given an arbitrary class label which is the output of a clustering process. The output from clustering is essential for training the neural network classification sub-model later. The user specifies the number of clusters generated (i.e., *k*) and the optimum cluster size is determined via elbow and silhouette analysis to determine the effectiveness of the *k* values in evaluating the clusters.

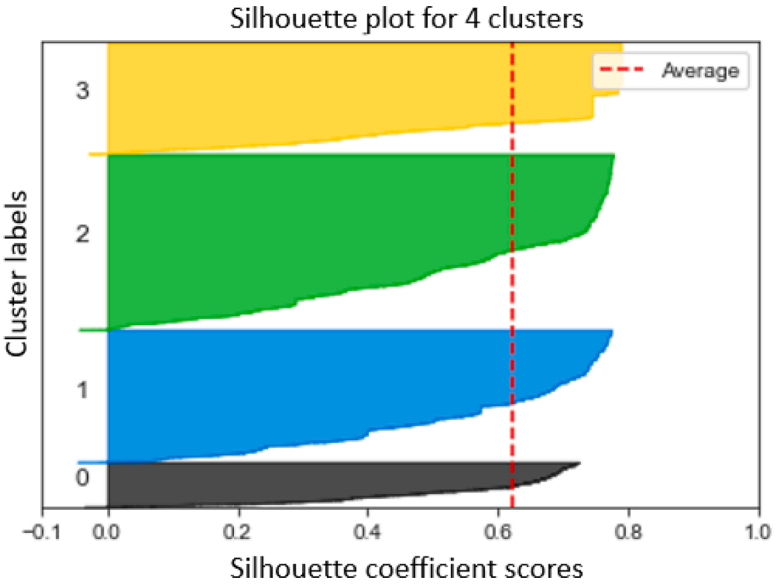


Fig. 9. Visual example of how silhouette score is calculated.

Elbow analysis method

The elbow analysis identifies the optimal size of the cluster k by evaluating the cluster's association with its decreasing error computed using the within-cluster sum squared (*WCSS*) values (Syakur et al., 2018) given as,

$$WCSS = \sum_{i=1}^{n} (x_i - c_i)^2 \tag{6}$$

where x_i is the ith data point of the cluster member and c_i the cluster's centroid.

In this process, the k value is gradually increased until the decline in error, i.e., *WCSS* is gradual and stable. The selection of the optimal k value is demonstrated in Fig. 8 for the parameters data. It can be seen that the cluster value declines from $k = 2$ to 5 and with a sudden and abrupt decrease in the *WCSS* value at $k = 4$, forming an elbow (sample shown in Fig. 8). $k = 4$ is thus the most optimal number of clusters for the parameters data. It is to be noted that from the *WCSS* in Eq. (6), the *WCSS* decreases with the increase in the cluster size k. However, taking the lowest *WCSS* value does not necessarily imply that the cluster is optimal; rather, it may lead to overfitting. Furthermore, the selection of k values using the elbow method is ambiguous since the sharp fall in *WCSS* is not always apparent. The drawback of the elbow method can be mitigated through silhouette analysis discussed in the next section.

Silhouette analysis method

The silhouette coefficient scores for various clusters k are obtained by using Eq. (7).

$$s(i) = \frac{b(i) - - a(i)}{max[\ a(i),\ b(i)]} \tag{7}$$

where a is the average intra-cluster distance, b the mean nearest cluster distance, and s the mean silhouette coefficient for the i^{th} input data.

The silhouette analysis selects the correct number of clusters (k) based on the highest average silhouette scores $s(i)$ (Rousseeuw, 1987). The most optimal k is identified with the highest $s(i)$ value, which is the same as the elbow method, i.e., $k = 4$. The silhouette technique takes into consideration overfitting in a cluster as k increases. Eq. (7) shows that a well-defined cluster having high silhouette values will have a greater mean closest cluster distance value and a smaller intra-cluster distance value. Fig. 9 shows an example of how the silhouette score is calculated for a cluster of four (clusters are labelled as 0 to 3). Each cluster member's silhouette coefficient is average to make up the silhouette score.

In the silhouette analysis method, the identification of the optimal k is relatively easier as compared to the elbow analysis method, as the former is determined by the highest silhouette scores, whereas the latter is by the identification of a drastic decrease in the *WCSS* which in some cases are not always apparent.

One-hot encoding

One-hot encoding is often used in classification. It is a binary representation of a class or a category. However, unlike binary numbers, only one bit may have a value of 1, while the other bits must contain a value of 0. Its length is dictated by the number of

Table 2
One-hot encoding for 8 classes.

Decimal	Binary	One-Hot Encoded
0	0000	00000001
1	0001	00000010
2	0010	00000100
3	0011	00001000
4	0100	00010000
5	0101	00100000
6	0110	01000000
7	0111	10000000

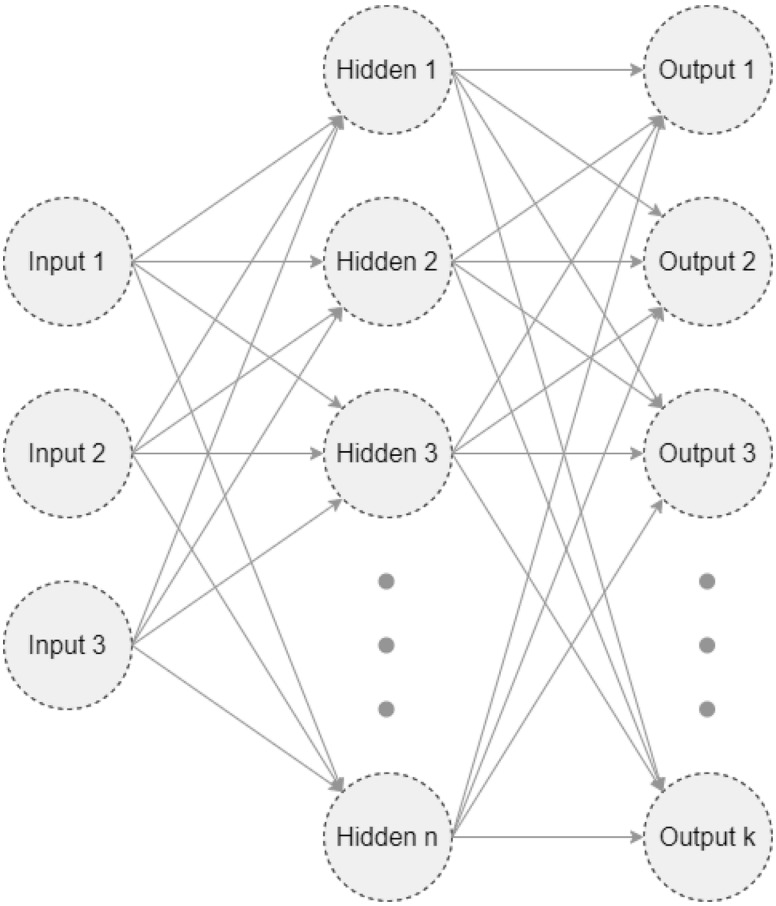


Fig. 10. Dense neural network architecture model.

unique categories. All members of the same category will have the same positional bit set as 1. Table 2 shows an example of one-hot encoding. The classification (clustering) output is one-hot encoded to form a classification dataset.

Classification sub-model

The ANN mimics the actual working of biological neural networks. It is a set of algorithms that uses a tensor (similar to a matrix) stacked as layers. These tensors, or layers, hold coefficients that are obtained after completing the training. Element in the layers can be likened to biological neurons and is also named as a neuron in the field of computer science. The tensor layers could be arranged one after another or branched to enable the tensor arithmetic operations to be performed between them. The set of coefficients in the tensor is used by a general function for each layer known as the activation function. Using the analogy of wisdom of the crowd – where the crowd is the group of activation functions – a well-designed and well-trained ANN could produce a very good approximation to a solution (also known as ground truth).

The classification sub-model (see Fig. 4) uses ANN. The classification sub-model aims to simulate *k*-means clustering in the neural

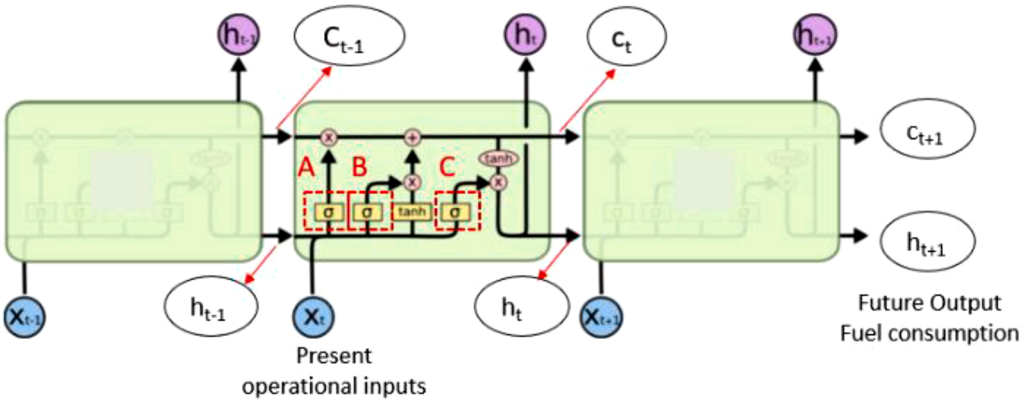


Fig. 11. LSTM network architecture model.

network. In Fig. 4, the output from *k*-means clustering is the cluster information to which each time-series data point is associated. The cluster information or the class label is used in training the classification sub-model. The classification sub-model uses a neural network with fully connected layers to achieve its aim, as shown in Fig. 10. The fully connected layer is also known as a dense layer. This sub-model accepts coordinate points in 3D space as training input, to predict the class label.

The number of inputs to the classification sub-model is three (one for each *FR*, *WE*, and *VS*), and the number of outputs is the number of class labels (*k*). The number of nodes in the hidden layer could be quite arbitrary. It is determined by exhausting a series of different values to achieve optimum *n*, which will be discussed in the Comparison of Results section.

Time-series sub-model

The time-series sub-model is an ANN model that uses the time-series data. Markers are placed at regular intervals in the time-series data. The most recent history data in the order of seconds from each marker are set as training input, and subsequent data in the order of seconds to the future from each marker are set as training output. The collection of input and output is used to train the neural network sub-model. This sub-model uses two layers of the LSTM layer.

Long short-term memory (LSTM)

The long short-term memory (LSTM) model networks (see Fig. 11) are well-suited to classifying, processing and making predictions based on time-series data since there can be lags of unknown duration between important events in a time series (Lianne and Justin, 2020). The LSTM architecture comprises three distinct gates, namely the forget gate (A), input gate (B) and output gate (C). Unlike the ANN model, the LSTM model has its specified activation neural network layers embedded inside it where the number of neural network layers could be added repeatedly. The LSTM model is effective when knowledge about the previous values has a substantial effect on the present values (Zhu et al., 2020).

Forget gate. For the forget gate (A), h_{t-1} is the hidden state vector (one of the output vectors of the LSTM unit). X_t is the input vector into the forget gate activation vector f_t computed based on the sigmoid activation function σ_s, which has an output value between 0 and 1. This helps in determining whether to discard the information retained in the cell state C_{t-1} originally as shown in Eq. (8),

$$f_t = \sigma_s\left(W_f \,.\, [h_{t-1},\ X_t] + b_f\right) \tag{8}$$

where W_f is the weight of the layer and b_f the bias of the layer with the subscript f represents the forget gate activation vector. It is to note that the subscript for W and b represents the different gate activation vectors, i.e., i, c, o and h for the input/update gate, current state, output state and hidden state activation vectors, respectively given in the subsequent Eqs. (9)–(11).

Input gate. In the input gate (B), new input information is determined through the sigmoid activation function σ_s in Eq. (9) to obtain the input/update gate activation vector i_t. Subsequently, the hyperbolic tangent activation function σ_t shown in Eq. (10) is used to convert, vectorize and update the new input information into the current cell input activation vector $c_t^{'}$.

$$i_t = \sigma_s(W_i.[h_{t-1}, X_t] + b_i) \tag{9}$$

$$c_t^{'} = \sigma_t(W_c.[h_{t-1}, X_t] + b_c) \tag{10}$$

At the end of the process of the input gate, the final information is updated into the current cell state by multiplying f_t with C_{t-1} to discard the previous information as given in Eq. (11).

$$c_t = f_t \,.\, c_{t-1} + i_t \,.\, c_t^{'} \tag{11}$$

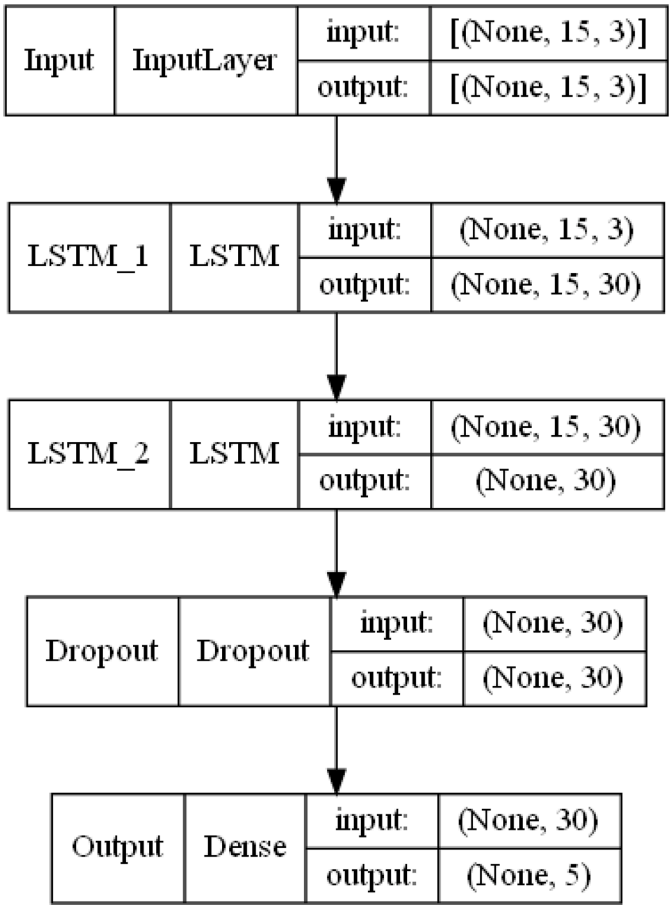


Fig. 12. Linear model.

Output gate. Finally, the output gate (C) decides which information to output depending on the cell state and sigmoid activation function σ_s multiplied by the hyperbolic tangent activation function σ_t. The intuitive idea is based on the previously learnt past information and predicts the information that is relevant for the future output state vector o_t and hidden state vector h_t given in Eqs. (12) and (13).

$$o_t = \sigma_s(W_o[h_{t-1}, x_t] + b_o) \tag{12}$$

$$h_t = o_t .\sigma_t(C_t) \tag{13}$$

In this case study, the historical fuel consumption rate is also included as an input into the LSTM model to predict the future fuel consumption rate. In addition to the LSTM layer, the dropout layer is also used to disable portions of the output from the LSTM network. The dropout rate is discussed in the Comparison of Results section.

Evaluation criteria

The R squared score ($R2$) is chosen as an assessment criterion for evaluating model prediction accuracy.

$$R2 = 1 - \frac{SS\ Error}{SS\ Total} = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y}_i)^2} \tag{14}$$

where *SS Error* is the sum of the square of the regression error and *SS Total* the sum of square of the total error. The *SS Error* is computed as the difference between the summation of squares of actual and predicted values whereas the *SS Total* is computed as the difference between the summation squares of actual and average values.

The mean absolute error (*MAE*) is adopted in model performance evaluation and as a training loss function metric to measure the error to update the weights of each neuron in the LSTM or ANN until a converging solution in the LSTM layer is sought.

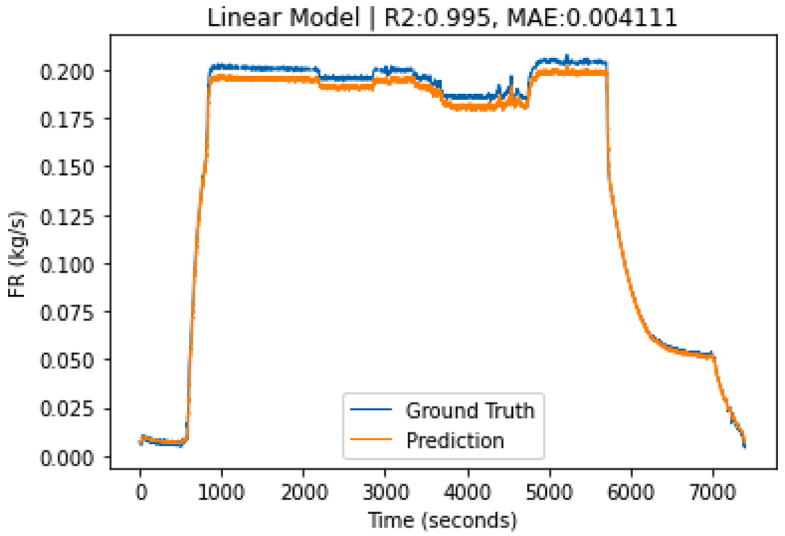


Fig. 13. Linear model result.

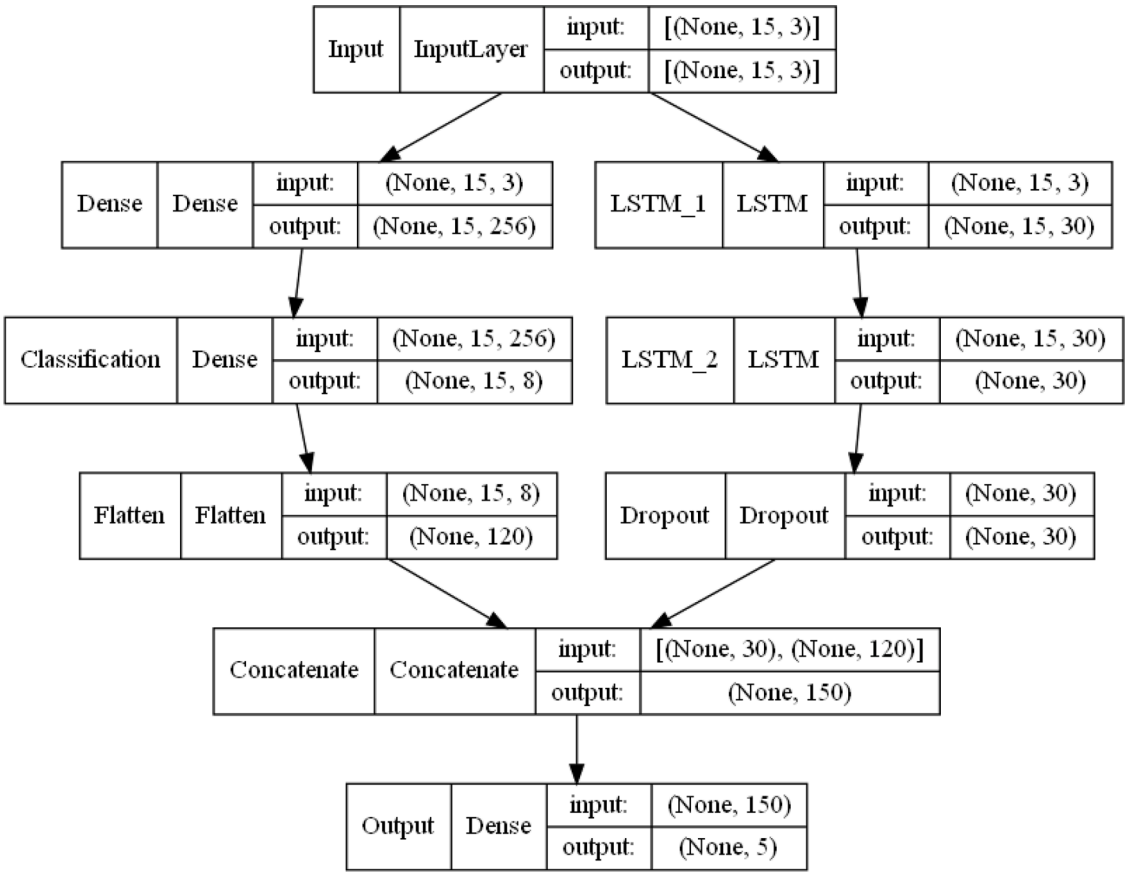


Fig. 14. Combined model.

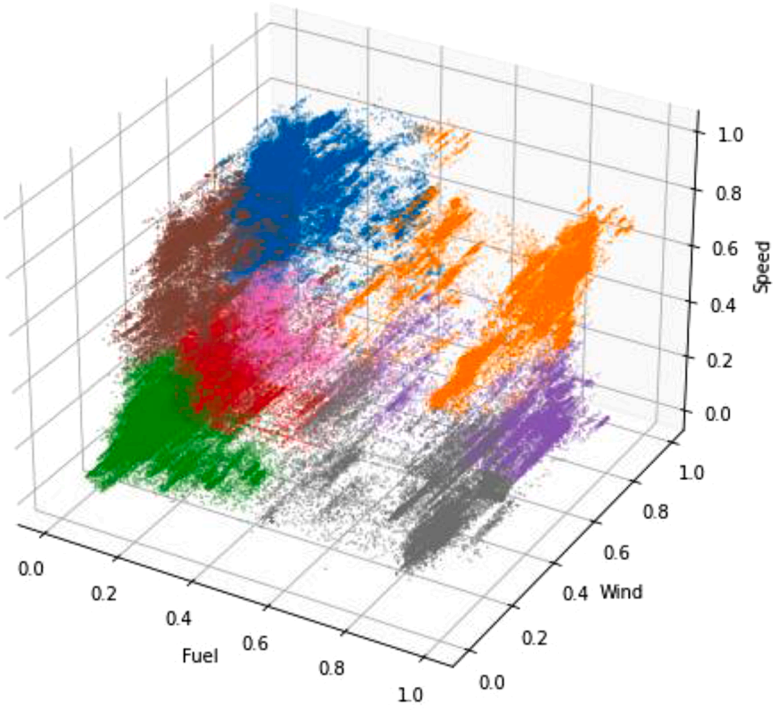


Fig. 15. Visualized clustering output ($k = 8$).

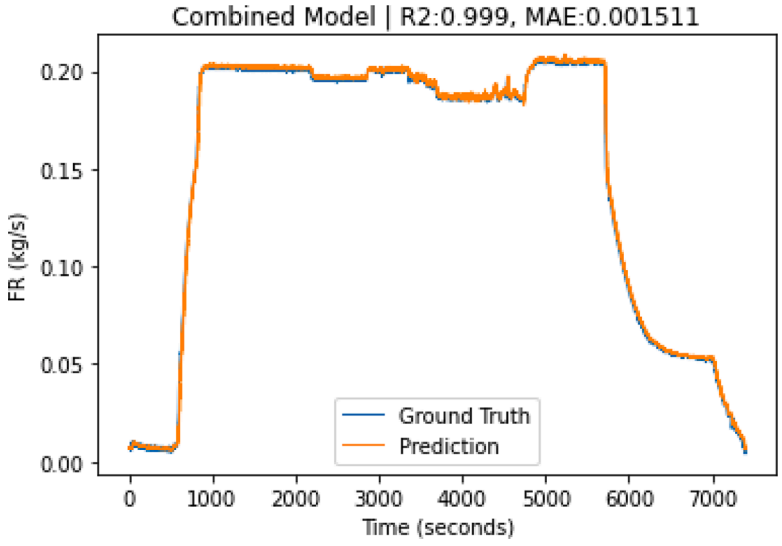


Fig. 16. Combined model result.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y}_i| \tag{15}$$

where y_i are the actual values for the ith observation, $\widehat{y}_i$ the predicted values for the ith observation and n the total number of samples in the dataset. The *MAE* is computed based on the summation of absolute values in the deviation of the actual values from the predicted values to the number of samples n considered in the analysis.

Table 3
Comparison of results with data from 15 s in the past to predict next 5 s in the future.

No.	Architecture	LSTM	Dropout	Time[1]	R2	MAE
1	Linear	15	0.1	6	0.973	0.010264
2	Linear	15	0.25	9	0.984	0.007603
3	Linear	15	0.5	12	0.982	0.008393
4	Linear	30	0.1	8	0.995	0.004468
5	Linear	30	0.25	9	0.997	0.002997
6	Linear	30	0.5	9	0.989	0.006424
7	Linear	60	0.1	11	0.998	0.002756
8	Linear	60	0.25	16	0.999	0.002134
9	Linear	60	0.5	12	0.996	0.003638
10	Combined	15	0.1	12	0.999	0.001326
11	Combined	15	0.25	16	0.999	0.001676
12	Combined	15	0.5	16	0.999	0.001241
13	Combined	30	0.1	21	0.999	0.001122
14	Combined	30	0.25	18	0.999	0.001618
15	Combined	30	0.5	50	0.999	0.001671
16	Combined	60	0.1	16	0.999	0.001407
17	Combined	60	0.25	20	0.999	0.001567
18	Combined	60	0.5	18	0.999	0.001743

[1] Processing time in seconds

Table 4
Comparison of results with data from 30 s in the past to predict next 10 s in the future.

No.	Architecture	LSTM	Dropout	Time[1]	R2	MAE
1	Linear	15	0.1	9	0.978	0.009175
2	Linear	15	0.25	15	0.991	0.006019
3	Linear	15	0.5	14	0.969	0.011040
4	Linear	30	0.1	13	0.994	0.004818
5	Linear	30	0.25	11	0.992	0.005646
6	Linear	30	0.5	15	0.993	0.005260
7	Linear	60	0.1	16	0.998	0.002265
8	Linear	60	0.25	16	0.998	0.002784
9	Linear	60	0.5	15	0.996	0.003490
10	Combined	15	0.1	14	0.997	0.001733
11	Combined	15	0.25	14	0.997	0.002236
12	Combined	15	0.5	15	0.998	0.002065
13	Combined	30	0.1	19	0.998	0.001655
14	Combined	30	0.25	20	0.998	0.001321
15	Combined	30	0.5	19	0.998	0.002133
16	Combined	60	0.1	23	0.998	0.001657
17	Combined	60	0.25	32	0.997	0.001725
18	Combined	60	0.5	17	0.998	0.001841

[1] Processing time in seconds

Table 5
Comparison of results with different number of neurons in dense layer.

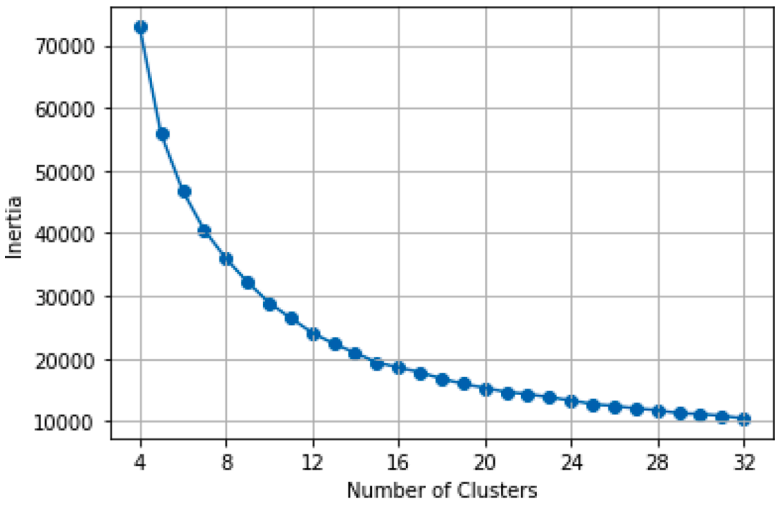
No.	Architecture	Cluster	LSTM	Dropout	Dense	Time[1]	R2	MAE
1	Combined	8	30	0.1	4	42	0.999	0.001188
2	Combined	8	30	0.1	256	21	0.999	0.001122
3	Combined	8	30	0.1	1024	30	0.999	0.001419

[1] Processing time in seconds

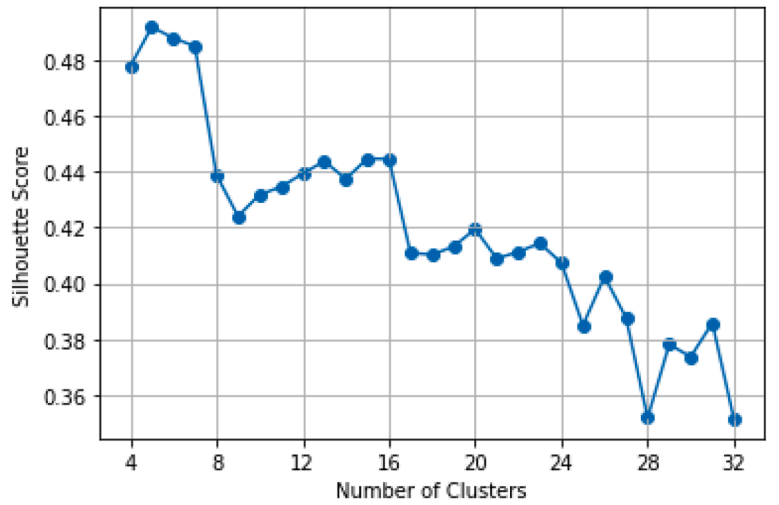
Table 6
Comparison of results with different number of clusters (class labels) in dense layer.

No.	Architecture	Cluster	LSTM	Dropout	Dense	Time[1]	R2	MAE
1	Combined	4	30	0.1	256	51	0.999	0.001798
2	Combined	8	30	0.1	256	21	0.999	0.001122
3	Combined	32	30	0.1	256	13	0.999	0.001445

[1] Processing time in seconds



(a) Elbow method number of clusters versus Inertia (WCSS)



(b) Silhouette score number of clusters versus silhouette score

Fig. 17. Cluster analysis of various number of clusters.

Result and discussion

This section presents the results from linear and combined model architectures. In addition, this section discusses the comparison between the two architectures with various configurations. Finally, the tugboat operational analysis Section explains one case of how the prediction model can be applied.

Linear (non-combined) model

In Fig. 4, the linear model is indicated by all the blocks except the ones pointed by dashed arrows and bypassing model concatenation. In other terms, it is a prediction model without clustering information. The model only uses a time-series dataset to train and generate the prediction. The time-series sub-model architecture is shown in Fig. 12. The *InputLayer* input accepts 15 sequential points of *FR*, *WE* and *VS* as training input. The *Output* accepts the next five sequential points for *FR* as training output. Once, the training is completed, the model can generate five future data points of *FR* from the most recent 15 data points of *FR*, *WE*, and *VS*.

The output of the linear model prediction versus ground truth is shown in Fig. 13. In Fig. 13, the prediction can trace the ground truth quite well, especially at lower values. However, at higher values, there is a separation between the prediction and ground truth. The separation contributes to lower *MAE* and *R*2 metrics than the combined model shown in Fig. 16 which will be discussed later in the Comparison of Results section.

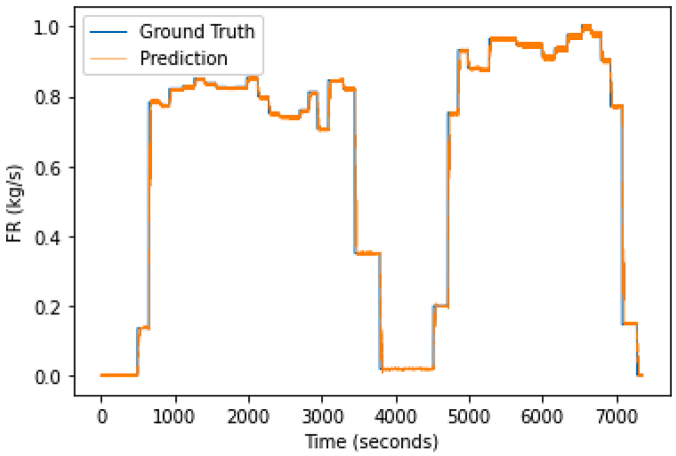


Fig. 18. VS (vessel speed) prediction.

Combined model

The combined model uses all the processes and datasets shown in Fig. 4. In other words, it uses both a time-series dataset and a classification dataset to train and make the prediction. The architecture of the combined (branched) model is shown in Fig. 14. Similar to the linear model, the *InputLayer* input uses 15 sequential points of three parameters (*FR*, *WE* and *VS*) as training input. The *Output Dense*, at the end of the figure, is trained using the next five sequential points of *FR* from the earlier 15 points of input. After training, the model can predict the next future five data points of *FR* given the history of the recent 15 points of *FR*, *WE* and *VS*.

Dataset from *k*-means clustering is used as an output to train the *Classification Dense* layer, while input uses the same time-series data for both classification sub-model (left branch) and time-series sub-model (right branch). Fig. 15 shows all the data points scattered in 3D space. The sequence information is removed while clustering, only the positional coordinates in space are represented. After applying *k*-means clustering, the eight class labels clustered in accordance with operational activities are shown in different colours. The operational activities of the vessel could be deduced from the *FR*, *WE* and *VS*. E.g., Fig. 15 shows that groups of data points are clustered in orange when both the *FR* and *VS* are at moderate to high levels. These orange clusters represent the data points when the vessels are in cruising operation. In contrast, data points with moderate to high levels of *FR* but relatively low *VS* and *WE* are clustered in grey in Fig. 15. These grey clusters represent the data points when the vessels are in tugging operation. By using the classification methodology, these classes (cluster information) of operational activity help to inform the neural network model in producing a better prediction of fuel consumption. The class labels are then one-hot encoded to make up the classification dataset.

The sample result from the combined model is shown in Fig. 16. It is immediately obvious that the issue of separation at higher values with a linear model is solved.

Comparison of results

Table 3 presents the results for the prediction of five future data points (seconds) using the past 15 data points. There are 18 combinations of training configurations such as architectures (Architecture), number of neurons in LSTM layers (LSTM), and dropout rates (Dropout). The *MAE* and *R*2 metrics, as well as processing time (Time) to train and predict, are also shown. The training processing time is produced using a computer with Intel 10th generation i7 CPU with 16 GB of RAM and Nvidia RTX 2080 GPU.

The combined models outperform the linear models in general according to both metrics. Among the combined models, result #13 is the best performing model. Result #13 indicates that the high number of LSTM neurons does not necessarily improve the model performance despite no harm from having a higher neuron count. In addition, having to drop out (to disable) too many connections from LSTM layers lowers the performance.

Table 4 discusses the performance of the same combinations to predict the future ten data points using the past 30 data points. In general, the linear models continue to show inferiority to the combined models. The combined models with 15 LSTM neurons slightly underperform combined models of 30 and 60 LSTM neurons.

Combined models in Tables 5 and 6 use 15 past data points to predict five future *FR* data points. Table 4 discusses the performance of the number of neurons (n) of hidden layers in *Classification Dense* layer. All three models perform comparably well. The higher number of n does not necessarily produce better results. Although the higher number of n causes the model to take a longer time to train.

Table 6 discusses the model performances with a varying number of clusters. Fig. 17 is the cluster analysis of clustering from Fig. 15. The recommended number of clusters using the elbow method is inconclusive as visually there is no apparent elbow edge. While using silhouette score analysis, the highest score is when the number of clusters is 4 to 7, and the scores decrease as the number of clusters increases. However, the better silhouette score by $k = 4$ does not necessarily produce a better model performance as shown

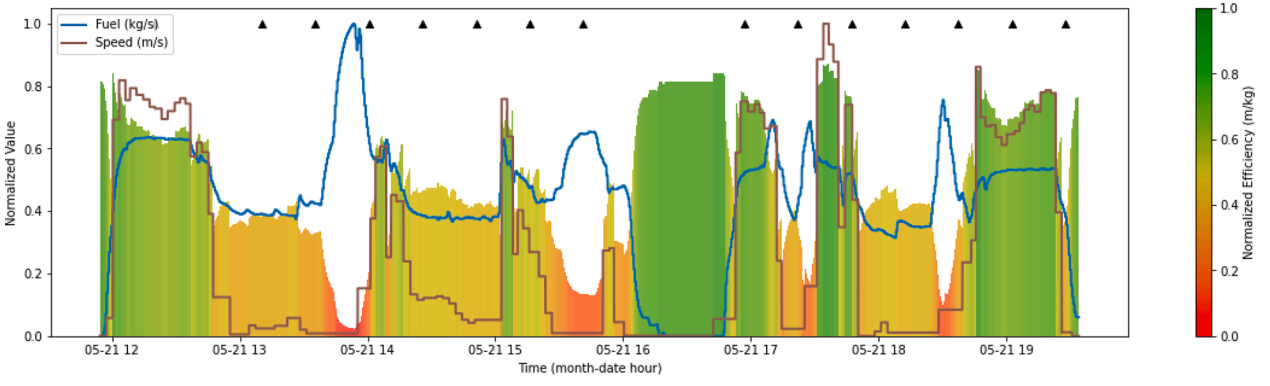


Fig. 19. Recommendation from VS prediction model. The black triangular markers above the plot suggest if increasing VS is worthwhile at respective times.

in Table 6. All three models also perform relatively well from one another, despite a lower number of clusters (k) taking a longer time to train. Although a higher number of k requires a shorter processing time, it takes a significantly longer time to produce the classification training dataset during clustering. The larger the number of clusters (k), the more iterations it requires to complete the clustering process. It is due to that there being more clusters centres for each data point to calculate to which one it is the nearest.

Tugboat operational analysis

The fuel consumption (*FR*) prediction is a valuable tool to project *FR* for the next few moments into the future, given the current situation. The projection is a piece of information that enables the vessel operator to monitor the expected *FR*. However, such information may not be explicit enough for the vessel operator to take concrete action. The vessel operator could face a situation to decide whether to increase the vessel speed or to maintain the current speed, represented by using the binary option, i.e. 0 or 1. More explicit information from the prediction model could allow the vessel operator to make better decisions.

The same methodologies could be used to create a prediction model for *VS* as shown in Fig. 18, instead of a prediction for *FR*. By artificially increasing and/or decreasing *FR*, the *VS* prediction model would be able to predict the future (hypothetical) *VS* value. The hypothetical *VS* over artificial *FR* produces the hypothetical *DPF*. The hypothetical distance per fuel metric could be compared with the real-time one, which could determine if increasing *VS* is worthwhile.

Fig. 3 shows the relationship of distance over fuel over one segment of towing work. As explained earlier, the sections/areas in green are when less fuel is consumed to move the tugboat. Hence, it is the low-intensity operations/zones or low-stress zones. In contrast, the areas in red are the high intensity/stress zones when more fuel is used while there is not much or no tugboat movement. The orange/yellow areas are the medium intensity/stress zones.

The low-intensity zones at both ends of the plot, at around 1200 and 1900 hour, could have been the times when the tugboat was on cruising operation to and from the worksite. The transporting/towing work could have been around 1400 and 1500 hour. During these operations, the hypothetical distance over fuel metric would become beneficial to determine if the tugboat is travelling at optimum speed. In Fig. 19, the black uptick triangular markers derived from the *VS* prediction model based on the artificial *FR* increment are added to Fig. 3 and they suggest the condition when it is worthwhile to increase the *VS*. The presence of the uptick markers informs the vessel operator that increasing *VS* is worthwhile and vice versa. This binary information could help vessel operators to achieve optimum vessel speed.

Conclusion

Both supervised and unsupervised machine learning methods based on the LSTM and k-means clustering were used in the prediction of fuel consumption to achieve fuel efficiency. The fuel and wind data were collected from the vessel onboard and the vessel speed is obtained from the vessel tracking website – Marine Traffic. The fuel (*FR*) prediction provided a projection of near-future fuel consumption information which is a valuable tool to anticipate fuel usage. On the other hand, the speed (*VS*) prediction (and subsequent suggestion) could relieve the vessel operators from a guessing game by providing valuable information in real-time for easier decision-making.

Findings in the Result and Discussion section demonstrated that the future prediction could be made for any parameter of the inputs. It is especially useful to predict future increases in vessel speed based on a hypothetical increase in the current fuel rate. The ratio of the future prediction of vessel speed and the hypothetical increase of fuel rate could become a valuable efficiency indicator. A higher efficiency would suggest to the vessel operator that increasing the fuel rate is a worthwhile action.

It is important to note that the methodologies and solutions offered in this paper do not directly optimize fuel usage. The main idea motivating this paper is to empower vessel operators with information to make informed decision-making to optimize fuel usage. The quality and timeliness of the information allow the vessel operator to make a proactive decision rather than a reactive one. The solution presented in this paper may also be adopted for other types of harbour crafts .

Acknowledgement The Authors wished to acknowledge the resources supported by SMI (R-SMI-A403- 693-0001) and MOE (R-MOE-A403-C002/MOE2018-TIF-1-G-008).

Declaration of Competing Interest

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

Acknowledgements

The Authors wished to acknowledge the resources supported by SMI (R-SMI-A403- 693 0001) and MOE (R-MOE-A403-C002/MOE2018-TIF-1-G-008).

References

Bialystocki, N., Konovessis, D., 2016. On the estimation of ship's fuel consumption and speed curve: a statistical approach. J. Ocean Eng. Sci. 1, 157–166. https://doi.org/10.1016/j.joes.2016.02.001.

Bocchetti, D., Lepore, A., Palumbo, B., Vitiello, L., 2015. A statistical approach to ship fuel consumption monitoring. J. Sh. Res. 59, 162–171.
Chaovalit, P., Gangopadhyay, A., Karabatis, G., Chen, Z., 2011. Discrete wavelet transform-based time series analysis and mining. ACM Comput. Surv. 43 https://doi.org/10.1145/1883612.1883613.
Fam, M.L., Tay, Z.Y., Konovessis, D., 2021. An Artificial Neural Network based decision support system for cargo vessel operations. Proceedings of the 31st European Safety and Reliability Conference 19–23.
Gkerekos, C., Lazakis, I., Theotokatos, G., 2019. Machine learning models for predicting ship main engine fuel oil consumption: a comparative study. Ocean Eng. 188, 106282 https://doi.org/10.1016/j.oceaneng.2019.106282.
Hadi, J., Tay, Z.Y., Konovessis, D., 2022. Ship navigation and fuel profiling based on noon report using neural network generative modeling. J. Phys. Conf. Ser. 2311, 12005. https://doi.org/10.1088/1742-6596/2311/1/012005.
Hu, Z., Jin, Y., Hu, Q., Sen, S., Zhou, T., Osman, M.T., 2019. Prediction of fuel consumption for enroute ship based on machine learning. IEEE Access 7, 119497–119505. https://doi.org/10.1109/ACCESS.2019.2933630.
International Maritime Organisation (IMO), 2011. Energy efficiency measures. IMO. https://www.imo.org/en/OurWork/Environment/Pages/Technical-and-Operational-Measures.aspx. (Accessed 1 September 2022).
Kee, K.K., Simon, B.Y., Renco, K.Y., 2018. Artificial neural network back-propagation based decision support system for ship fuel consumption prediction. In: Proceedings of the 5th IET International Conference on Clean Energy and Technology (CEAT2018). IEEE, pp. 1–6. https://doi.org/10.1049/cp.2018.1306.
Kim, Y.R., Jung, M., Park, J.B., 2021. Development of a fuel consumption prediction model based on machine learning using ship in-service data. J. Mar. Sci. Eng. 9, 1–25. https://doi.org/10.3390/jmse9020137.
Leifsson, L.T., Sævarsdóttir, H., Sigurdsson, S.T., Vésteinsson, A., 2008. Grey-box modeling of an ocean vessel for operational optimization. Simul. Model. Pract. Theory 16, 923–932. https://doi.org/10.1016/j.simpat.2008.03.006.
Liu, D., Lee, S., Huang, Y., Chiu, C., 2020. Air pollution forecasting based on attention-based LSTM neural network and ensemble learning. Expert Syst. 37, e12511.
Leong, S.C., Singhal, P., 2015. Estimation of CO2 emission from marine traffic in Singapore Straits using Automatic Identification Systems data. In *Environmental Science and Information Application Technology*. CRC Press, pp. 177–184.
Lianne and Justin, 2020. 3 Steps to forecast time series: LSTM with TensorFlow Keras | towards data science. Just into Data. https://towardsdatascience.com/3-steps-to-forecast-time-series-lstm-with-tensorflow-keras-ba88c6f05237 . (Accessed 1 September 2022).
Liu, F., Cai, M., Wang, L., Lu, Y., 2019. An ensemble model based on adaptive noise reducer and over-fitting prevention LSTM for multivariate time series forecasting. IEEE Access 7, 26102–26115.
Meng, Q., Du, Y., Wang, Y., 2016. Shipping log data based container ship fuel efficiency modeling. Transp. Res. Part B Methodol. 83, 207–229.
MPA, 2017. Mass Flow Meter for Bunkering. Maritime Port Authority. https://www.mpa.gov.sg/web/portal/home/port-of-singapore/services/bunkering/mass-flow-meter-for-bunkering. (Accessed 1 September 2022).
Norwegian Government Action Plan, 2019. The Government's Action Plan for Green Shipping. Norwegian Ministry of Climate and Environment. https://www.chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.regjeringen.no/contentassets/2ccd2f4e14d44bc88c93ac4effe78b2f/the-governments-action-plan-for-green-shipping.pdf.
Petersen, J.P., Winther, O., Jacobsen, D.J., 2012. A machine-learning approach to predict main energy consumption under realistic operational conditions. Sh. Technol. Res. 59, 64–72.
Rabiner, L.R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE 77, 257–286.
Rousseeuw, P.J., 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. 20, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7.
Syakur, M.A., Khotimah, B.K., Rochman, E.M.S., Satoto, B.D., 2018. Integration *K*-means clustering method and elbow method for identification of the best customer profile cluster. IOP Conf. Ser. Mater. Sci. Eng. 336, 12017. https://doi.org/10.1088/1757-899X/336/1/012017.
Tay, Z.Y., Hadi, J., Chow, F., Loh, D.J., Konovessis, D., 2021. Big data analytics and machine learning of harbour craft vessels to achieve fuel efficiency: a review. J. Mar. Sci. Eng. 9 (12), 1351. https://doi.org/10.3390/jmse9121351.
Tay, Z.Y., Hadi, J;, Konovessis, D., Loh, D.J., Tan, D.K.H, Chen, X., 2021. Efficient harbour craft monitoring system: time-series data analytics and machine learning tools to achieve fuel efficiency by operational scoring system. In: Proceedings of the ASME 2021 40th International Conference on Ocean, Offhsore and Arctic Engineering OMAE. ASME, OMA62658, 2021. Virtual Online.
Tran, T.A., 2021. Comparative analysis on the fuel consumption prediction model for bulk carriers from ship launching to current states based on sea trial data and machine learning technique. J. Ocean Eng. Sci. 6, 317–339. https://doi.org/10.1016/j.joes.2021.02.005.
US EPA, 2018. Greenhouse Gas Emissions from a Typical Passenger Vehicle. US EPA. Ann Arbor.
Uyanık, T., Karatuğ, Ç., Arslanoğlu, Y., 2020. Machine learning approach to ship fuel consumption: a case of container vessel. Transp. Res. Part D Transp. Environ. 84, 102389.
Wang, S., Ji, B., Zhao, J., Liu, W., Xu, T., 2018. Predicting ship fuel consumption based on LASSO regression. Transp. Res. Part D Transp. Environ. 65, 817–824.
Wen, S., Zhang, C., Lan, H., Xu, Y., Tang, Y., Huang, Y., 2019. A hybrid ensemble model for interval prediction of solar power output in ship onboard power systems. IEEE Trans. Sustain. Energy 12, 14–24.
Zhu, Y., Zuo, Y., Li, T., 2020. Predicting ship fuel consumption based on LSTM neural network. In: Proceedings of the 7th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS). Guangzhou, China, pp. 310–313. https://doi.org/10.1109/ICCSS52145.2020.9336914, 13- 15 November 2020.