Anomaly Detection of Marine Diesel Engines: A Novel Approach using Transformer Neural Networks for Reconstruction and Residual Analysis

Qin Liang[1], Erik Vanem[2], Knut Erik Knutsen[3], Vilmar Æsøy [4] and Houxiang Zhang [5]

[1,2,3] *DNV Group Research & Development, Høvik, Viken, 1351, Norway*
Qin.Liang@dnv.com
Erik.Vanem@dnv.com
Knut.Erik.Knutsen@dnv.com

[1,4,5] *Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology, Ålesund, 6009, Norway*
qinlia@stud.ntnu.no
vilmar.aesoy@ntnu.no
hozh@ntnu.no

ABSTRACT

This paper proposes an unsupervised approach for anomaly detection in marine diesel engines using a transformer neural network based AutoEncoder (TAE) and residual analysis with Sequential Probability Ratio Test (SPRT) and Sum of Squares of Normalized Residuals (SSNR). This approach effectively captures temporal dependencies within normal time-series data, eliminating the need for labeled failure data. To assess the performance of the proposed methodology, faulty data is collected under the same operational profile as normal training data. The TAE is trained on the normal data, after which the faulty data is tested using the trained model. Subsequently, the SPRT and SSNR methods are used to analyze residuals from the observed (input) and reconstructed (output or tested) faulty data. Deviations exceeding a predefined threshold are identified as anomalous behavior. Furthermore, this study explores various architectures of transformer neural networks and other types of neural networks to conduct a comprehensive comparative analysis of the performance of the proposed approach. Insights and recommendations derived from the performance analysis are also presented, which offers valuable information for potential users to leverage. The test results demonstrate the ability of the proposed approach to accurately and efficiently detect anomalies in marine diesel engines. Specifically, it can detect anomalies more than 1000 time steps ahead of system alarms, outperforming other tested models.

1. INTRODUCTION

The maritime industry plays a critical role in global trade and transportation, over 80% of the volume of international trade in goods is carried by sea (Stalk, 2021). Ships and their onboard equipment form the backbone of the operation of the maritime industry. Maintaining onboard ship equipment is critical to ensuring safe and efficient vessel operations. However, maintaining marine equipment poses significant challenges due to the remote and harsh environment of the sea. To ensure that equipment performs optimally, ship owners and operators need to have an effective maintenance strategy in place that balances safety, cost, and operational efficiency. In addition to the challenges of maintaining equipment, the development of autonomous ships adds another layer of complexity. Autonomous ships require higher equipment safety and reliability standards, making the need for an effective maintenance strategy even more critical.

In this context, it is essential to continue to explore innovative and effective maintenance approaches to ensure the safe and reliable operation of marine equipment. The development of advanced data analytics, Internet of Things (IoT) technologies, and machine learning algorithms holds great promise to improve maintenance of onboard equipment and achieve optimal performance (Knutsen et al., 2022). Over the recent years, much has been implemented on these topics through two main approaches: data-driven approaches and model-based approaches (Bernardo & Reichard, 2017). Especially data-driven approaches applying Deep Learning (DL) tech-

https://doi.org/10.36001/IJPHM.2024.v15i3.3853

niques have become a popular direction with successful implementation in different domains. Neural networks are a type of machine learning algorithms that are modeled after the structure and function of the human brain. Deep learning is a subset of machine learning, and neural networks make up the backbone of deep learning algorithms (Kriegeskorte & Golan, 2019). In fact, it is the number of node layers, or depth, of neural networks that distinguishes a single neural network from a deep learning algorithm, which normally have more than three.

Neural networks and deep learning models can be categorized based on their architecture and working mechanisms. There are various types of neural networks, such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and AutoEncoder (AE). On the other hand, new architectures are actively being developed by researchers. Over the last couple of years, there has been a surge in the development of large generative language models like ChatGPT. The key to their success lies in the transformer architecture, which serves as the foundational pillar for these models.

This paper proposes and tests a new method for detecting anomalies in marine diesel engines using a transformer neural network. The approach is unsupervised, meaning that the model is trained on normal operational data and tested on data from faulty operations. Once reconstruction is complete, SPRT and SSNR are used to evaluate the model's performance.

The remainder of the paper is organized as follows: Section II discusses the latest trends in data-driven equipment anomaly detection. Section III presents the methodology used in this study. Section IV covers the model and data used in the study, including the collection process and model training. Results and analysis are presented in Section V. Finally, Section VI concludes the paper and proposes future work.

2. RELATED STUDIES

Data-driven anomaly detection has garnered significant research attention in recent years. (Liang, Knutsen, Vanem, Æsøy, & Zhang, 2024) provides a comprehensive review of anomaly detection in maritime equipment, highlighting the current research emphasis on data-driven methodologies. (Vanem & Brandsæter, 2021) presents a comprehensive implementation of cluster-based anomaly detection for marine engine systems. This study highlights the potential of employing statistical techniques for effective anomaly detection. Concurrently, numerous researchers are also exploring the application of DL methods in this domain, i.e., (Han, Li, Skulstad, Skjong, & Zhang, 2020), (Ellefsen, Bjørlykhaug, Æsøy, Ushakov, & Zhang, 2019) and (Hu, Cheng, Wu, Zhu, & Shao, 2021). An autoencoder (AE) is a type of artificial neural network that is used for unsupervised learning of efficient data representations. It consists of an encoder that compresses the input into a latent representation and a decoder that decompresses it back to a reconstructed output. By minimizing the reconstruction error between the input and output, the AE learns to capture the essential features in the data. After training, the encoder can be used to encode new data, while the decoder can reconstruct an approximation of the original input from the encoded representation. Because of the mechanism, AEs can be used for anomaly detection, where they are trained on healthy data and then used to detect any deviations from the normal behavior of the machinery or system. The key advantage is that AEs adopt an unsupervised learning architecture, they do not require large amounts of data to be labeled. (Han, Ellefsen, Li, Holmeset, & Zhang, 2021) proposed an LSTM-based Variational AutoEncoder (LSTM-VAE) for fault detection in maritime components. (Listou Ellefsen et al., 2020) proposed a fault-type independent spectral anomaly detection algorithm for marine diesel engine degradation based on Variational AutoEncoder (VAE). (Hemmer, Klausen, Khang, Robbersmyr, & Waag, 2020) introduced an unsupervised learning approach for detecting defects in large, slow-rotating axial bearings by developing a Health Indicator (HI). The proposed method utilizes variational inference and involves the use of a VAE and a Conditional Variational AutoEncoder (CVAE).

RNN is a type of neural network that is designed to handle sequential data. Unlike feed-forward neural networks like Multilayer perceptron (MLP), which process input data in a fixed order and don't have any memory (Liang, Tvete, & Brinks, 2019) and (Liang, Tvete, & Brinks, 2020), RNN maintain an internal state that allows them to process sequences of varying lengths and capture the temporal dependencies between successive inputs. (Hu et al., 2021) introduced a new Deep Bidirectional Recurrent Neural Networks (DBRNN) ensemble method for Remaining Useful Life (RUL) prediction of aircraft engines. CNN has shown promising results in detecting faults based on acoustic signals, vibration data, and thermal images. For example, (Massoudi, Verma, & Jain, 2021) used CNN to classify engine sounds based on the type and severity of the fault.

After conducting a literature survey, it appears that the use of Transformer Neural Network for anomaly detection is not widely explored. To address this gap, (Zhang, Song, & Li, 2022) proposed a new deep method for RUL prediction called Dual-Aspect Self-attention based on transformer (DAST) to improve the overall efficiency of predictive maintenance tasks. The results demonstrated that DAST outperforms DBRNN and CNN methods in terms of Root Mean Squared Error (RMSE) and score values for most engines. It is important to note that DAST is a supervised learning approach that requires labeled data for training. However, in reality, obtaining sufficient fault or RUL data can be challenging, which may limit the performance of the model. In another transformer related study, (Tuli, Casale, & Jennings, 2022) introduced TranAD, a

deep transformer network for efficient and accurate anomaly detection and diagnosis in multivariate timeseries data. TranAD outperforms state-of-the-art baseline methods in both detection and diagnosis performance while offering data and time-efficient training. The paper uses seven publicly available datasets in their experiments. The authors acknowledge some concerns about the lack of quality benchmark datasets for time series anomaly detection.

3. METHODOLOGY

3.1. Proposed TAE

Transformer neural networks are deep learning models introduced by (Vaswani et al., 2017), and have gained great popularity in the field of natural language processing. Unlike RNNs, Transformer Neural Networks (TNN) have a parallelizable architecture, making them faster for certain tasks. They also require fewer training iterations and are less prone to the vanishing gradient problem than RNNs. Although RNNs have been widely used for sequence modeling, their limitations have led to the development of TNN. TNN have shown superior performance in several natural language processing tasks, including language translation, compared to RNNs.
RNNs use recurrent connections to process sequential data, while TNNs rely on self-attention mechanisms to capture dependencies between all elements in a sequence, without using any connections between the elements themselves. One key challenge in applying self-attention to sequential data is that the order of the elements in the sequence is lost when computing the attention weights. This is because the attention mechanism computes the attention weights based on the similarity between the query vector and the keys associated with each element in the sequence, regardless of their position. This makes it challenging for the model to differentiate between elements at different positions in the sequence. To address this issue, the TNN introduces positional encoding. Positional encoding is a technique that adds a fixed positional vector to the input embeddings, providing the model with information about the position of each element in the sequence. The purpose of this is to provide the model with positional information, allowing it to distinguish between different elements in the sequence.

Instead of using TNN for prediction, in this study TNN was used in an auto-encoder manner to reconstruct the data. This study introduces two transformer auto-encoder architectures, namely TAE and TNN-MLP. The architectural depiction of the proposed framework is presented in Figure 1. Notably, the difference between these two architectures lies in the design of the decoder. Specifically, TAE incorporates transformer layers in both encoder and decoder, whereas TNN-MLP employs MLP structures within the decoder layer. This variation in architectural design aims to facilitate a comprehensive evaluation of transformer architecture. The details of the proposed architecture are illustrated as follows.

1. Feed-forward Neural Network: An FNN is a fundamental type of artificial neural network that can be used as a building block for constructing more complex models such as MLPs. It is characterized by its fully connected structure, where each unit in one layer is directly connected to all units in the subsequent layer via weight connections.
2. Positional encoding is a way of incorporating position information into the input embeddings by adding a fixed vector to each embedding, which varies based on its position in the sequence. There are various ways of positional encoding methods to choose. In this paper, the method from (Vaswani et al., 2017) is used.
3. Residual connection and normalization layer (Add & Norm): This layer is added after each sublayer in TNN encoder. The function of residual connections is to ease the challenge of training deep neural networks. Meanwhile, layer normalization can quicken the training progress and promote faster convergence of the model by normalizing the activation value of each layer.
4. Multihead self-attention layer: The encoder employs multihead self-attention to extract the significance of various sensors along the sensor dimension, enabling it to autonomously learn to prioritize characteristics with higher weights. As a consequence, there is no need for human intervention during the training process, resulting in an automated and efficient feature selection process.
5. Layer normalization and residual connections: Each sublayer in the TNN, including multi-head self-attention and position-wise feed-forward networks, is surrounded by residual connections and followed by layer normalization. Residual connections allow the output of each sublayer to be added to its input, which helps to mitigate the vanishing gradient problem by allowing gradients to flow directly to earlier layers. Layer normalization, on the other hand, is a technique that is used to normalize the activations of each layer. This helps to stabilize the learning process by reducing the internal covariate shift.

3.2. Sequential Probability Ratio Test

SPRT is a statistical method (Wald, 1992) for testing a hypothesis based on a sequential analysis of data. The method involves taking samples of data sequentially and updating the probability of a hypothesis after each sample is taken. (Vanem & Storvik, 2017) , (Brandsæter, Vanem, & Glad, 2019) and (Brandsæter, Manno, Vanem, & Glad, 2016) have already explored the application of SPRT on maritime equipment and proved its capability of detecting anomalies.

The trained model introduced previously provides a reconstruction $x_t^{'}$ of the observed signal values x_t at each time step

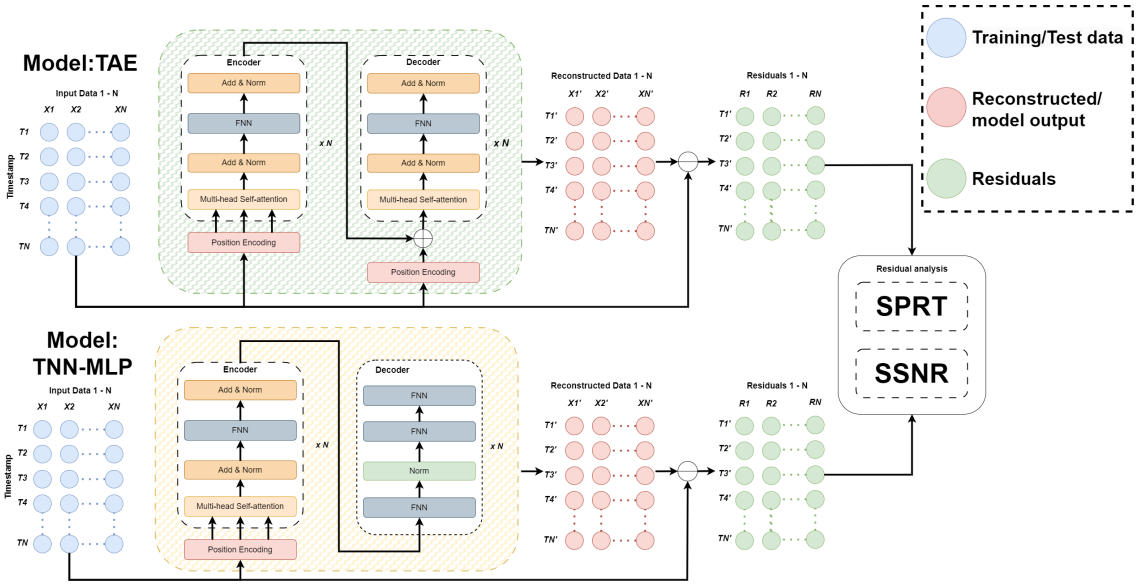

Figure 1. Proposed architecture

t. The residuals, i.e. the difference between the reconstructed and the observed value $r_t = x_t^{'} - x_t$ are analyzed sequentially by the SPRT to determine if the signal indicates a normal or anomalous state of the system. To employ SPRT for analyzing residual data, it is necessary to define two competing hypotheses: a null hypothesis H_0 and an alternative hypothesis H_1. The null hypothesis H_0 asserts that residuals are normally distributed with mean 0 and a standard deviation σ, representing the system in its normal state. This normal state is a condition where the system operates as expected, and these residual properties serve as a reference for anomaly detection. Comparing observed residuals to this reference allows the SPRT to identify deviations indicating anomalies, ensuring system performance and reliability. In contrast, the alternative hypothesis H_1 assumes that the residuals are normally distributed with a specific mean μ and/or a different standard deviation σ', indicating an anomalous state. The SPRT is performed independently for each feature to detect these potential anomalies.

$$\begin{aligned} H_0 &: r \sim N(0, \sigma) \\ H_1 &: r \sim N(\mu, \sigma') \end{aligned} \tag{1}$$

The SPRT can be calculated in the following steps. It is assumed both follow normal distribution, the normal distribution probability function is:

$$f(r) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{(r-\mu)^2}{2\sigma^2}\right) \tag{2}$$

where r is the residuals of reconstructed and observed signal value. Then, the likelihood of the data for the hypotheses L_0 and L_1 can be calculated using the following functions:

$$\begin{aligned} L_0(x) &= \prod_{i=1}^{n} f_1(r_i) \\ &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{(r_i)^2}{2\sigma^2}\right) \\ L_1(x) &= \prod_{i=1}^{n} f_0(r_i) \\ &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma'^2}} \cdot \exp\left(-\frac{(r_i-\mu)^2}{2\sigma'^2}\right) \end{aligned} \tag{3}$$

After that, the likelihood ratio can be calculated based on L_1 and L_0 when H_0 has a mean of 0. If L_1 is greater than L_0, it indicates that the distribution aligns more closely with H_1 than with H_0, and vice versa. The log likelihood ratio can be calculated as:

$$log\frac{L_1}{L_0} = log(\prod_{i=1}^{n} \frac{\sigma}{\sigma'} \exp\left[\frac{(r_i)^2}{2\sigma^2} - \frac{(r_i-\mu)^2}{2\sigma'^2}\right]) \tag{4}$$

In this case, r_i represents the residuals at each time step i. Once the hypotheses are defined and the log likelihood is calculated, the SPRT index can be sequentially calculated and updated. To achieve this, two threshold values, A and B, must be specified. The calculated SPRT index at each time step is then compared with these lower and upper decision

boundaries. At each time step, three possible outcomes can occur:

- If the value falls below the lower limit (A), it indicates the acceptance of the normal state (H_0). Consequently, the test statistic is reset.
- If the value exceeds the upper limit (B), it suggests the acceptance of the anomalous state (H_1). Accordingly, the test statistic is reset.
- When the value lies between the defined threshold values, it signifies an insufficiency of available information to reach a conclusive decision.

$$SPRT = \begin{cases} \text{if } \log \frac{L_1}{L_0} > B, \ 0 \\ \text{if } A \leq \log \frac{L_1}{L_0} \leq B, \ \log \frac{L_1}{L_0} \\ \text{if } \log \frac{L_1}{L_0} < A, \ 0 \end{cases} \tag{5}$$

The thresholds A and B can be calculated based on the following equations:

$$\begin{aligned} A &= \log\left(\frac{\beta}{1-\alpha}\right) \\ B &= \log\left(\frac{1-\beta}{\alpha}\right) \end{aligned} \tag{6}$$

where α is the probability of Type I error (false alarm), which represents the probability of rejecting the true H_0 (i.e., falsely identifying normal states as anomalous). β is the probability of Type II error (i.e., missing anomalous states), which represents the probability not rejecting H_0 when it is false. Rejecting H_0 inherently implies the acceptance of the alternative hypothesis H_1. In this study, both α and β are set as 0.01. Figure 2 illustrates how SPRT works for each feature to be tested.

3.3. Sum of Squares of Normalized Residuals

The chi-square distribution, also written as χ^2 distribution, is a continuous probability distribution widely used in statistical inference and hypothesis testing. It is particularly relevant in scenarios where the sum of squared independent, identically distributed random variables is being analyzed. The chi-square distribution is a special case of the gamma distribution and is often used in goodness-of-fit tests, independence tests for contingency tables, and the estimation of confidence intervals.

The chi-square distribution is characterized by its degrees of freedom, which determine the shape of the distribution. The degrees of freedom are typically related to the number of independent observations or constraints in a given problem. Specifically, the sum of the squares of k independent standard normal distribution variables follows a chi-square distribution with k degrees of freedom. This concept forms the basis of the SSNR. In this study, the assumption is made that the SSNR follows a chi-square distribution with k degrees of freedom equal to the number of features, which is 21.

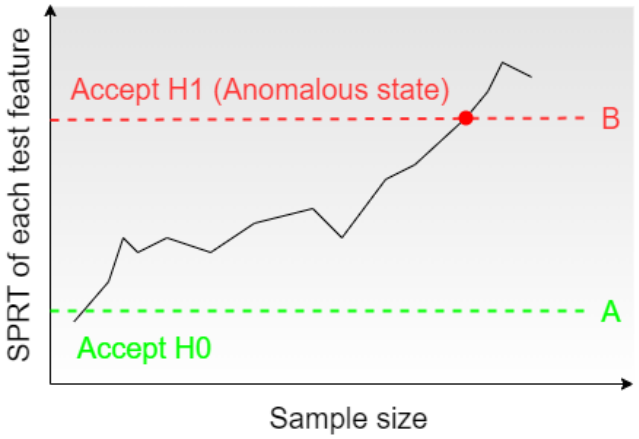

Figure 2. SPRT method illustration

To assess the significance of the SSNR, a comparison is made with the corresponding chi-squared distribution. This enables the determination of the probability of observing an SSNR value as large or larger than a defined threshold. The threshold for hypothesis testing can be derived using the inverse cumulative distribution function (CDF), which allows the mapping of probabilities back to values from the distribution. Three confidence levels are selected in this study: 99.99%, 99.7%, and 95% for evaluation. The confidence level represents the threshold probability. For instance, a confidence level of 99.7% implies a 0.3% chance of making a false alarm. By considering the given confidence level and degrees of freedom (equal to the number of features), threshold values of 47.56, 37.37, and 27.58 are obtained using the inverse CDF. The selection of a 99.7% confidence level is guided by the three-sigma rule (Pukelsheim, 1994), which serves as a widely recognized benchmark. However, in practical applications, the confidence level can be adjusted to meet specific requirements. Different applications may exhibit varying degrees of sensitivity to inaccuracies in reconstructed signals. A higher confidence level of 99.99% is also selected based on the condition of this study.

By applying the threshold to the SSNR, the reconstruction error can be effectively monitored. The SSNR is defined as follows:

$$SSNR = \sum_{i=1}^{d_i} \left(\frac{r_i - \mu_o}{\sigma_o}\right)^2 \tag{7}$$

where d_i is the number of features, r_i is the residuals of re-

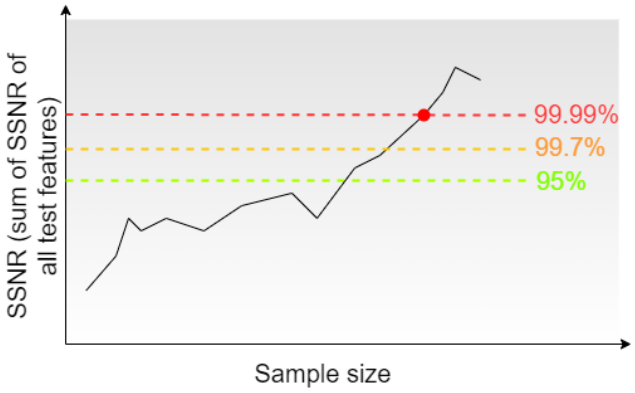

Figure 3. SSNR method illustration

constructed faulty data, μ_o is the mean of residuals of reconstructed normal data, σ_o is the standard deviation of residuals of reconstructed normal. Figure 3 provides an example of the application of SSNR. In this figure, different dashed lines represent distinct confidence levels, while the Y-axis depicts the cumulative SSNR of all tested features. Notably, for each sample, only a single SSNR value is obtained. Further details regarding the test results will be discussed in the following sections.

4. EXPERIMENTAL STUDY

4.1. Data collection and processing

The Department of Ocean Operations and Civil Engineering at NTNU Ålesund has established a hybrid power lab for data collection. The laboratory consists of a compact marine diesel engine integrated with a generator, a marine battery system, a marine DC switchboard equipped with essential power converters, and a comprehensive marine automation system that supervises the entire operational process. Notably, the generated power is seamlessly redirected back into the power grid to effectively simulate dynamic load fluctuations within the system. The test bench is illustrated in Figure 4, which shows the marine diesel engine, the installed sensors, and the power switchboard. The system's energy flow is shown in Figure 5.

Data is collected by running the engine on an operational profile simulating a ferry crossing on Norway's west coast. The ferry departs from shore at a safe and constant speed, then accelerates until it reaches a suitable speed. The speed is maintained at a constant level before safely decreasing and finally braking just before docking. The entire ferry crossing process takes 20 minutes, and the complete engine operating profile is depicted in Figure 6. Both the normal operation data and the faulty degradation data are collected while running the same engine operating profile. The only difference between the two data sets is that a fault is introduced in the faulty degradation data. Therefore, the primary objective is to predict the fault time step on time.

The engine is equipped with two water cooling systems - a primary and a secondary system, where the latter cools the former. The primary cooling system is regulated by an internal bimetal thermostatic valve, which commences opening at a temperature of 78°C and reaches full opening at 90°C. On the other hand, the secondary cooling system relies on a frequency operated fan that circulates air through a heat exchanger. A malfunction of the fan is intentionally introduced to create a fault that subsequently leads to a decline in cooling efficiency within the secondary cooling system to generate the test data. To prevent potential issues, the system triggers an alarm when the cooling water temperature exceeds 85°C. A total of 2336 time steps were recorded over a 1168-second period, with a frequency of 2 Hz. In this study, the feature selection process was implemented by employing the domain knowledge and expertise of the engine operator to select 21 input features. All the features have been used in the training of proposed AE models.

The data used for training has undergone zero-mean and unit variance normalization. This technique involves scaling the features to have a zero mean and unit variance, which ensures that all features are on a comparable scale and prevents the dominance of features with large variances during the learning process. Moreover, such normalization can improve the stability and performance of machine learning models during training. It is worth noting that the normalization statistics derived from the normal operation data are also utilized for the faulty degradation data.

The reconstruction of the data is a time series problem. When dealing with time series data, it is often useful to divide it into smaller sub-sequences or sequences that have a fixed length. This can be done using a sliding window approach where a window of fixed length is moved across the time series data at a fixed stride. At each window position, the sub-sequence of data within the window is extracted and added to a list of sub-sequences.

4.2. Model training

In the proposed architecture for time series reconstruction using the transformer, several key hyper-parameters must be defined to ensure its effective implementation. The number of TNN layers refers to the number of encoding and decoding layers in the architecture. Increasing the number of layers can improve the model's ability to capture complex temporal patterns but may also increase the risk of overfitting. The number of heads is the number of parallel attention mechanisms that are applied in each encoder and decoder layer. Increasing the number of heads can enhance the model's ability to attend to multiple parts of the input sequence simultaneously. The time sequence length is the length of the input

Figure 4. Test bench: Marine diesel engine, sensors and power switchboard

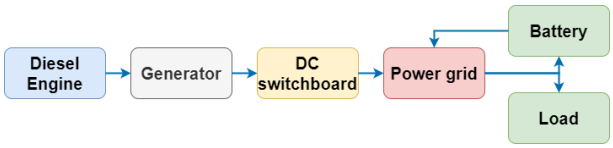

Figure 5. System energy flow

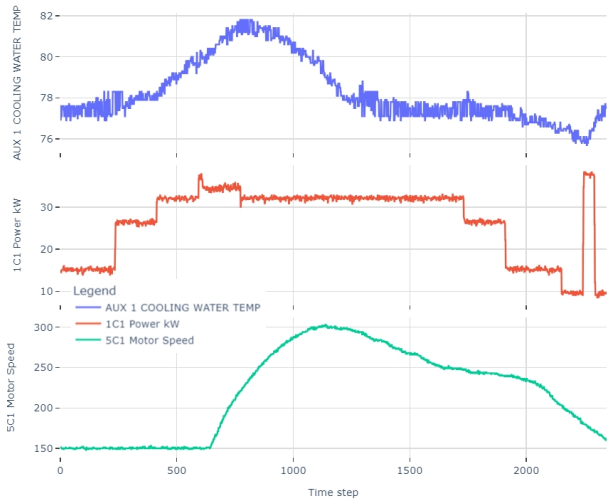

Figure 6. Engine operation profile

time series sequence that is fed into the model. This parameter determines how much historical data the model can use to make its predictions. The feedforward network's dimension refers to the hidden layer size in the FNN component of the TNN. Increasing this dimension enables the model to capture more complex feature relationships.

This study trains four model series, each contributing a unique architecture to the reconstruction task:

1. **TAE (Transformer-based AutoEncoder)**: A transformer-based autoencoder, where transformer layers are employed both in the encoder and decoder modules. The integration of transformer structures in both components enhances the model's ability to capture intricate relationships within the input data.
2. **TNN-MLP (Transformer with MLP Decoder)**: This alternative architectural proposal distinguishes itself by incorporating a transformer layer in the encoder and a Multilayer Perceptron (MLP) in the decoder. This hybrid design seeks to leverage the strengths of transformer mechanisms for encoding, while employing the flexibility of MLP structures for decoding, thereby offering a versatile approach to information representation.
3. **MLP (Multilayer Perceptron)**: A standalone multilayer perceptron neural network model, this architecture relies on a series of interconnected layers.
4. **LSTMAE (LSTM Autoencoder)**: This configuration

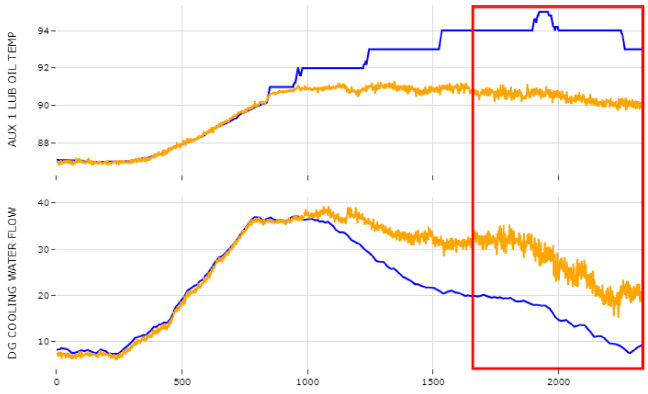

Figure 7. Observed (Blue) vs Reconstructed (Orange) data

adopts an autoencoder architecture based on LSTM layers in the encoder.

In this study, TAE, TNN-MLP, and LSTMAE models require time-series data as input. The training data will be processed with a defined sequence length, and a sliding window will be applied across the entire dataset for these models. On the other hand, the MLP can accept input data directly. For TNN-MLP, LSTMAE, and MLP, only the best-performing models are chosen for evaluation. In the context of TAE, a variety of TAE models with diverse architectures and parameters have been trained to identify the optimal model. The initial architecture is proposed based on the common practices (Hongchun, Fanlun, & Xiaoyong, 2001) and test results. Based on that, around 80 TAE models with different parameters have been trained, only the best of each training group is selected for evaluation. Five representative TAEs have been selected for more detailed examination. In addition, the parameters tested on TAE can be observed in Table 2. In total, as presented in Table 1, nine models have been chosen for evaluation, comprising six TAE models, one TNN-MLP, one MLP, and one LSTMAE. All selected time series models are trained using a sequence length of 60 time steps.

5. RESULT ANALYSIS

As previously mentioned, the system triggers an alarm if the cooling water temperature exceeds 85°C. In this particular case, a cooling fault occurred due to a malfunction in the fan at the beginning of the test, resulting in a reduction in cooling efficiency in the secondary cooling system. The evaluation of the trained model is carried out by analyzing both the observed and reconstructed data. Figure 7 shows observed (blue) and reconstructed (orange) data features from the TNN-MLP model. The red square marks the time step (1658) when the system alarm was triggered, persisting until the test ended.

5.1. Evaluation of reconstruction performance

In this study, all models utilized for analysis are autoencoders. The ability of autoencoders to reconstruct the original data is an important metric of their performance. The Root Mean Square Error (RMSE) is calculated for each of the 21 features across all time steps, and then a mean RMSE across all features is calculated. The results are shown in Figure 8, indicating that the reconstruction performance was good, with only the LSTMAE model exhibiting a slightly worse performance compared to the other models.

5.2. Evaluation on SPRT

As introduced previously, the SPRT is computed using the log likelihood and two thresholds. The SPRT index is reset whenever it surpasses both thresholds. In this study, the normal data set is referred to as the training data, while the faulty data are referred to as the test data. Standardization of residuals from the reconstructed test data is crucial for consistent SPRT performance. The residuals of the reconstructed test data are standardized using the mean and standard deviation derived from the first 500 time steps of the same data. It is important to note that during these initial 500 time steps, the system is in normal operation mode, with all signals falling within their respective normal working ranges. This standardization approach ensures a stable performance of the SPRT.

In this study, two alternative hypotheses are examined: deviations in the positive and negative directions of the mean. The three-sigma rule is applied in this step, with the alternative means of 4 and -4 being utilized in the tests. The result of the positive test of model TAE-1 is presented in Figure 9 as an example. In Figure 9, each subplot depicts the application of SPRT to a specific feature. The red dashed line represents the upper limit (B), and the red dot indicates the first time index at which an anomaly is detected.

For the anomalies detected in the first 500 time steps are taken as faulty warning. It is worth noting that the SPRT indices are calculated for selected number of features, with some features indicating errors earlier than others. The detected anomalous time step for each feature is defined as the first instance when it crosses the threshold after 500 time steps. Subsequently, the overall detected anomaly time step is determined by computing the average of all detected anomaly time steps across all features. The detailed average anomalous time steps identified by positive mean test of all the models is presented in Figure 10. The different dots for the same model means the indication of different number of flagged features (from 5 to 21). Selecting a higher number of flagged features provides greater confidence in the identified anomalies; however, it will result in delayed notification. 10 features is tested as the balancing point with good performance. The SPRT result with mean of first 10 flagged features can be seen in Figure 11.

Table 1. 9 selected models

	Model	# encoder layers	# decoder layers	# Hidden dimensions	# Trainable parameters
1	TAE-1	6	1	2048	632009
2	TAE-2	6	1	1024	323785
3	TAE-3	3	1	2048	361958
4	TAE-4	3	3	2048	545772
5	TAE-5	6	3	2048	815823
6	TAE-6	10	1	2048	992077
7	TNN-MLP	5	5	2048	577685
8	MLP	4	4	160-640	212501
9	LSTMAE	2	2	512	817429

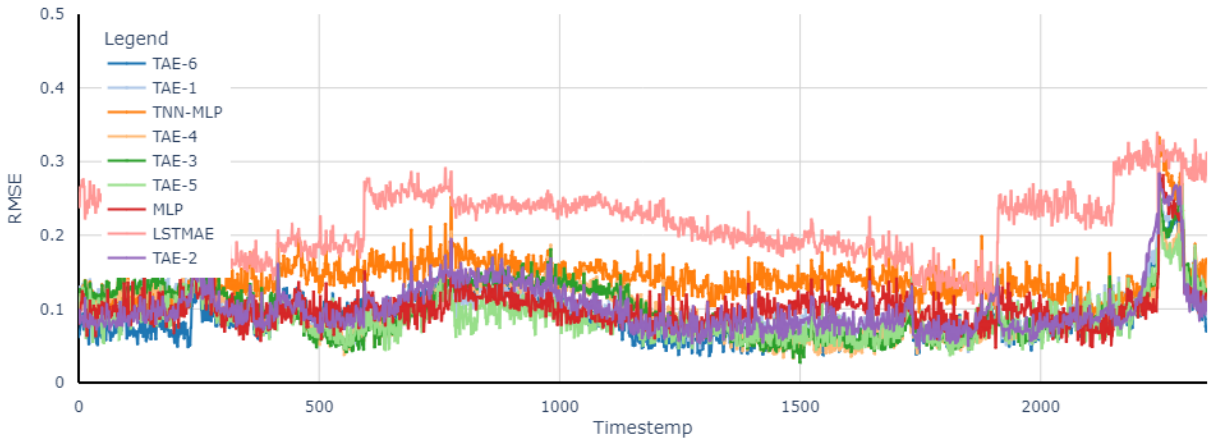

Figure 8. Reconstration RMSE of training data

Table 2. Transformer autoencoders

Parameters	Values
Sequence length	10,30,60
Transformer encoder layers	3,6,10
Transformer decoder layers	1,3
Hidden dimensions	512,1024,2048
Number of head	7
Learning rate	0.001, 0.005, 0.01
Optimizer	SGD, ADAM

As shown in both Figure 10 and Figure 11, all the transformer related models show better performance than MLP, LSTMAE. With transformer architecture, the ones with transformer in both encoder and decoder shows better performance than the half transformer architecture (TNN-MLP).

The TAE-4 is the first model capable of indicating anomalies. However, it requires more time to establish confidence in its detection, specifically to reach the required number of features with anomalous flags. Conversely, TAE-1 and TAE-2 exhibit comparable and commendable performances in early anomaly detection and confidence building. The difference between TAE-1 and TAE-2 resides in the dimensionality of the feedforward network model within each transformer layer, with TAE-2 employing half the hidden dimension of TAE-1. Despite this, TAE-2 can detect anomalies slightly earlier than TAE-1, suggesting adequacy in the number of hidden neurons in TAE-2.

TAE-6 adopts a 10-1 encoder/decoder architecture with the deepest structure and the highest number of trainable parameters, consequently demanding the longest training time. However, TAE-6 fails to exhibit better performance compared to other TAE models, indicating that the deeper encoder does not significantly contribute to performance. Models with six layers in the encoder (TAE-1, TAE-2, TAE-5) are deemed appropriate. Furthermore, the decoder does not require more than one layer, as evidenced by the superior performance of TAE-1 and TAE-2 over TAE-5, and the increase in decoder complexity results in additional training time.

The lower ranking of TNN-MLP compared to the TAE models suggests that the compressed representations from its encoder may indeed require a more sophisticated decoder to achieve effective reconstruction. Despite having similar trainable parameters as TAE-4, TNN-MLP requires significantly less training time.

Both MLP and LSTMAE can detect anomalies before system alarms are triggered, but not as effectively as transformer-related models. Even when employing a half-transformer ar-

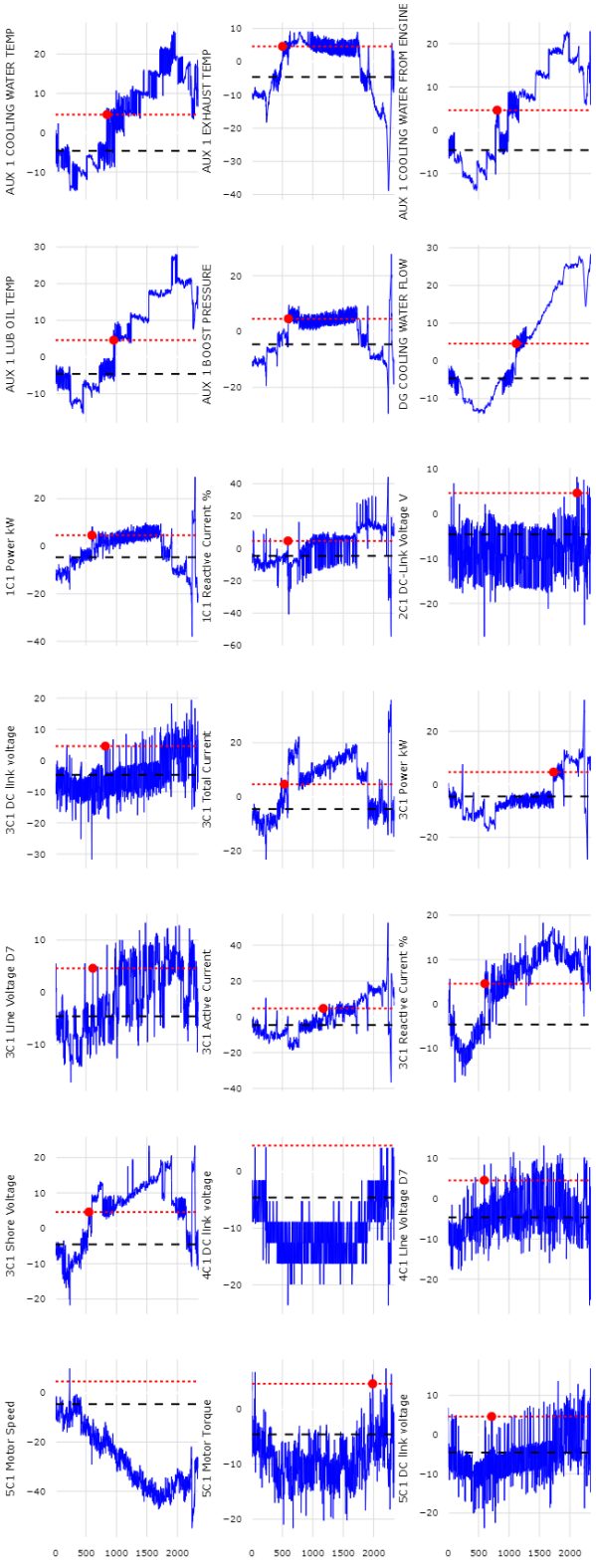

Figure 9. TAE-1 SPRT on test data - positive mean change

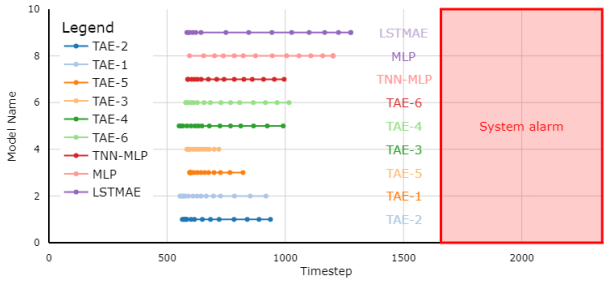

Figure 10. SPRT on test data with different number of selected features with Flag

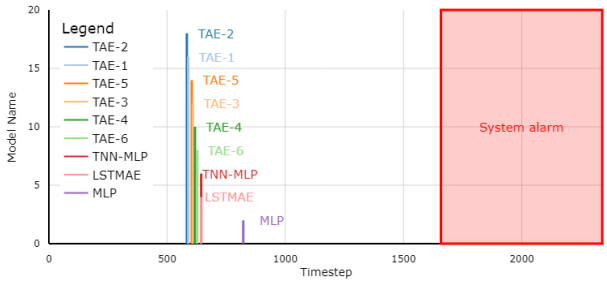

Figure 11. SPRT on test data of the first 10 features with Flag

chitecture like TNN-MLP, it demonstrates better performance compared to MLP and LSTMAE. It is noteworthy that across all test categories, longer sequence lengths (time steps) consistently yield superior performance compared to shorter ones. This observation rationalizes the choice to train all selected models using a sequence length of 60 time steps.

5.3. Evaluation on Sum of Squares of Normalized Residuals

In addition to assessing the models' performance using SPRT, the performance is also evaluated using SSNR, as previously introduced. To implement SSNR, the test data residuals undergo standardization, adopting the same approach used in SPRT. The mean and standard deviation of the test data residuals from the first 500 time steps are utilized for standardization. A time step is considered a potential warning if its SSNR value exceeds the average. The SSNR performance is illustrated in Figure 12, where it is evident that the SSNR can detect anomalies well before the system alarm is activated. Figure 13 provides a closer examination of the SSNR performance. Similarly, anomalies detected within the first 500 time steps are considered as faulty warnings.

The SSNR frequently detects faults from time step 600, indicating anomalies well before the system alarm. However, the SPRT exhibits a more stable performance in comparison with the SSNR. This discrepancy can be attributed to the features utilized for evaluation. The SSNR calculates SSNR values across all features, whereas the SPRT only computes features

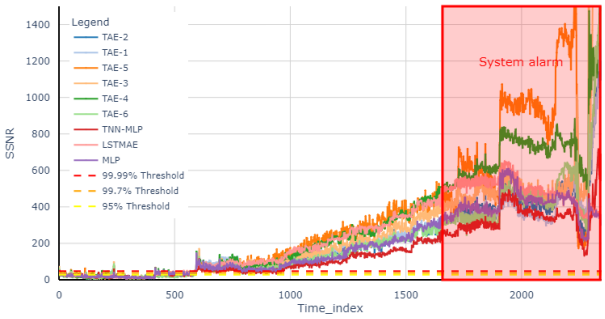

Figure 12. SSNR with full scale

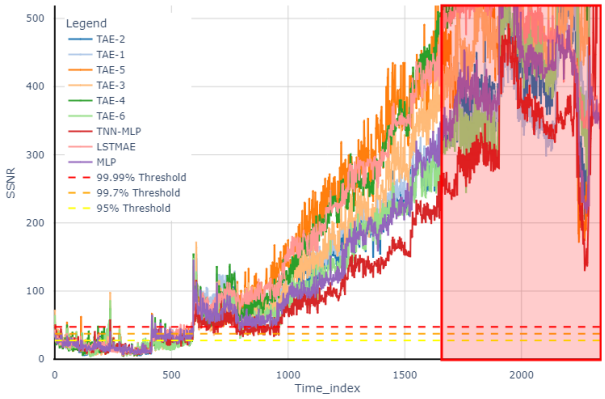

Figure 13. SSNR with selected scale

that surpass the threshold. Features failing to meet the threshold criteria are excluded from the overall SPRT evaluation.

6. CONCLUSION AND FUTURE WORK

This paper introduces a novel unsupervised approach for detecting anomalies in marine diesel engines, leveraging a transformer based autoencoder and residual evaluation methods utilizing SPRT and SSNR. The utilized dataset comprises normal and faulty operational data gathered under identical operating conditions, where the normal data are employed for model training and the faulty data for testing. A detailed description of the proposed architecture is provided, along with a comprehensive performance evaluation comparing the TAE with other models, including TNN-MLP, MLP, and LSTMAE models. In the performance evaluation, TAE models demonstrate superiority over other models in the early detection of anomalies.

The evaluation of observed and reconstructed data highlights the stable performance of the proposed TAE in timely anomaly detection. The TAE models are capable of detecting anomalies approximately 1000 time steps prior to the system raising an alarm. TNN-MLP also demonstrates commendable performance with 40% less training time comparing with TAE. Additionally, MLP and LSTMAE exhibit capability in anomaly detection before system alarms, but not as early as transformer-related models. Notably, transformer-related models require longer training time compared to neural networks without transformer architecture. However, a half transformer architecture, represented by TNN-MLP, also proves effective if stringent training time constraints apply.

In comparison to our previous study (Liang, Knutsen, Vanem, Zhang, & Æsøy, 2023), the current architecture undergoes a more comprehensive evaluation alongside neural networks with diverse architectures, sequence lengths, and parameters. The results provide stronger evidence supporting the enhancement of time series anomaly detection performance through the utilization of a transformer architecture. However, it is acknowledged that the model's performance could potentially be further improved with the use of larger training datasets.

ACKNOWLEDGMENT

This work has partly been carried out within the RealTOPs project with grant number 331634, supported by DNV and the Research Council of Norway. The authors report there are no competing interests to declare.

NOMENCLATURE

TNN	Transformer Neural Networks
AE	Autoencoder
$LSTM$	Long Short-Term Memory
MLP	Multilayer perceptron neural network
TAE	TNN based autoencoder
$TNN-MLP$	TNN as encoder and MLP as decoder
RNN	Recurrent Neural Networks
$LSTMAE$	RNN with LSTM based autoencoder
$SPRT$	Sequential Probability Ratio Test
CNN	Convolutional Neural Networks
$SSNR$	Sum of Squares of Normalized Residuals

REFERENCES

Bernardo, J. T., & Reichard, K. M. (2017). Trends in research techniques of prognostics for gas turbines and diesel engines. In *Annual conference of the phm society* (Vol. 9).

Brandsæter, A., Manno, G., Vanem, E., & Glad, I. K. (2016). An application of sensor-based anomaly detection in the maritime industry. In *2016 ieee international conference on prognostics and health management (icphm)* (pp. 1–8).

Brandsæter, A., Vanem, E., & Glad, I. K. (2019). Efficient online anomaly detection for ship systems in operation. *Expert Systems with Applications*, *121*, 418–437.

Ellefsen, A. L., Bjørlykhaug, E., Æsøy, V., Ushakov, S., & Zhang, H. (2019). Remaining useful life predictions for turbofan engine degradation using semi-supervised

deep architecture. *Reliability Engineering & System Safety*, *183*, 240–251.

Han, P., Ellefsen, A. L., Li, G., Holmeset, F. T., & Zhang, H. (2021). Fault detection with lstm-based variational autoencoder for maritime components. *IEEE Sensors Journal*, *21*(19), 21903-21912. doi: 10.1109/JSEN.2021.3105226

Han, P., Li, G., Skulstad, R., Skjong, S., & Zhang, H. (2020). A deep learning approach to detect and isolate thruster failures for dynamically positioned vessels using motion data. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1–11.

Hemmer, M., Klausen, A., Khang, H. V., Robbersmyr, K. G., & Waag, T. I. (2020). Health indicator for low-speed axial bearings using variational autoencoders. *IEEE Access*, *8*, 35842-35852. doi: 10.1109/ACCESS.2020.2974942

Hongchun, Y., Fanlun, X., & Xiaoyong, H. (2001). A method for deciding the number of hidden neurons of the feed-forward neural networks. *IFAC Proceedings Volumes*, *34*(26), 31-35. (4h IFAC/CIGR Workshop on Artificial Intelligence in Agriculture 2001, Budapest, Hungary, 6-8 June 2001) doi: https://doi.org/10.1016/S1474-6670(17)33628-5

Hu, K., Cheng, Y., Wu, J., Zhu, H., & Shao, X. (2021). Deep bidirectional recurrent neural networks ensemble for remaining useful life prediction of aircraft engine. *IEEE Transactions on Cybernetics*.

Knutsen, K. E., Liang, Q., Karandikar, N., Ibrahim, I. H. B., Tong, X. G. T., & Tam, J. J. H. (2022). Containerized immutable maritime data sharing utilizing distributed ledger technologies. In *Journal of physics: Conference series* (Vol. 2311, p. 012006).

Kriegeskorte, N., & Golan, T. (2019). Neural network models and deep learning. *Current Biology*, *29*(7), R231–R236.

Liang, Q., Knutsen, K. E., Vanem, E., Zhang, H., & Æsøy, V. (2023). Unsupervised anomaly detection in marine diesel engines using transformer neural networks and residual analysis. In *Phm society asia-pacific conference* (Vol. 4).

Liang, Q., Knutsen, K. E., Vanem, E., Æsøy, V., & Zhang, H. (2024). A review of maritime equipment prognostics health management from a classification society perspective. *Ocean Engineering*, *301*, 117619. doi: https://doi.org/10.1016/j.oceaneng.2024.117619

Liang, Q., Tvete, H., & Brinks, H. (2020). Prediction of vessel propulsion power from machine learning models based on synchronized ais-, ship performance measurements and ecmwf weather data. In *Iop conference series: Materials science and engineering* (Vol. 929, p. 012012).

Liang, Q., Tvete, H. A., & Brinks, H. W. (2019). Prediction of vessel propulsion power using machine learning on ais data, ship performance measurements and weather data. In *Journal of physics: Conference series* (Vol. 1357, p. 012038).

Listou Ellefsen, A., Han, P., Cheng, X., Holmeset, F. T., Æsøy, V., & Zhang, H. (2020). Online fault detection in autonomous ferries: Using fault-type independent spectral anomaly detection. *IEEE Transactions on Instrumentation and Measurement*, *69*(10), 8216-8225. doi: 10.1109/TIM.2020.2994012

Massoudi, M., Verma, S., & Jain, R. (2021). Urban sound classification using cnn. In *2021 6th international conference on inventive computation technologies (icict)* (pp. 583–589).

Pukelsheim, F. (1994). The three sigma rule. *The American Statistician*, *48*(2), 88–91.

Stalk, P. (2021). Review of maritime transport. *PUNITED NATIONS CONFERENCE ON TRADE AND DEVELOPMENT*, 1-177.

Tuli, S., Casale, G., & Jennings, N. R. (2022). Tranad: Deep transformer networks for anomaly detection in multivariate time series data. *arXiv preprint arXiv:2201.07284*.

Vanem, E., & Brandsæter, A. (2021). Unsupervised anomaly detection based on clustering methods and sensor data on a marine diesel engine. *Journal of Marine Engineering & Technology*, *20*(4), 217–234.

Vanem, E., & Storvik, G. O. (2017). Anomaly detection using dynamical linear models and sequential testing on a marine engine system. In *Annual conference of the phm society* (Vol. 9).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Wald, A. (1992). *Sequential tests of statistical hypotheses*. Springer.

Zhang, Z., Song, W., & Li, Q. (2022). Dual-aspect self-attention based on transformer for remaining useful life prediction. *IEEE Transactions on Instrumentation and Measurement*, *71*, 1-11. doi: 10.1109/TIM.2022.3160561

BIOGRAPHIES

Qin Liang works as a Senior Researcher in Group Research and Development - Maritime programe DNV. He worked as a Data Scientist in Rolls Royce Marine (2015-2018). In additon, he is currently pursuing the Ph.D. degree with the Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology (NTNU). His current research interests include ship performance, equipment condition monitoring, machine learning and deep learning.

Erik Vanem received the Cand. Scient. degree (Master of Science equivalent) in physics and the Ph.D. degree in statistics from the University of Oslo in 1996 and 2012, respectively. He worked three years at the Research Department of Telenor, three years at PGS Reservoir, one year at the Oslo University College, and spent some time at the Norwegian Defence Research Establishment. He has been working at legacy DNV R&I since 2003 on a number of research projects related to maritime safety and risk assessment. Since 2016, he has also been an Associate Professor with the University of Oslo in a 20% position. He is currently working as a Principal Researcher with the Maritime Transport Group, DNV Group Research and Development, Høvik, Norway. As a Researcher, he has authored and coauthored a number of papers in international journals and international conference proceedings and authored a recent monograph.

Knut Erik Knutsen is a Principal Researcher at DNV – Group Research and Development – Maritime Transport and team lead for the Data Driven Services group. He holds a PhD in Semiconductor Physics (2013) and a MSc in Materials, Energy and Nanotechnology (2009) from the University of Oslo. He also has experience as an Avionics Technician from the Royal Norwegian Airforce (1999-2004) where he was performing maintenance and calibration of aircraft equipment. Currently his research focuses on Data Driven Services and in particular data integrity solutions on the edge to increase the level of trust in data driven applications, and further enable novel digital solutions to support DNVs purpose of safeguarding life, property and the environment.

Vilmar Æsøy graduated from the Norwegian University of Science and Technology (NTNU), in 1989, and continued his research on natural gas fueled marine engines at NTNU MARINTEK, until 1997. He received the Ph.D. degree for his research on natural gas ignition and combustion through experimental investigations and numerical simulations, in 1996. From 1989 to 1997, he was engaged in several large R&D projects developing gas fueled engines and fuel injection systems for the diesel engine manufacturers, Wärtsilä, and Bergen Diesel, Roll-Royce. From 1998 to 2002, he was a R&D Manager for Rolls-Royce Marine Deck Machinery. Since 2002, he has been employed in teaching with the Ålesund University College, where he is also developing and teaching courses in marine product and systems design on bachelor's and master's level. In 2010, he received the green ship machinery professorship. His current research interest includes energy and environmental technology, with focus on combustion engines and the need for more environmental friendly and energy efficient systems.

Houxiang Zhang (M'04-SM'12) received the Ph.D. degree in mechanical and electronic engineering, in 2003, and the Habilitation degree in informatics from the University of Hamburg, in 2011. Since 2004, he has been a Postdoctoral Fellow and a Senior Researcher with the Department of Informatics, Faculty of Mathematics, Informatics and Natural Sciences, Institute of Technical Aspects of Multimodal Systems, University of Hamburg, Germany. He was with the Aalesund University College, in 2016. In 2011, he joined the Norwegian University of Science and Technology, Norway, where he is currently a Professor on mechatronics. His current research interests include biological robots and modular robotics, especially on biological locomotion control, and virtual prototyping in demanding marine operation. He has applied for and coordinated more than 20 projects supported by the Norwegian Research Council, German Research Council, and industry. In these areas, he has published over 160 journal and conference papers as author or co-author. He received four best paper awards and four finalist awards for Best Conference Paper at the International Conference on Robotics and Automation.