applied sciences

MDPI

Article

AI-Driven Predictive Maintenance in Modern Maritime Transport—Enhancing Operational Efficiency and Reliability

Dragos Simion [1,*], Florin Postolache [2], Bogdan Fleacă [1] and Elena Fleacă [1]

1 Faculty of Entrepreneurship, Business Engineering and Management, National University of Science and Technology Politehnica Bucharest, Splaiul Independenței 313, 060042 București, Romania; bogdan.fleaca@upb.ro (B.F.); elena.fleaca@upb.ro (E.F.)
2 Faculty of Marine Engineering, Romanian Naval Academy "Mircea cel Batran", Fulgerului 1, 900218 Constanța, Romania; florin.postolache@anmb.ro
* Correspondence: simion.di@yahoo.com; Tel.: +40-765355303

Abstract: Maritime transport has adapted to recent political and economic shifts by addressing stringent pollution reduction requirements, redrawing transport routes for safety, reducing onboard technical incidents, managing data security risks and transitioning to autonomous vessels. This paper presents a novel approach to predictive maintenance in the maritime industry, leveraging Artificial Intelligence (AI) and Machine Learning (ML) techniques to enhance fault detection and maintenance planning for naval systems. Traditional maintenance strategies, such as corrective and preventive maintenance, are increasingly ineffective in meeting the high safety and efficiency standards required by maritime operations. The proposed model integrates AI-driven methods to process operational data from shipboard systems, enabling more accurate fault diagnosis and early identification of system failures. By analyzing historical operational data, ML algorithms identify patterns and estimate the functional states, helping prevent unplanned failures and costly downtime. This approach is critical in environments where technical failures are a leading cause of incidents, as demonstrated by the high rate of machinery-related accidents in maritime operations. Our study highlights the growing importance of AI and ML in predictive maintenance and offers a practical tool for improving operational safety and efficiency in the naval industry. The paper discusses the development of a fault detection approach, evaluates its performance on real shipboard data-through tests on a seawater cooling system from an oil tanker and concludes with insights into the broader implications of AI-driven maintenance in the maritime sector.

Keywords: artificial intelligence; machine learning; kNN; maritime fault diagnosis; data-driven maintenance; predictive maintenance; maritime maintenance optimization

Citation: Simion, D.; Postolache, F.; Fleacă, B.; Fleacă, E. AI-Driven Predictive Maintenance in Modern Maritime Transport—Enhancing Operational Efficiency and Reliability. *Appl. Sci.* **2024**, *14*, 9439. https://doi.org/10.3390/app14209439

Received: 10 September 2024
Revised: 30 September 2024
Accepted: 12 October 2024
Published: 16 October 2024

1. Introduction

In the maritime industry, predictive maintenance has gained significant importance due to the complexity and scale of naval operations. The objective is to optimize maintenance activities by analyzing data provided by the system, thus enhancing operational efficiency and safety. Predictive maintenance involves transmitting operational data from sensors to storage units, where the onboard monitoring and control system triggers alarms and protections. Maintenance teams evaluate the received data, respond to alarms, and take appropriate actions to restore system functionality. This proactive approach aims to prevent system failures and minimize downtime, especially in high-risk maritime environments.

Fault diagnosis is crucial in the naval industry, given the high number of incidents related to technical failures. Between 2014 and 2023, machinery damage or failure accounted for 11,506 maritime incidents, almost four times the number of collisions, which totaled 3014 cases. In 2023 alone, over 50% of incidents were caused by technical equipment failures. This highlights the need for effective fault detection and diagnosis methods to prevent such occurrences. Maritime regions like the British Isles, North Sea, English Channel and the Bay of Biscay, where a significant portion of naval incidents occur, could benefit from improved fault diagnosis systems to ensure safer and more reliable operations [1].

Artificial Intelligence (AI) and Machine Learning (ML) have become indispensable in improving predictive maintenance. Traditional maintenance strategies, such as corrective and preventive maintenance, are increasingly insufficient to meet the stringent safety and efficiency standards required by the maritime industry. AI-based learning systems, which simulate the state of industrial processes or components using available measurement data, are widely adopted in maintenance engineering. These systems help increase equipment availability, reduce maintenance costs and enhance system reliability by predicting potential failures before they occur [2].

ML, in particular, plays a critical role in analyzing large volumes of multidimensional data, enabling operators to predict system failures and estimate Remaining Useful Life (RUL) [3]. By learning from historical operational data, ML applications can identify patterns and trends that signal potential equipment failures. This is especially important in maritime operations, where incidents often result from a chain of risky events. By applying ML techniques, maintenance teams can detect faults more accurately and quickly, improving overall system safety and operational efficiency.

Prognostics and Health Management (PHM), also known as Condition-Based Maintenance (CBM), addresses the limitations of traditional maintenance strategies [4]. PHM focuses on predicting and diagnosing system failures to optimize maintenance programs. It includes three main approaches: model-based, data-driven and hybrid methods [5]. Model-based approaches are highly accurate when system degradation is well-understood but require specialized expertise. Data-driven approaches, on the other hand, rely on large datasets and are more practical as they demand less specialized knowledge. Hybrid approaches combine the strengths of both methods to provide a comprehensive solution for predictive maintenance.

By integrating reliability and risk analysis methods with ML, we propose a novel algorithm for enhancing equipment maintenance activities. The proposed approach monitors the functional status of naval systems and detects early signs of failure, allowing maintenance teams to act before failures become safety risks. This ML-driven maintenance solution is particularly effective in maritime operations, where the complexity of the systems and the environment requires advanced predictive capabilities.

The use of AI in maintenance seeks to replicate human reasoning and decision-making processes but with greater efficiency. AI-driven maintenance management increases operational safety, optimizes maintenance plans based on real-time data, and enhances resource efficiency and cost control. The maritime industry is undergoing a digital transformation, with stakeholders—including ship designers, equipment manufacturers and shipping companies—collaborating to develop fully autonomous vessels. As part of this transformation, AI-based solutions play a key role in predictive maintenance, ensuring that technical systems operate at their full potential.

This study aims to develop a tool that leverages AI and ML techniques for fault detection in naval auxiliary systems. By analyzing operational parameters, we aim to identify trends in system component failures and improve maintenance planning. The proposed solution will provide a diagnosis tool and decision support system, enabling more precise and timely maintenance actions. Ultimately, this approach will help prevent future incidents, increase equipment efficiency and improve the safety and reliability of maritime operations.

The remainder of the paper is structured as follows: First, we discuss the foundational concepts and related work in the field. Subsequently, we provide an overview of naval equipment maintenance. Following this, the paper presents our methodology, proposes a fault detection algorithm and discusses the results of testing on a shipboard system. The research concludes with a summary, limitations and recommendations for future research directions.

2. Literature Review

Fault detection and diagnosis (FDD) ensures system reliability and safety by identifying, isolating and addressing faults using techniques like Fault Tree Analysis (FTA) and Failure Mode Effect Analysis (FMEA), while recent advances in AI and ML enhance predictive maintenance and optimize fault management in complex systems.

FDD involves identifying, isolating and understanding system faults to ensure the reliability, safety and efficiency of complex systems across various domains. The first step is detecting abnormalities using sensors, monitoring devices, data analysis techniques and expert knowledge. Once detected, the root cause is isolated by analyzing system behavior and sensor data through fault isolation techniques such as FTA, root cause analysis and model-based reasoning. Faults are then classified based on type, severity, duration and impact on performance. Diagnosis determines the fault's nature and effects, while prognosis predicts the system's future behavior considering fault progression. Fault diagnosis systems offer decision support tools, helping operators and engineers respond effectively to faults. Continuous system monitoring refines fault detection algorithms, improving fault isolation techniques and system reliability.

Fault diagnosis (FD) varies by system type and fault nature, but it is crucial for timely detection and resolution to prevent downtime and catastrophic failures and optimize system performance. Safety and reliability are critical in technical system design and operation, whether involving machines, vehicles, or complex installations. Probabilistic risk assessment (PRA) processes, including FTA, FMEA and Event Tree Analysis (ETA), are widely used to manage these risks. These methods depend on component failure data for quantitative analysis, though data collection is often incomplete due to monitoring limitations in maritime systems [6].

FTA, developed by Bell Telephone Laboratories in 1962 and later advanced by Boeing, analyzes potential failure causes and provides a logical, graphical model of system failures. It is a structured method for qualitative and quantitative failure analysis [7,8] but requires expert involvement and focuses only on potential failures [9–11]. FMEA, first used in the aerospace industry, identifies and prioritizes potential failure modes by assessing severity, occurrence and detectability, aiming to mitigate risks and improve system reliability [12]. When paired with Criticality Analysis (CA), FMEA becomes FMECA, which quantitatively prioritizes system vulnerabilities. Both FTA and FMEA are recommended for reliability applications, with research suggesting forward and backward integration of the methods for improved risk evaluation [13].

Recent studies have focused on enhancing maritime system reliability and risk assessment through the integration of FTA, FMEA and other techniques [14–18]. Research [19–22] focused on assessing risks and improving reliability in maritime systems. For example, some researchers have proposed reconfigurations to increase system reliability, others have used reliability block diagrams to modify ship systems, and some have applied advanced techniques like the Fuzzy Multi-Criteria Decision-Making Approach [21] (FMCDMA) and Dynamic Fault Tree Analysis (DFTA) to prioritize maintenance actions [22]. The Risk Priority Number (RPN) method (1) in FMEA prioritizes failure modes based on severity (S), occurrence (O) and detectability (D), though its simple multiplication formula has been widely critiqued.

$$\mathrm{RPN} = \mathrm{S} \times \mathrm{O} \times \mathrm{D}, \tag{1}$$

Alternative approaches [23–26], such as weighted RPN calculations and fuzzy logic, have been proposed to improve accuracy.

Moreover, the integration of AI and ML in predictive maintenance has gained significant attention, with AI techniques used to predict equipment failures and optimize maintenance schedules [27–29]. ML algorithms—supervised, unsupervised, semi-supervised and reinforcement learning—can automate tasks [30–32], discover patterns in data and make intelligent decisions without explicit human intervention. Recent applications of deep learning, neural networks and other ML algorithms like decision trees and support vector machines have shown promise in fault pattern recognition and classification in technical systems [33,34].

Finally, performance metrics for ML algorithms [35], such as the confusion matrix, help assess classification accuracy by comparing predicted labels with actual outcomes, providing insights into precision, recall and overall model performance.

Naval Maintenance

Degradation is traditionally considered a measured performance characteristic of cumulative changes over time leading to system failures [36]. Equipment maintenance aims to prevent the degradation of systems and thus reduce downtime. Traditionally, maintenance is classified based on whether it is performed as preventive or corrective maintenance [37]. Preventive maintenance is conducted to avert accidental equipment failure through scheduled repairs or the replacement of worn components, following established procedures. In addition, corrective maintenance involves performing the necessary repairs to restore the equipment to its working condition after a breakdown. Understanding the concept of maintenance is crucial despite the significant confusion associated with the terminology used to define the types of maintenance.

A systematic classification proposed by Trojan [38] considers factors such as associated risks, intervention modes, action planning, costs and available resources to assist decision-makers in choosing the most suitable type of maintenance for parts, equipment, facilities or systems, resulting in four major domains: reactive, proactive, predictive and advanced maintenance. Although traditional ship maintenance relies on the practical knowledge of ship personnel, modern maritime maintenance programs adhere to the standards set by the International Maritime Organization (IMO), employing written procedures and Planned Maintenance Systems (PMS) to ensure effective maintenance practices.

The development of Computerized Maintenance Management Systems (CMMS), akin to onboard PMS, was driven by advancements in computer operation systems and dedicated software creation. Since 2015, these programs have included modules for maintenance management and feature modules designed to manage information and enhance operational safety [39]. The introduction of CMMS in maritime transport has brought improvements in communication efficiency and speed with the office, easier monitoring of maintenance activities and procurement and simplification of data exchange processes. Currently, the maritime industry estimates the existence of over 70 CMMS programs with various operating features and designs [40].

Studies conducted by Wang Chaowe [41], Vlatko Kneževic [42] and Eriksen [43] and Park [44] addressed maintenance strategies and reliability in maritime systems. Wang Chaowe introduced Maintenance Progress-FMECA (MP-FMECA), which combines technical activities to prevent failure. Vlatko Kneževic focused on enhancing the maintenance plans and optimizing the turbocharger operation. Eriksen examined the limitations of Reliability Centered Maintenance (RCM) in autonomous ships, particularly in the main engine cooling system, emphasizing the need to consider the voyage duration when developing failure mode. Using data from alarm monitoring and ML, an algorithm was developed to discover anomalistic symptom judgments to be used for ship maintenance prediction [44]

3. Methodology

A comprehensive fault diagnosis methodology involves several steps to systematically detect, isolate and understand faults in a system. The proposed approach involves system understanding, data acquisition and preprocessing, feature selection and extraction, fault detection, fault localization and isolation, fault classification and severity assessment, diagnostic decision-making, feedback and continuous improvement.

Understanding the system architecture and expected behaviors is fundamental to fault diagnosis methodologies, ensuring a comprehensive grasp of critical components and operational norms. Acquiring and preprocessing relevant data involves collecting and refining information from sensors and monitoring devices and refining it through filtering and normalization to enhance accuracy. The selection of informative features is crucial for capturing meaningful patterns in the data, which facilitates effective fault detection and localization.

Applying fault diagnosis algorithms enables the identification of anomalies or deviations from normal system behavior, laying the foundation for subsequent diagnostic steps. Localizing and isolating faults involves narrowing down potential causes to specific components or subsystems, thereby aiding in the identification of root causes and effective mitigation strategies. Classifying and assessing fault severity helps prioritize response actions, ensuring that critical issues are addressed promptly to minimize disruptions and optimize system performance. Decision support systems play a pivotal role in guiding informed fault response decisions considering various factors such as safety, reliability and cost implications. Continuous feedback and improvement drive the evolution of fault diagnosis methodologies, ensuring their effectiveness in terms of addressing emerging insights and adapting to changes in system behavior.

To properly analyze a problem, it is necessary to utilize three primary data sources: operational parameters, maintenance history and the technical condition of components, to allow for the estimation of future values based on trends, seasonal variations and cyclical phenomena, in accordance with the theoretical principles of ML in the product–program lifecycle [45,46].

Fault detection and diagnosis methods, such as FTA, FMEA and ETA, are integral components of PRA methodologies. FTA, a systematic approach to identifying and analyzing potential system failures, is often utilized within PRA to assess the likelihood and consequences of various fault scenarios. Similarly, FMEA and ETA complement PRA by providing detailed evaluations of the failure modes and their effects, as well as the sequences of events following fault occurrence, respectively. In research [47], authors identified limitations in the traditional risk assessment method and proposed an improved analysis model using frequency–impact values plotted on a four-quadrant evaluation chart. Meanwhile, research [48] evaluated the failure modes using the FMECA method and prioritized the risk factors, which were then used for opportunity analysis regarding the implementation of Condition Monitoring Systems (CMS) and CBM practices. Potential failures in FTA-based naval propulsion systems are outlined in research [49]. The impact of disruptive events on critical ship components is assessed in research [50], which proposed a method for evaluating operational time without human intervention. Research [51] explored FTA's applicability across industries, including maritime ones and incorporated fuzzy logic to address event interdependencies. Also, research [52] analyzed failures in the propulsion systems of four identical ships, identifying key components for reliability calculations and suggesting improvements through Reliability Availability and Maintainability (RAM) analysis and FTA.

Furthermore, in fault diagnosis tasks, techniques like k Nearest Neighbors (kNN), Approximate Nearest Neighbors (ANN) and Condensed Nearest Neighbors (CNN) can be employed to classify and identify faults based on similarities with known fault patterns, which contributes to the overall reliability and safety analysis of complex systems. The ML algorithm used in this study is versatile for both classification and regression tasks, utilizing a simple approach to classify new cases based on a majority vote from its nearest neighbors, assessed through a distance function [53].

To determine the final accuracy of the tests, the confusion matrix is used. The confusion matrix is a performance evaluation tool in ML that represents the accuracy of a classification model, particularly for multi-class methods. It assesses the performance of a classification model by comparing predicted and actual class labels [54]. This provides insight into the classification model's operation and the types of errors it makes. The comparison of actual target values with those predicted by the model is displayed within an $n \times n$ matrix, where n represents the total number of target classes. The difference from the regular confusion matrix was in calculating the final results with the cumulative overall precision accuracy for the test data.

In this study, the proposed predicted algorithm is used to make predictions across multiple classes and follows a lifecycle guided by theoretical principles of ML, resulting in an algorithm adhering to this cycle, as depicted in Figure 1.

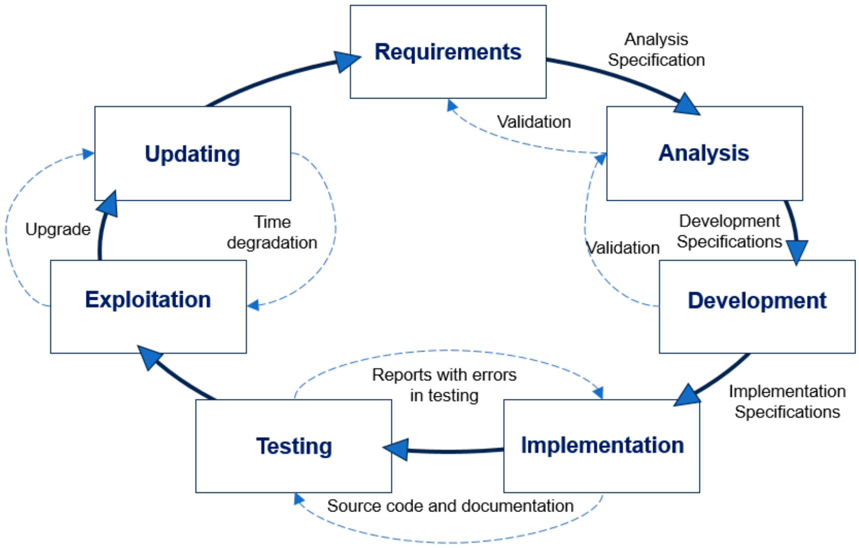

Figure 1. The life cycle of a program (adapted from [45,46]).

Beginning with developing requirements, the problem is stated, and a solution is proposed that emphasizes the purpose and benefits. The analysis evaluates the existing situation and proposes methods and algorithms for problem-solving, often using intuitive graphical representations for clarity.

The design involves abstracting the facts from the analysis to realize information and action modeling, while implementation and testing bring the product model to life through programming and debugging. Once completed, the algorithm is deployed for operation, and updates are provided to address issues and improve performance.

This life cycle repeats with each iteration of development. In a specific research context, analysis, design and implementation are essential for developing predictive data analysis programs using ML techniques that are adapted from existing works [55,56]. System analysis identifies critical points and variables, enabling the model's application to various installations and enhancing operational understanding and automation potential [57].

The algorithm further involves data collection from installation sensors and processing raw data to distinguish normal and faulty states through techniques like feature generation. This iterative process ensures a high-quality database, facilitating reliable analysis and results generation [58]. The algorithm creation process involves flexibility in terms of strategy and technique selection, which is tailored to the available data types and volume. Multiple techniques are simultaneously employed to build various predictive models, each offering various solutions to the problem at hand, the notable results will be considered in defining the optimal method.

When dealing with large-scale data, kNN can become computationally expensive due to the need to calculate distances between all data points. To address this, we take into account suitable solutions, such as ANN, Dimensionality Reduction, Data Sampling, CNN, Parallel Processing and Efficient Distance Metrics.

In high-dimensional datasets, distance calculations become more expensive (curse of dimensionality). To improve computational efficiency without losing too much information, techniques like Principal Component Analysis (PCA) help to reduce the number of dimensions. In our case, PCA projects the data onto a lower-dimensional space while retaining most of the variance, making kNN faster.

In the case of storing or processing historical data, techniques like random sampling or stratified sampling are used to maintain the dataset's statistical properties.

Some distance metrics (e.g., Euclidean distance) may be computationally expensive in high dimensions. Choosing faster distance metrics like Manhattan (Cityblock) distance or precomputing distances can reduce computation time.

Combining these techniques based on the nature of our available data and resources, we managed large-scale data efficiently with kNN.

Fault Detection Algorithm

This paper proposes a new fault detection approach, integrating risk analysis tools like FTA and FMECA, along with ML techniques, to enhance maritime maintenance practices. In addition, it introduces the Supplier Inputs Process Output Customer (SIPOC) diagram as a graphical representation tool to illustrate the inputs, utility and benefits of a maritime maintenance system. Originating from Edward Deming's Total Quality Management principles, the SIPOC diagram visualizes processes as integrated entities, emphasizing the interconnectedness of various elements (Figure 2).

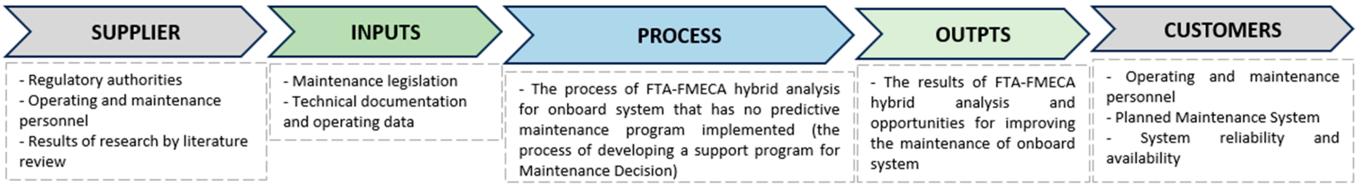

Figure 2. The SIPOC diagram for integrating research methods.

This concept portrays manufacturing as a complete and integrated process rather than a series of disjointed elements, aiding in understanding the diverse applications and stages preceding and succeeding any action. To pinpoint real issues within the manufacturing process, SIPOC components include both internal and external customers, product requirements and the inputs and outputs of the process [59,60].

The proposed approach effectively analyzes a problem and utilizes three primary data sources: operating parameters encompassing both normal and faulty operations, detailed maintenance history and the technical condition of components, including degradation over time, to estimate operating hours until failure [44]. By analyzing the recorded parameter values, it is possible to estimate values for future moments, the type of trend (long-term increase or decrease in values), seasonal phenomena (changes in values over equal time periods) and cyclical phenomena (fluctuations that do not always have the same duration, i.e., are not periodic).

The proposed algorithm named FaultApp, tailored to meet the operational and maintenance requirements of a ship's crew, utilizes planned activities and data from onboard monitoring systems to efficiently manage information, provide insights and enhance operational and maintenance processes through five structured operational modules, which are integrated for comprehensive functionality as described in Algorithm 1.

Algorithm 1. FaultApp

```
1.While data are being imported (import_data):
2.  Set MonitoringArray to the import_data;
3.  Call the MONITOR function;
4.  Call the GRAPH function.
5.    If a deviation is detected:
6.      Call the ANALYSIS function;
7.      Call the DIAG function;
8.      Call the LINK_PMS function;
9.    end if;
10.end the while loop.
```

Algorithm 1 has a universal character, is applicable to monitoring all installations onboard a ship and contains five modules: MONITOR and GRAPH were enhanced based on similar monitoring systems onboard, while DIAG, ANALYSIS and LINK_PMS were developed to conduct a technical state analysis of shipboard installations. Particularly, for the implementation, testing and operation stages, an onboard ship installation (e.g., seawater cooling system) was chosen to validate the results.

The algorithm utilizes operational parameters from sensors installed on the ship's system and calculates additional parameters by comparing input and output values for the same component. It also evaluates the efficiency of redundant system components, storing this information in a Monitoring Array used by the MONITOR, DIAG, and GRAPH modules. The algorithm transmits real-time data to a database, flags functional imbalances, and saves data for analysis, identifying defects and deviations from nominal values.

The MONITOR module displays, in real-time, within the interface of the chosen installation, the values of the functional parameters obtained from the sensors.

The DIAG module (2) provides a graphical representation of the evolution of the operating data values according to the user's preference over set time intervals. The deviation of these parameters lies between the minimum and maximum values accepted for the optimal operating regime. Any value exceeding the maximum or minimum limits (values exceeding 100%) indicates an abnormal functional state. To normalize the data presented in the graph, the domain (−100, 100) was utilized, and deviations from optimal parameters were measured as percentage values, where −100% and 100% denote negative and positive deviations, respectively, and 0 represents the optimal operation.

This information is required to correlate the functional states of the components with the values of the operating data.

Step 1. Initialize GraphData as MonitoringArray;
Step 2. Determine the dimensions of GraphData and store them in variables m (rows) and n (columns);
Step 3. Loop through each element of GraphData:
- For each row index i from 1 to m:
    - For each column index j from 1 to n:
        a. If GraphData(i,j) == min_max(2,j):
            - Set DiagArray(i,j) to 0.
        b. Else if GraphData(i,j) < min_max(2,j) and ≥ min_max(1,j):
            - Compute and set DiagArray(i,j) as follows: (2)
              (−1 + (GraphData(i,j) − min_max(1,j))/(min_max(2,j) − min_max(1,j))) × 100.
        c. Else if GraphData(i,j) > min_max(2,j) and ≤ min_max(3,j):
            - Compute and set DiagArray(i,j) as follows:
              (GraphData(i,j) − min_max(2,j))/(min_max(3,j) − min_max(2,j)) × 100.
        d. Otherwise,
            - Set DiagArray(i,j) to fault value.
End the loops.

The ANALYSIS module, having possible faults in the functional state as reference points, analyzes the provided dataset online or offline to determine the trend for each variable. The module extracts the time and variables provided by the sensors from the installation under test, sets a threshold for small variations and fits a linear model to each variable (3) to analyze, determine and display trends for each variable (5) based on a model (4). Thus, depending on the size of the database or the multitude of data provided by the sensors placed on board the ship, combining these techniques (kNN, ANN, CNN, PCA, EDM) based on the nature of the available data and resources, we can efficiently handle large-scale data with kNN.

```
mdl = fitlm(time,sensors(:,i));
Mdl = fitcknn(X,Y,'NumNeighbors',5,'Standardize',1);;
Mdl = fitcknn(X, Y, 'Distance', 'cityblock');
Mdl = KDTreeSearcher(import_data); [idx, dist] = knnsearch(mdl, query
Point, 'K', 5);                                                          (3)
[coeff, score] = pca(import_data);
reducedData = score(:, 1:k); % Keep top k principal components
Mdl = fitcknn(reducedData, labels, 'NumNeighbors', 5);
```

```
slope = mdl.Coefficients.Estimate,                                       (4)
```

```
trends{i} = determineTrend(slope,threshold),                             (5)
```

Taking into account that our scenario dataset is small to moderately sized and low-dimensional and the fault classes are well-separated, for fault diagnosis, the kNN model is most suitable based on its simplicity and non-parametric nature, which makes it effective for real-time fault diagnosis and applications in which minimal assumptions about data distribution are required.

The module trains a kNN classifier where the number of nearest neighbors in the predictors (k) is 5. The kNN model (Mdl) is created to classify data points based on the given features and labels and ensures that the numeric data are standardized to realize better performance (6).

```
Mdl = fitcknn(X,Y,'NumNeighbors',5,'Standardize',1),                     (6)
```

The predicted class labels are generated to evaluate the model's performance on the training set using trained regression models (7). Possible increasing or decreasing trends are translated into +1 and -1, respectively. A very small fluctuation is transferred to the value 0. Every possible failure mode of the system is characterized by a unique combination of trend values (+. 0, -1) of each monitored parameter.

```
predictedY = resubPredict(Mdl),                                          (7)
```

A suitable ML technique for multi-class fault diagnosis depends on the specific characteristics of the dataset and the diagnosis requirements. In the proposed algorithm, the kNN model is best fitted for fault diagnosis based on a predefined fault and confusion array because the dataset is small to moderately sized, the feature space is low-dimensional, and the fault classes are well separated. In this case study, the simplicity, ease of implementation and effectiveness of the proposed method in scenarios with distinct clusters of faults make it a viable choice for real-time and straightforward fault diagnosis applications. However, careful consideration of the dataset's size and dimensionality is critical to ensure kNN's efficiency and accuracy in fault diagnosis.

Therefore, a confusion array/chart and the derived metrics are crucial for assessing performance and guiding the selection of the best model for fault diagnosis, providing a summary of prediction results by showing the counts of true positives, false positives, true negatives and false negatives. The confusion chart (cm) visually represents the performance of the kNN classifier by displaying the actual versus predicted class labels, which helps assess the model's accuracy and identify any misclassifications (8).

The confusion chart's rows correspond to the true class, the columns correspond to the predicted class and diagonal and off-diagonal cells correspond to correctly and incorrectly classified observations, respectively.

$$\text{cm} = \text{confusionchart(Y,predictedY)}, \tag{8}$$

$$\text{CM} = \text{cm.NormalizedValues}, \tag{9}$$

The normalized confusion matrix (CM) provides a more interpretable view of the classification performance by showing the proportion of each predicted class relative to the actual class, which can be useful for comparing performance across different classes (9).

Thus, in the proposed algorithm, the typology of the faults was structured as an array (the MonitoringArray) corresponding to the 12 functional states, each containing 41 functional parameters, described in Section 4.

Algorithm testing, validation and updating were performed using test files obtained during the operation of the onboard installations, online or offline, during or after the trip was completed. Using kNN, we classify a new data point by analyzing and voting on the classes of the k closest points from the initial dataset and visualize the predictions in the confusion matrix chart created from the true labels and the predicted labels, and we compare the performance across different classes. In the end, the algorithm indicates the defects in order of their incidence.

The GRAPH module allows a graphical representation of the operating data values according to the user option. The module serves as an essential tool for operational personnel to observe changes over time in monitored and calculated parameter values, functioning as an analysis instrument for failure modes in cases where the analysis report from the preceding module does not provide sufficient information for classification, when results are very close in value, or when necessary, analysis trends are not identified. Using data stored in the Monitoring Array matrix for monitored pressure and temperature parameters, the algorithm enables real-time or continuous two-dimensional (2D) displays of the target parameters throughout the recording period. The information is necessary to correlate the functional state of the components with the values of the operating data.

The LINK_PMS module assists users in managing maintenance processes by providing relevant information on facility failures and displaying possible causes and remedial measures from FTA and FMECA analyses in a table format. It integrates with PMS on transport vessels to demonstrate the key features of the affected components and details of recent maintenance work.

4. Results

The algorithm was tested on a centralized seawater cooling system from a tanker ship. This system is designed to cool the freshwater liquid that is forced circulated by designated pumps in a closed circuit through heat exchangers and absorbs the heat from engine room equipment and machinery. The interest parameters are as follows:

- Twenty basic operational parameters provided by sensors at the inlet and outlet for each component;
- Ten calculated parameters, referred to as "Delta" (difference between inlet and outlet values on the same component of the system);
- Eleven calculated parameters, referred to as "Load" (in terms of efficiency for redundant components of the systems), to identify unbalance between similar components that are used for an improved prediction using the kNN classifier for the ANALYSIS module.

These parameters are stored as a matrix called the ''Monitoring Array", which serves as input data for subsequent modules (MONITOR, DIAG and GRAPH). The initial module transmits real-time operational data to a database, and the algorithm's novelty lies in its calculation of DELTA parameters, which are crucial for signaling functional imbalances in component failures.

4.1. The MONITOR Module

The MONITOR module (Figure 3) displays the real-time values of the functional parameters obtained from the sensors within the interface of the selected installation.

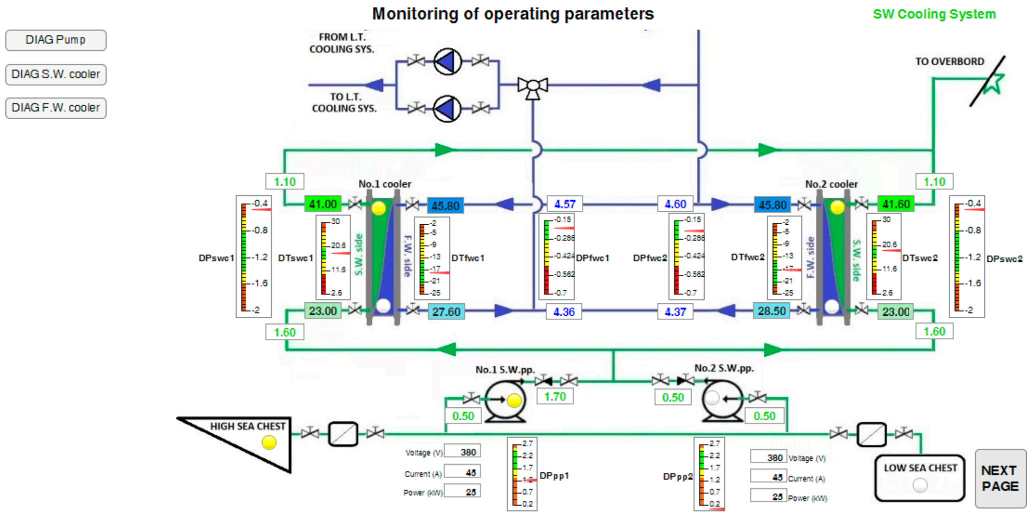

Figure 3. Graphical interface for MONITOR module.

After each display, the data are saved for the analysis phase in the ANALYSIS module, which also offers additional data calculation and displays defect states representing deviations from nominal parameter values.

4.2. The DIAG Module

The DIAG module (Figure 4) graphically represents the evolution of the operating data values over user-defined time intervals, highlighting deviations (10) within a normalized domain of (−100, 100) where −100% and 100% denote deviations and 0 represents optimal operation, where any value exceeding these limits indicates an abnormal functional state, thereby correlating the component functionality with the operating data values. The range of optimum values for each monitored parameter represents the nominal operating values given by the manufacturer. Any value outside them is categorized as a deviation from the nominal operation. The minimum and maximum values, noted with min_max, are assimilated to the equipment fault operation (in which it runs with very low performance or overload, close to the safety limit).

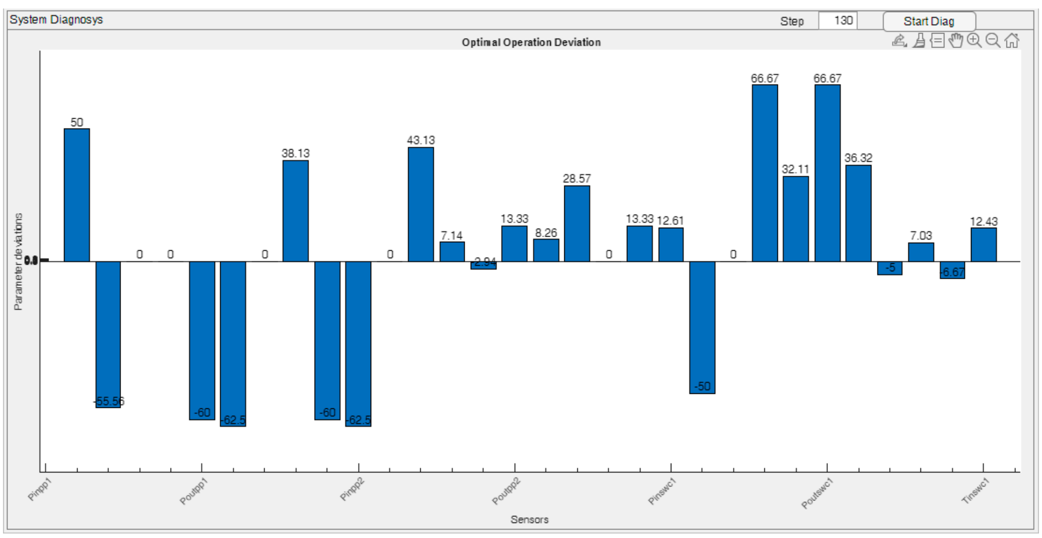

Figure 4. DIAG module results.

4.3. The ANALYSIS Module

The ANALYSIS module examines sensor data to determine trends and detect faults by fitting linear regression models to each variable and using a kNN classifier for real-time fault diagnosis. It standardizes numeric data, visualizes predictions with a confusion matrix to assess model performance and indicates defects based on their incidence. The proposed approach leverages kNN's simplicity and effectiveness of kNN on small, well-separated datasets, making it suitable for real-time fault diagnosis in maritime installations. To validate the results of the algorithm, 12 functional states on the tested installation, 12 functional states were taken as a reference. Following the specific analysis, for the targeted installation, a malfunction matrix (Fault Array) was created for 12 existing functional states, 11 functional states with deviations from optimal parameters and 1 operating state with nominal parameters (and in this situation, some fluctuations of parameter values were identified). The functional states are as follows:

- F1—seawater chest malfunction;
- F2—seawater pump malfunction;
- F3—clogging main seawater cooler;
- F4—clogging main freshwater cooler;
- F5—low-efficiency main freshwater cooler;
- F6—clogging secondary seawater cooler;
- F7—clogging secondary freshwater cooler;
- F8—low-efficiency secondary freshwater cooler;
- F9—seawater pump failure and clogging main sea water cooler;
- F10—seawater pump failure and low-efficiency main freshwater cooler;
- F11—clogging main seawater cooler and low-efficiency main freshwater cooler;
- F12—operation with minimal deficiencies.

In the proposed algorithm, fault types were structured as arrays corresponding to 12 functional states, each containing 41 functional parameters defined above. The algorithm was tested, validated and updated using test files from onboard installations, either online or offline, during or after trips. Using kNN, new data points are classified by analyzing and voting on the classes of the k closest points from the initial dataset, with predictions visualized in a confusion matrix to compare performance across classes, ultimately indicating defects in order of incidence and descending highlighting possible failure modes (Figure 5).

Furthermore, incorporation of these techniques—noise filtering, outlier detection, robust ML models and redundancy—the approach can indeed operate under interference and disturbances. However, the success of this approach will depend on how well the system is designed to detect and mitigate interference while maintaining accuracy in predictions and fault diagnosis. Regular calibration, monitoring and validation against real-world scenarios enhance system reliability in the face of interference.

Instead of relying on fixed thresholds for detecting anomalies or faults, the implementing Dynamic Thresholds can adapt based on historical data and the current state of the system and accommodate variations in sensor readings due to environmental factors. Also, Self-Learning Mechanisms can adapt over time by learning from new data, thus improving the algorithm's robustness against changes in data patterns due to disturbances.

The algorithm indicates the defects in the order of their incidence, as is depicted in Figure 6.

Figure 5. KNN classification in the ANALYSIS module.

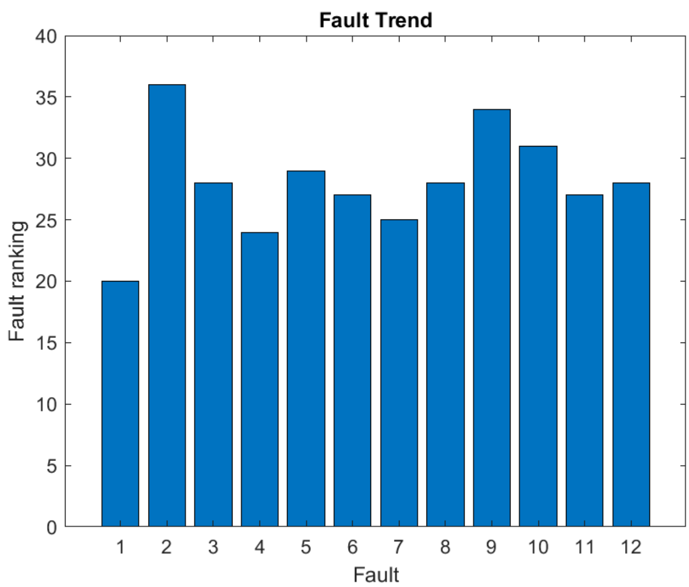

Figure 6. Fault hierarchy in the ANALYSIS module.

4.4. The GRAPH Module

The GRAPH module (Figure 7) graphically represents the evolution of operating data values based on user preferences, serving as a crucial tool for operational personnel to monitor changes in parameter values over time, analyze failure modes when the previous module's report is insufficient and display real-time or continuous 2D data from the Monitoring Array matrix for monitored parameters (pressure and temperature) to correlate the component functional states with the operating data.

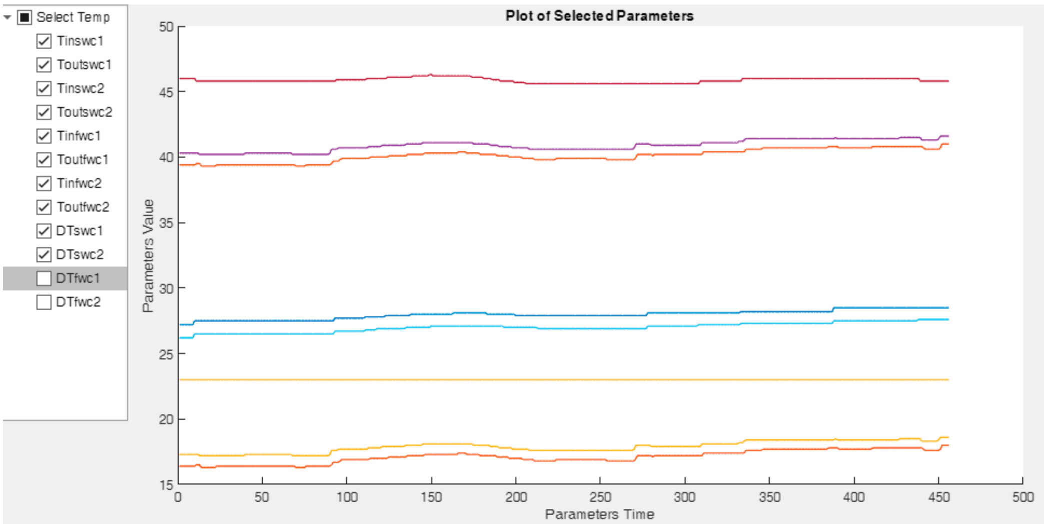

Figure 7. Two-dimensional graphical representation for the selected parameters in the GRAPH module.

4.5. The LINK_PMS Module

The LINK_PMS module is designed to provide users with relevant information for managing the maintenance process based on the type of component failure. Possible causes and remedial measures for the identified defects extracted from qualitative analyses of FTA and FMECA are displayed.

5. Discussion

This paper proposes an innovative approach to enhancing the reliability and safe operation of equipment by implementing a new model for the automated evaluation of the technical conditions of equipment using ML techniques. The objective is to provide early warning of operational deviations, eliminate human errors, notify upcoming maintenance stages in advance and reduce costs.

In developing the algorithm, several methods were considered, including fault tree analysis, failure mode and effects analysis (FMEA) and fault detection using ML techniques. Using the SIPOC diagram, all these methods were integrated to construct the algorithm. The algorithm was designed to handle both continuous data flow from onboard the ship and offline analysis by exporting recorded data for shore-based analysis during and after a voyage. Based on specified requirements and development possibilities and through the integration of ML techniques, the algorithm facilitates the monitoring of operations and the evaluation of the functional state of naval installations by analyzing operational data to assist maintenance decision-making.

The algorithm enhances the monitoring phase of onboard installations by alerting potential defects and their effects on the components. In addition, the proposed algorithm facilitates the visualization of trends and functional deviations, mapping them to possible failure modes.

Scalability is achievable, accommodating the complexities of installations and enabling the integration of additional sensors. The scalability of the algorithm can be demonstrated and validated by Handling Large Datasets, which involves progressively introducing larger datasets by simulating extra sensor data or incorporating historical records into the monitoring process. Additionally, Validation Through Predictive Accuracy can assess the algorithm's capacity to identify potential defects and correlate them with failure modes as system complexity increases. Stress Testing for Real-Time Monitoring serves as a method to conduct stress tests by simulating a high volume of real-time sensor data from ship systems under various operational conditions, including peak loads and fault occurrences.

The obtained results contribute to the field of maintenance for ship systems and installations in several ways: (1) the integration of risk and reliability analysis (FTA and FMECA), simulation tools and ML techniques is achieved using the SIPOC method; (2) a database has been created to identify fault precursors by analyzing operational parameter provided by ship sensor and calculated parameters for measuring the load and efficiency of system components; (3) the monitoring system and graphical analysis of operational data have been enhanced; (4) a fault matrix has been constructed to identify possible failure modes in naval installations; (5) the analysis of operational parameter trends is integrated with ML techniques to identify potential failure modes in naval installations. Finally, an algorithm was developed leveraging the research findings to enhance the current monitoring system and assist in maintenance decision-making for naval installations. The primary advantage lies in the reduced time required for diagnosing the root cause of failures through rapid system evaluation using operational data. However, testing and validating the proposed algorithm on a generic installation revealed potential limitations arising from the complexity of the employed methods and tools.

Methodologically, two risk analysis methods (FTA and FMECA) were employed, which provided the best results in the naval field. The limitations of each method were minimized by integrating them into the research. Although the choice of installation for the case study was thoroughly justified, applying the algorithm to a single installation was a limitation of the study. Due to the varying technological levels of the testing installation (limited number of monitored parameters, depending on the sensors placed on the installation), evaluating trends can be challenging. However, by calculating additional parameters, functional trends become evident, and the early detection of these trends can differentiate between possible failure modes.

Considering the above points, combined with the rapid technological development and application of ML techniques, the proposed algorithm represents a significant advancement in naval maintenance. It contributes to the safe operation of naval equipment by immediately reporting parameter deviations and improving the activities of personnel involved in ship operation, thereby enhancing autonomous maritime transport.

Furthermore, new research directions are thus outlined, aiming at the development of an application that enables (1) automation of the diagnosis process for naval equipment by integrating the application into the ship's maintenance management system; (2) application of the study to other onboard installations by replicating the steps described in the SIPOC diagram; (3) mapping fault matrices for other naval installations and automating the diagnosis process; (4) updating the application with new technological capabilities (additional sensors for measuring operational parameters) to unequivocally differentiate between the various and complex possible failure modes; (5) creating support materials (operating manuals and training videos) to develop new competencies for operating personnel.

This work aims to serve as a foundation for developing an application that integrates the study conducted by creating a pilot model to diagnose faults in naval installations and evaluate operational parameters. The proposed model is intended to estimate functional deviations and assist maintenance decision-making.

6. Conclusions

This research offers a significant contribution to the field of naval maintenance by advancing fault diagnosis automation, improving the monitoring of onboard systems and supporting the development of autonomous maritime transport.

Incorporating ML algorithms like kNN, ANN and CNN into fault diagnosis tasks improves fault classification by comparing new cases with known fault patterns. These techniques effectively classify and identify faults by comparing new data points to known fault patterns, contributing to the overall reliability and safety of the system. Performance evaluation tools, such as the confusion matrix, assess the classification model's accuracy, providing insights into errors and overall model performance.

The proposed algorithm in this study, designed for multi-class predictions, follows an ML lifecycle involving problem identification, analysis, design, implementation and testing, with iterative development to refine the model. The algorithm demonstrates the potential to improve fault detection, classification and visualization of operational trends, enabling more informed and timely maintenance decisions for complex installations. By combining strategies for efficiency (dimensionality reduction, efficient distance metrics), robustness (cross-validation, noise handling), flexibility (adaptive parameters, weighted kNN), and adaptability (incremental learning, real-time processing), the algorithm can perform effectively across a wide range of general-purpose scenarios. This makes it suitable for diverse real-world applications, from fault detection in industrial systems to dynamic decision-making in online environments.

Future research should focus on expanding the algorithm's applicability to other naval systems, integrating more sensors and developing tools for training and personnel development to optimize system operation and maintenance management.

Author Contributions: Conceptualization, D.S., E.F. and F.P.; methodology, D.S., E.F. and F.P.; software, F.P.; validation, D.S., E.F., F.P. and B.F.; formal analysis, D.S., E.F. and B.F.; investigation, D.S., E.F. and B.F.; resources, D.S. and F.P.; data curation, E.F. and B.F.; writing—original draft preparation, D.S. and F.P.; writing—review and editing, F.P.; visualization, D.S., E.F., F.P. and B.F.; supervision, E.F. and F.P.; project administration, D.S. and F.P.; funding acquisition, E.F. and B.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National University of Science and Technology Politehnica Bucharest under Grant PubArt. The APC was funded by PubArt.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Safety and Shipping Review. 2024, Allianz Commercial, Online. Available online: https://commercial.allianz.com/content/dam/onemarketing/commercial/commercial/reports/Commercial-Safety-Shipping-Review-2024.pdf (accessed on 19 August 2024).
2. Konieczny, J.; Stojek, J. Use of the K-Nearest Neighbour Classifier in Wear Condition Classification of a Positive Displacement Pump. *Sensors* **2021**, *21*, 6247. [CrossRef] [PubMed]
3. Lee, J.; Wu, F.; Zhao, W.; Ghaffari, M.; Liao, L.; Siegel, D. Prognostics and Health Management Design for Rotary Machinery Systems—Reviews, Methodology and Applications. *Mech. Syst. Signal Process.* **2014**, *42*, 314–334. [CrossRef]
4. Zhao, Z.; Liang, N.B.; Wang, X.; Lu, W. Remaining Useful Life Prediction of Aircraft Engine Based on Degradation Pattern Learning. *Reliab. Eng. Syst. Saf.* **2017**, *164*, 74–83. [CrossRef]
5. Li, X.; Ding, Q.; Sun, J.-Q. Remaining Useful Life Estimation in Prognostics Using Deep Convolution Neural Networks. *Reliab. Eng. Syst. Saf.* **2018**, *172*, 1–11. [CrossRef]
6. Kabir, S.; Papadopoulos, Y. A Review of Applications of Fuzzy Sets to Safety and Reliability Engineering. *Int. J. Approx. Reason.* **2018**, *100*, 29–55. [CrossRef]
7. Ericson, C. Fault Tree Analysis—A History. In Proceedings of the 17th International System Safety Conference, Orlando, FL, USA, 16–21 August 1999; Scientific Research Publishing: Wuhan, China, 1999. Available online: https://www.scirp.org/reference/referencespapers?referenceid=1049972 (accessed on 7 July 2024).
8. Barbelian, M.A.; Bălan, C.V. Fault Tree Event Classifiction by Neural Network Analysis. *UPB Sci. Bull. Ser. D* **2017**, *79*, 165–176. Available online: https://www.scientificbulletin.upb.ro/rev_docs_arhiva/full03e_888924.pdf (accessed on 22 July 2024).
9. Lind, M.; Zhang, X. Functional modeling for fault Diagnosis and its application for NPP. *Nucl. Eng. Technol.* **2014**, *46*, 753–772. [CrossRef]

10. Lin, J.; Yuan, Y.; Zhang, M. Improved FTA Methodology and Application to Subsea Pipeline Reliability Design. *PLoS ONE* **2014**, *9*, e93042. [CrossRef]
11. Liu, P.; Wu, Y.; Li, Y.; Wu, X. An Improved FMEA Method Based on the Expert Trust Network for Maritime Transportation Risk Management. *Expert Syst. Appl.* **2024**, *238*, 121705. [CrossRef]
12. Carpitella, S.; Certa, A.; Izquierdo, J.; La Fata, C.M. A Combined Multi-Criteria Approach to Support FMECA Analyses: A Real-World Case. *Reliab. Eng. Syst. Saf.* **2018**, *169*, 394–402. [CrossRef]
13. Liu, P.; Li, Y. An Improved Failure Mode and Effect Analysis Method for Multi-Criteria Group Decision-Making in Green Logistics Risk Assessment. *Reliab. Eng. Syst. Saf.* **2021**, *215*, 107826. [CrossRef]
14. Lazakis, I.; Raptodimos, Y.; Varelas, T. Predicting Ship Machinery System Condition through Analytical Reliability Tools and Artificial Neural Networks. *Ocean Eng.* **2018**, *152*, 404–415. [CrossRef]
15. Emovon, I.; Norman, R.A.; Murphy, A.J.; Pazouki, K. An Integrated Multicriteria Decision Making Methodology Using Compromise Solution Methods for Prioritising Risk of Marine Machinery Systems. *Ocean Eng.* **2015**, *105*, 92–103. [CrossRef]
16. Liu, N.; Guo, N.; Zhang, N. An Improved Assessment Method for FMEA for a Shipboard Integrated Electric Propulsion System Using Fuzzy Logic and DEMATEL Theory. *Energies* **2019**, *12*, 3162. [CrossRef]
17. Shafiee, M.; Enjema, E.; Kolios, A. An Integrated FTA-FMEA Model for Risk Analysis of Engineering Systems: A Case Study of Subsea Blowout Preventers. *Appl. Sci.* **2019**, *9*, 1192. [CrossRef]
18. Liu, C.-T.; Hwang, S.-L.; Lin, I. Safety Analysis of Combined FMEA and FTA with Computer Software Assistance—Take Photovoltaic Plant for Example. *IFAC Proc. Vol.* **2013**, *46*, 2151–2155. [CrossRef]
19. Allal, A.A.; Mansouri, K.; Youssfi, M.; Qbadou, M. Toward a Reliable Main Engine Lubricating Oil System for a Safe Operation of Autonomous Ship. In Proceedings of the 2nd International Conference on System Reliability and Safety (ICSRS), Milan, Italy, 20–22 December 2017; Volume 2017.
20. Dionysiou, K.; Bolbot, V.; Theotokatos, G. A functional model-based approach for ship systems safety and reliability analysis: Application to a cruise ship lubricating oil system. *Proc. Inst. Mech. Eng. M* **2022**, *236*, 228–244. [CrossRef]
21. Velasco-Gallego, C.; Lazakis, I. RADIS: A Real-Time Anomaly Detection Intelligent System for Fault Diagnosis of Marine Machinery. *Expert Syst. Appl.* **2022**, *204*, 117634. [CrossRef]
22. Daya, A.A.; Lazakis, I. Developing an advanced reliability analysis framework for marine systems operations and maintenance. *Ocean Eng.* **2023**, *272*, 113766. [CrossRef]
23. Bluvband, Z.; Polak, R.; Grabov, P. Bouncing failure analysis (BFA): The unified FTA-FMEA methodology. In Proceedings of the Annual Reliability and Maintainability Symposium, Alexandria, VA, USA, 24–27 January 2005; pp. 463–467.
24. Franceschini, F.; Galetto, M. A new approach for evaluation of risk priorities of failure modes in FMEA. *Int. J. Prod. Res.* **2001**, *39*, 2991–3002. [CrossRef]
25. Mandal, S.; Maiti, J. Risk analysis using FMEA: Fuzzy similarity value and possibility theory based approach. *Expert Syst. Appl.* **2014**, *41*, 3527–3537. [CrossRef]
26. Gupta, G.; Ghasemian, H.; Janvekar, A.A. A Novel Failure Mode Effect and Criticality Analysis (FMECA) Using Fuzzy Rule-Based Method: A Case Study of Industrial Centrifugal Pump. *Eng. Fail. Anal.* **2021**, *123*, 105305. [CrossRef]
27. Machine Learning Yearning, Online. Available online: https://nessie.ilab.sztaki.hu/~kornai/2020/AdvancedMachineLearning/Ng_MachineLearningYearning.pdf (accessed on 30 September 2024).
28. Lee, J.; Ardakani, H.D.; Yang, S.; Bagheri, B. Industrial Big Data Analytics and Cyber-Physical Systems for Future Maintenance & Service Innovation. *Procedia CIRP* **2015**, *38*, 3–7. [CrossRef]
29. Mobley, R.K. *An Introduction to Predictive Maintenance*; Elsevier: Amsterdam, The Netherlands, 2002.
30. Lughofer, E.; Sayed-Mouchaweh, M. *Predictive Maintenance in Dynamic Systems*; Springer: Berlin/Heidelberg, Germany, 2019; ISBN 978-3-030-05645.
31. Dhillon, B.S. *Reliability, Maintainability and Safety for Engineers*; CRC Press: Boca Raton, FL, USA, 2020; ISBN 978-0367352653.
32. Naeem, S.; Ali, A.; Anam, S.; Ahmed, M.M. An unsupervised machine learning algorithms: Comprehensive review. *Inter-Natl. J. Comput. Digit. Syst.* **2023**, *13*, 911–921. [CrossRef] [PubMed]
33. Carvalho, T.P.; Soares, F.A.A.M.N.; Vita, R.; Francisco, R.d.P.; Basto, J.P.; Alcalá, S.G.S. A systematic literature review of machine learning methods applied to predictive maintenance. *Comput. Ind. Eng.* **2019**, *137*, 106024. [CrossRef]
34. Joshi, A.; Khosravy, M.; Gupta, N. Machine learning for predictive analysis. In *Proceedings of the ICTIS 2020*; Springer: Berlin/Heidelberg, Germany, 2020.
35. Pommé, L.-E.; Bourqui, R.; Giot, R.; Auber, D. Relative Confusion Matrix: Efficient Comparison of Decision Models. In Proceedings of the 2022 26th International Conference Information Visualisation (IV), Vienna, Austria, 19–22 July 2022. [CrossRef]
36. Hu, X.; Qian, X.; Yang, X. Remaining useful life prediction based on a joint model with degradation-failure association structures. *UPB Sci. Bull. Ser. D* **2023**, *85*, 161–180. Available online: https://www.scientificbulletin.upb.ro/rev_docs_arhiva/full002_778083.pdf (accessed on 11 May 2024).
37. EN13306; Maintenance Terminology. *European Standard*; European Committee for Standardization: Brussels, Belgium, 2017.
38. Trojan, F.; Marçal, R.F. Proposal of maintenance-types classification to clarify maintenance concepts in production and operations management. *J. Bus. Econ.* **2017**, *8*, 560–572.
39. Gašpar, G.; Poljak, I.; Orović, J. Computerized Planned Maintenance System Software Models. *Pomorstvo* **2018**, *32*, 141–145. [CrossRef]

40. Stazić, L.; Račić, N.; Stanivuk, T.; Dobrota, Đ. Determination of Benefits of the Application of CMMS Database Improvement Proposals. *Appl. Sci.* **2023**, *13*, 2731. [CrossRef]
41. Wang, C.; Zhou, B.; Li, P.; Li, P. Study on Maintenance Progress FMECA for Ship Equipment. In Proceedings of the 2017 International Conference on Computer Technology, Electronics and Communication (ICCTEC), Dalian, China, 19–21 December 2017. [CrossRef]
42. Knežević, V.; Orović, J.; Stazić, L.; Čulin, J. Fault Tree Analysis and Failure Diagnosis of Marine Diesel Engine Turbocharger System. *J. Mar. Sci. Eng.* **2020**, *8*, 1004. [CrossRef]
43. Eriksen, S.; Utne, I.B.; Lützen, M. An RCM Approach for Assessing Reliability Challenges and Maintenance Needs of Unmanned Cargo Ships. *Reliab. Eng. Syst. Saf.* **2021**, *210*, 107550. [CrossRef]
44. Park, J.; Oh, J. Anomalistic Symptom Judgment Algorithm for Predictive Maintenance of Ship Propulsion Engine Using Machine Learning. *Appl. Sci.* **2023**, *13*, 11818. [CrossRef]
45. Postolache, F. *Programarea Calculatoarelor si Limbaje de Programare Prin Exemple Practice*; Editura Academiei Navale Mircea cel Bătrân Constanţa: Constanta, Romania, 2018.
46. Yu, J. Research Process on Software Development Model. *IOP Conf. Ser. Mater. Sci. Eng.* **2018**, *394*, 032045. [CrossRef]
47. Tinga, T.; Tiddens, W.; Amoiralis, F.; Politis, M. Predictive maintenance of maritime systems: Models and challenges. In *European Safety and Reliability Conference, ESREL 2017*; Taylor & Francis: Oxfordshire, UK, 2017.
48. Scheu, M.N.; Tremps, L.; Smolka, U.; Kolios, A.; Brennan, F. A Systematic Failure Mode Effects and Criticality Analysis for Offshore Wind Turbine Systems towards Integrated Condition Based Maintenance Strategies. *Ocean Eng.* **2019**, *176*, 118–133. [CrossRef]
49. Van Ta, T.; Vu, N.H.; Triet, M.A.; Thien, D.M.; Cang, V.T. Assesment of Marine Propulsion System Reliability Based on Fault Tree Analysis. *Int. J. Transp. Eng. Technol.* **2017**, *2*, 55. [CrossRef]
50. Abaei, M.M.; Hekkenberg, R.; BahooToroody, A. A Multinomial Process Tree for Reliability Assessment of Machinery in Autonomous Ships. *Reliab. Eng. Syst. Saf.* **2021**, *210*, 107484. [CrossRef]
51. Ehiagwina, F.O.; Kehinde, O.O.; Sidiq, N.; Afolabi, L.O.; Olatinwo, I.S. Fault tree analysis and its modifications as tools for reliability and risk analysis Ebgineering systems—An overview. *Int. J. Res. Publ. Rev.* **2022**, *3*, 383–396. Available online: https://ijrpr.com/uploads/V3ISSUE1/IJRPR2327.pdf (accessed on 22 July 2024).
52. Bayraktar, M.; Nuran, M. Reliability, availability and maintainability analysis of the propulsion system of a fleet. *In Zeszyty Naukowe Akademii Morskiej w Szczecinie; Scientific Journals of the Maritime University of Szczecin Poland.* **2022**, *70*, 63–70. Available online: https://repository.am.szczecin.pl/handle/123456789/2710 (accessed on 14 June 2024).
53. Vahed, A.T.; Ghodrati, B.; Hossienie, H. Enhanced K-Nearest Neighbors Method Application in Case of Draglines Reliability Analysis. In Proceedings of the 27th International Symposium on Mine Planning and Equipment Selection—MPES; Springer eBooks: Berlin/Heidelberg, Germany, 2019; pp. 481–488. [CrossRef]
54. Tiwari, A. Supervised Learning: From Theory to Applications. In *Artificial Inteligence and Machine Learning for EDG Computing*; Academic Press: Cambridge, MA, USA, 2022; pp. 23–32. [CrossRef]
55. Sisode, M.; Devare, M. A Review on Machine Learning Techniques for Predictive Maintenance in Industry 4.0. In Proceedings of the International Conference on Applications of Machine Intelligence and Data Analytics (ICAMIDA), Aurangabad, India, 22–24 December 2022; Advances in Computer Science Research. Springer: Berlin/Heidelberg, Germany, 2022; pp. 774–783. [CrossRef]
56. Zhang, P.; Gao, Z.; Cao, L.; Dong, F.; Zou, Y.; Wang, K.; Zhang, Y.; Sun, P. Marine Systems and Equipment Prognostics and Health Management: A Systematic Review from Health Condition Monitoring to Maintenance Strategy. *Machines* **2022**, *10*, 72. [CrossRef]
57. De Aguiar Vallim Filho, A.R.; Moraes, D.F.; De Aguiar Vallim, M.V.B.; Da Silva, L.S.; Da Silva, L.A. A Machine Learning Modeling Framework for Predictive Maintenance Based on Equipment Load Cycle: An Application in a Real World Case. *Energies* **2022**, *15*, 3724. [CrossRef]
58. Traini, E.; Bruno, G.; D'Antonio, G.; Lombardi, F. Machine Learning Framework for Predictive Maintenance in Milling. *IFAC-Pap.* **2019**, *52*, 177–182. [CrossRef]
59. Brown, C. Why and How to Employ the SIPOC Model. *J. Bus. Contin. Emerg. Plan.* **2019**, *12*, 198–210. [CrossRef]
60. Mishra, P.; Sharma, R.K. A Hybrid Framework Based on SIPOC and Six Sigma DMAIC for Improving Process Dimensions in Supply Chain Network. *Int. J. Qual. Reliab. Manag.* **2014**, *31*, 522–546. [CrossRef]