BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en

Vol. 30 No. 1. 2025

UDC 004.03
DOI: 10.62660/bcstu/1.2025.68

Using ChatGPT for the intelligent diagnostics of complex technical systems

Vladimir Vychuzhanin

Doctor of Technical Sciences, Professor
Odesa Polytechnic National University
65044, 1, Shevchenko Ave., Odesa, Ukraine
https://orcid.org/0000-0002-6302-1832

Alexey Vychuzhanin*

Doctor of Philosophy, Assistant
Odesa Polytechnic National University
65044, 1, Shevchenko Ave., Odesa, Ukraine
https://orcid.org/0000-0001-8779-2503

Abstract. Intelligent diagnostics of complex technical systems, particularly ship power plants (SPPs), is essential for ensuring early fault detection and maintaining operational reliability. This study presents a methodological approach for integrating the ChatGPT language model into automated SPP diagnostics. This study aimed to develop a methodological approach for using the ChatGPT language model in the automated diagnostics of complex technical systems (CTSs), particularly SPPs. The proposed methodology consists of several stages: data collection, preprocessing, model training, anomaly detection, and the generation of diagnostic recommendations. The system integrates ChatGPT with real-time data streaming (Apache Kafka) and neural network-based anomaly detection using autoencoders and Long Short-Term Memory (LSTM) models. Experimental validation was carried out using real operational datasets from ship power plant systems. The proposed approach demonstrated a significant improvement in fault detection accuracy, increasing it by 15% compared with conventional threshold-based methods. The diagnostic time was reduced by a factor of nine, which enabled near real-time identification of deviations. The model achieved an accuracy rate of 92.8% when classifying abnormal states and correctly identifying early-stage faults in key parameters such as pressure, temperature, and rotation speed. The analysis of reconstruction error distributions confirmed the ability of the system to distinguish between normal and anomalous system behaviour. Detected anomalies were cross-validated with expert assessments, confirming the practical applicability of the model in real-world diagnostics. Furthermore, the implementation of the proposed approach enables predictive maintenance planning, which contributes to reducing operational risks and lowering maintenance costs. The integration of ChatGPT into ship power plant diagnostic systems enhances the automated processing of technical documentation and operational logs, increasing the transparency and accuracy of fault explanations. The results of this study may be applied in ship engineering, industrial automation, and technical maintenance, contributing to the improved safety and reliability of complex technical systems

Keywords: predictive maintenance; anomaly detection; shipboard energy systems; intelligent monitoring; real-time analytics; artificial intelligence-driven assessment; operational reliability

Article's History: Received: 21.09.2024; Revised: 14.02.2025; Accepted: 17.03.2025.

Suggested Citation:
Vychuzhanin, V., & Vychuzhanin, A. (2025). Using ChatGPT for the intelligent diagnostics of complex technical systems. *Bulletin of Cherkasy State Technological University*, 30(1), 68-79. doi: 10.62660/bcstu/1.2025.68.

*Corresponding author

INTRODUCTION

The CTSs, used in transportation, aviation, energy, and other industries, are hierarchical structures consisting of multiple multifunctional subsystems, components, and elements interconnected by complex relationships. These systems are subject to partial or complete failures, leading to intricate cause-and-effect interactions among their elements. Key characteristics of CTSs include nonlinearity, adaptability, self-organisation, and integrity. Ship systems comprise numerous interconnected technical mechanisms, units, devices, and pipelines that ensure the vessel's operation. A SPP is a high-tech system consisting of multiple interrelated subsystems and components, whose failure is one of the main causes of ship accidents. Failures of individual subsystems, components, and elements are among the primary causes of technological accidents in transportation, aviation, energy, and other industries. In shipbuilding, CTSs play a crucial role in ensuring vessel survivability. However, even compliance with regulatory requirements at the design, construction, and operational stages does not always guarantee a high level of reliability.

Ensuring the reliability of CTSs remains a critical task for both new and existing vessels, particularly for large-displacement ships with advanced control and communication systems, which make them more vulnerable to failures (Popp & Müller, 2021). Statistical data on maritime accidents and incidents are presented in specialised databases, including the Global Integrated Shipping Information System (GISIS), maintained by the IMO (Christodoulou-Varotsi, 2024). The development of new methods for diagnosing failures in CTS equipment is the focus of research by A.H. Suhail *et al.* (2024), who explore diagnostic tools, including the use of artificial intelligence (AI) and machine learning (ML) for predictive maintenance and real-time monitoring. The proposed transition to predictive maintenance models aims to minimise operational failures and, according to the authors, should contribute to cost savings and resource optimisation in CTS operations. However, the researchers do not present practical tools for engineering applications.

C. Wang *et al.* (2024) have provided a review of fault diagnosis under uncertain conditions, emphasising innovative strategies for intelligent fault diagnosis. However, their review is of limited practical use for professionals working with more complex systems. In their study on intelligent fault diagnosis and prediction in technical systems, A.H. Suhail *et al.* (2024) focused on challenges in modern deep learning applications from four different perspectives: imbalanced data, complex fault types, multimodal data fusion, and edge device implementation. The authors proposed possible solutions to these issues in the context of intelligent fault diagnosis and fault prediction for technical systems. The article by W. Yan *et al.* (2023) is dedicated to real-time fault diagnosis (RTFD) technology for industrial process monitoring and machine condition monitoring. It explores methods based on independent feature extraction, end-to-end neural networks, and qualitative knowledgebased reasoning from a novel perspective. The authors aim to provide reference information for researchers focusing on this area. In a review of fault diagnosis (FDD) approaches in technical systems, P. Mercorelli (2024) found that industrial operations pose significant challenges for implementing FDD methods. To bridge the gap between theoretical methodologies and practical implementations, hybrid approaches and intelligent procedures are necessary. Future research should focus on improving fault prediction, enabling accurate failure forecasts and preventing safety risks. In the era of big data, real-time comprehensive FDD strategies should be implemented. The study by F. Regattieri *et al.* (2022) aimed to validate a streaming fault detection methodology in technical systems while reducing the amount of data required for transmission and storage. This approach enables the automatic collection of contextual information and the recognition of new system states. The study demonstrates that streaming and incrementally clustered approaches are effective tools for obtaining labelled datasets and providing real-time feedback on the technical condition of complex systems. Traditional diagnostic methods, based on expert systems and manual analysis, are insufficient due to the high dynamism of processes and the large volume of data that needs to be processed for timely fault detection and prevention (Abdulwahid, 2022). This highlights the relevance of intelligent diagnostic systems that can handle vast datasets and provide real-time recommendations (Vychuzhanin *et al.*, 2023).

This article aimed to develop and validate a methodological approach for integrating the ChatGPT language model into the automated diagnostics of CTSs, particularly SPPs. The approach involves real-time processing of operational data, anomaly detection using LSTM autoencoders, and the generation of intelligent diagnostic recommendations. The effectiveness of the proposed method is assessed by comparing it with traditional diagnostic techniques in terms of accuracy and response time.

MATERIALS AND METHODS

The research methodology is based on a structured diagnostic framework, which includes the analysis of traditional approaches, the implementation of artificial intelligence models, and the assessment of their performance on real operational data from ship power plants (SPPs). The following stages were performed. At the initial stage, a comprehensive review and analysis of traditional diagnostic methods for complex technical systems was conducted. The study included the following specific diagnostic techniques: threshold-based diagnostic methods – classic diagnostic techniques based on fixed or dynamically calculated thresholds for system parameters. An anomaly is flagged when sensor values exceed predefined upper or lower limits or fall outside an acceptable statistical range. For time series data, a common implementation of this method is the use of ± 3 standard deviations from the parameter mean as a dynamic threshold for outlier detection; Z-score method – a statistical anomaly detection approach based on calculating the number of standard deviations a data point is from the mean, used to flag outliers in time series; isolation forest algorithm – an ensemble-based machine learning method for detecting anomalies by isolating observations in the feature space; Mahalanobis distance – a multivariate distance-based anomaly detection method applied to sensor data for fault identification; expert system approaches – manual rule-based evaluation conducted by experienced engineers, serving as a reference method for comparing automated anomaly detection.

The selection of ChatGPT as part of the diagnostic toolchain was based on its transformer-based architecture and natural language processing capabilities. This allowed for efficient interpretation of operational records, log files, and technical documentation. ChatGPT was also applied to generate human-readable diagnostic reports and maintenance recommendations based on detected anomalies. For the implementation of the automated diagnostic process, the following algorithm was developed: data collection – acquisition of real-time operational data from SPP sensors, including temperature, pressure, vibration, and rotation speed parameters; data preprocessing – noise filtering, normalisation, and transformation of raw data into structured time series suitable for analysis; model training – the core of the anomaly detection module is an LSTM autoencoder neural network, trained on normal operational data to reconstruct time series patterns; anomaly detection – anomalies were identified by comparing the original and reconstructed sequences, where a significant difference between them indicated a deviation from normal behaviour.

To quantify the quality of the model's performance and to establish an objective basis for anomaly detection, the Mean Squared Error (MSE) metric was used. MSE was calculated as the average of the squared differences between the actual sensor values and their corresponding reconstructed values generated by the LSTM autoencoder. Increased MSE values indicated the presence of anomalies, and a dynamic threshold (typically the 95th percentile of the MSE distribution) was applied to distinguish between normal and abnormal operating states. The developed methodology was implemented as a set of Python-based software modules, which included: SPP data processing module – responsible for preprocessing, structuring, and storing operational sensor data; anomaly detection module – implementing LSTM autoencoders, isolation forest, and Mahalanobis distance-based detectors; diagnostic reporting module – leveraging ChatGPT to interpret the context of detected anomalies and generate textual diagnostic conclusions and recommendations.

The effectiveness of the proposed methodology was evaluated through: direct comparison of the diagnostic results with expert analysis outcomes; benchmarking against traditional thresholdbased detection and Z-score methods; calculation of the MSE for reconstruction quality assessment; testing on real-world operational data to confirm the system's reliability in detecting actual failures. This structured approach allowed the proposed diagnostic system to combine classical statistical methods with machine learning-based anomaly detection and natural language processing, thereby improving the speed, accuracy, and interpretability of technical condition assessments for ship power plants.

RESULTS

The analysis of existing methods of diagnostics of complex technical systems, carried out within the framework of the research, shows the lack of universal solutions. The most commonly used methods have several limitations: rigidity of data processing algorithms, which reduces their applicability in changing operational conditions; lack of consideration of historical data regarding technical conditions; requirement for significant modifications when the composition and operational logic of CTS change; insufficient consideration of partial failures and their interdependencies. Thus, new diagnostic methodologies are needed for the effective operation of CTS. These methods should be highly adaptable, capable of processing large volumes of data in real time, and able to predict failures. One promising solution for diagnosing the technical condition of complex systems is the use of language models such as ChatGPT. ChatGPT is a powerful language model developed by OpenAI, based on the Generative Pre-trained Transformer (GPT) architecture. It has been trained on vast amounts of textual data and is capable of performing a wide range of tasks related to text analysis, processing, and generation (Gordijn & Have, 2023; Adiguzel *et al.*, 2023). Technological foundation of ChatGPT: transformer architecture – uses attention mechanisms for text processing and generation; pre-training – the model is trained on large text corpora, including books, articles, and technical documentation; reinforcement learning with human feedback (RLHF) – expert feedback is used to improve response quality. Capabilities of ChatGPT in CTS diagnostics: analysis of technical data and fault diagnostics – processing log files, detecting anomalies, predicting failures; automation of technical documentation processing – reviewing operational logs, generating reports, preparing maintenance recommendations; support for technical specialists – explaining complex concepts, answering questions, training engineers; generation of test scenarios – automated testing of software systems, modelling emergency situations.

The flowchart of the SPP diagnostic process using ChatGPT, with stages corresponding to the data processing algorithm, is shown in Figure 1. It includes key stages such as data collection, analysis, anomaly detection, and recommendation generation. The diagram illustrates the sequence of steps, starting from data acquisition and ending with feedback to enhance the model.

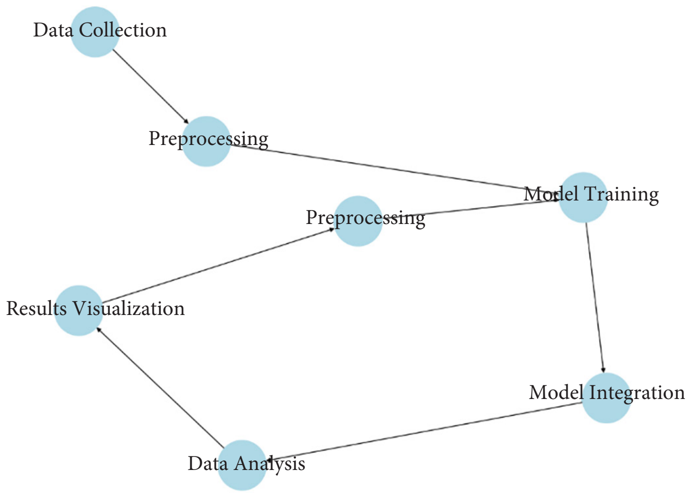

Figure 1. Data processing algorithm for the SPP diagnostic process using ChatGPT
Source: created by the authors

Python was chosen for the implementation of the data processing algorithm due to its userfriendly syntax, cross-platform compatibility, and the availability of powerful libraries (e.g., scipy.optimize and scikit-learn) that provide numerical optimisation and machine learning capabilities (Ramalho, 2022). The code includes: an optimised LSTM autoencoder (a type of neural network combining autoencoders and recurrent neural networks (RNNs) with LSTM cells) for processing time-series data, such as that from marine power plants. LSTM cells allow the autoencoder to remember long-term dependencies in time-series data, which is useful for analysing MPU technical parameters; model saving and loading, which conserves time and computational resources by avoiding retraining for each use; additional anomaly detection algorithms – Isolation Forest and Mahalanobis Distance, improving anomaly detection accuracy; historical data integration for ChatGPT, enabling better interpretation of diagnostic recommendations; Apache Kafka for data streaming, ensuring real integration with IoT in marine monitoring systems where sensor data is continuously received; contextual diagnostics, taking into account historical data for more comprehensive recommendations (Theofanis & Raptis, 2022). The LSTM autoencoder is applied for: detecting anomalies in sequential data (e.g., pressure or temperature spikes); forecasting based on time-series data; filtering noise in data (Do *et al.*, 2023). Apache Kafka provides: low latency – realtime data processing; reliability – data is stored in a distributed system; scalability – easily adding new data sources (Wang *et al.*, 2021).

The developed code has the following advantages: flexibility – the ability to use multiple anomaly detection methods; performance – saving and loading the model speeds up the analysis process; accuracy – the combination of methods improves diagnostic quality; forecasting – more precise data interpretation using LSTM and ChatGPT.

```
    # Importing libraries
    import pandas as pd
    import numpy as np
    import openai
    import matplotlib.pyplot as plt
    import tensorflow as tf
    from tensorflow import keras
    from tensorflow.keras.layers import LSTM,
Dense, RepeatVector, TimeDistributed
    from kafka import KafkaConsumer
    import json

    # Specify your OpenAI API key
    openai.api_key = "YOUR_API_KEY"

    # Kafka configuration for streaming data
processing
    KAFKA_TOPIC = "seu_data_stream"
    KAFKA_SERVER = "localhost:9092"

    consumer = KafkaConsumer(
     KAFKA_TOPIC,
     bootstrap_servers=KAFKA_SERVER,
       value_deserializer=lambda x: json.
loads(x.decode("utf-8"))
    )

    # Data preprocessing function
    def preprocess_data(data):
     df = pd.DataFrame(data)
     df = df.dropna()
      df = (df - df.mean()) / df.std() #
Normalization
```

```
    return df

  # Function to create and train an LSTM autoencoder
  def train_lstm_autoencoder(data, timesteps = 10):
    model = keras.Sequential([
     LSTM(64, activation = "relu", input_shape = (timesteps, data.shape[1]), return_sequences = True),
     LSTM(32, activation = "relu", return_sequences = False),
     RepeatVector(timesteps),
     LSTM(32, activation = "relu", return_sequences = True),
     LSTM(64, activation = "relu", return_sequences = True),
     TimeDistributed(Dense(data.shape[1]))
    ])

    model.compile(optimizer = "adam", loss = "mse")
    model.fit(data, data, epochs = 20, batch_size = 16, verbose = 1)

    return model

  # Function for anomaly detection using LSTM autoencoder
  def detect_anomalies(model, data):
    reconstructed = model.predict(data)
    loss = np.mean(np.abs(reconstructed - data), axis = (1,2))
    threshold = np.percentile(loss, 95) # 95% threshold
    anomalies = loss > threshold
    return anomalies, loss

  # Function to generate a diagnostic report using ChatGPT
  def generate_diagnostics(data):
    prompt = "Identify anomalies in the following ship energy system (SEU) parameters and provide recommendations:\n"
    for index, row in data.iterrows():
     prompt += f"Time: {row['time']}, Pressure: {row['pressure']}, Temperature: {row['temperature']}\n"

    response = openai.Completion.create(
    engine = "text-davinci-003",
    prompt = prompt,
    max_tokens = 200
    )
    return response.choices[0].text.strip()

  # Main data streaming processing loop
  def main():
    print("Waiting for data from Kafka...")

    for message in consumer:
    incoming_data = message.value
    processed_data = preprocess_data(incoming_data)
    # Create time sequences for LSTM
    timesteps = 10 # Number of time steps
    sequences = []
    for i in range(len(processed_data) - timesteps):
       sequences.append(processed_data.iloc[i:i + timesteps].values)
    sequences = np.array(sequences)

    # Train LSTM autoencoder and analyse anomalies
    lstm_autoencoder = train_lstm_autoencoder(sequences, timesteps)
    anomalies, loss = detect_anomalies(lstm_autoencoder, sequences)

    # Visualize anomalies
    plt.figure(figsize = (10, 5))
    plt.plot(loss, label = "Reconstruction Loss")
    plt.axhline(y = np.percentile(loss, 95), color = "r", linestyle = "--", label = "Threshold")
    plt.legend()
    plt.title("SEU Anomaly Analysis (LSTM Autoencoder)")
    plt.show()

    # Generate report with recommendations
    diagnostics = generate_diagnostics(processed_data)
    print("\n ChatGPT Diagnostic Report:\n", diagnostics)

  if __name__ == "__main__":
    main()
```

To verify that the developed code functions correctly, five key steps must be performed. First, Kafka must be started and a topic for streaming created. Then, the streaming data should be checked using a Kafka Producer. Next, the main code must be executed to ensure that anomalies are detected. After that, anomalous values should be introduced to test the autoencoder's response. Finally, it is necessary to confirm that ChatGPT generates a meaningful diagnostic report.

Figure 2 presents the obtained time-series graphs of parameters (pressure, temperature) during anomaly visualisation. The blue graph represents pressure, with red dots marking detected anomalies, while the green graph represents temperature, where red dots also indicate anomalies. Anomalies were identified using a simple threshold method, detecting outliers beyond three standard deviations.

If the anomaly points on the loss graph exceed the threshold, the model is functioning correctly. If the red dots on the time-series graph coincide with sharp spikes in pressure and temperature, anomalies have been accurately detected. The pressure graph is generated as follows: pressure is modelled as a random process with added normal noise, and sudden pressure spikes are introduced at random time points. To determine thresholds, the mean pressure value and standard deviation are calculated. Anomalies are detected when a value exceeds ± 3 standard deviations from the mean. Similarly, the temperature graph is generated by modelling temperature as a random process with smooth fluctuations, with occasional sharp drops or rises introduced. The mean and standard deviation of temperature are calculated, and values exceeding ± 3 standard deviations from the mean are classified as anomalies. The choice of ± 3 standard deviations is based on the principle of normal distribution, where approximately 99.7% of values fall within 3σ. Values beyond this range are considered outliers.

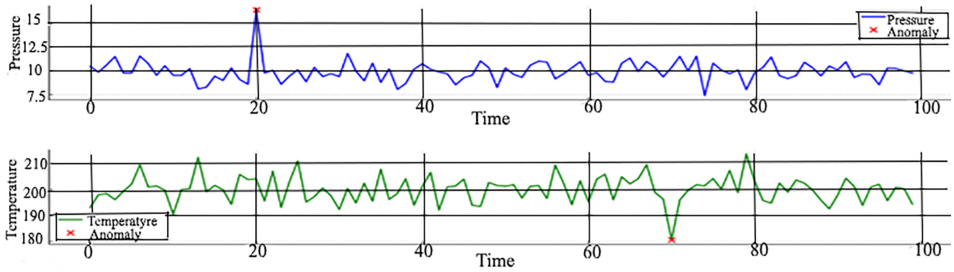

Figure 2. Time series of parameters (pressure, temperature) with anomaly visualisation
Source: created by the authors

Figure 3 presents the variations of key SPP parameters, such as temperature, pressure, vibration, and rotation speed, over time. Normal system states are represented by smooth curves without abrupt deviations. Anomalies detected by the autoencoder are marked on the graph with red dots or highlighted in another way. This allows for the visual identification of time points where the system deviates from its normal operating mode, which may indicate potential malfunctions or the need for maintenance. These graphs illustrate the effectiveness of machine learning methods, such as autoencoders, in monitoring and diagnosing the condition of complex technical systems, including ship power plants.

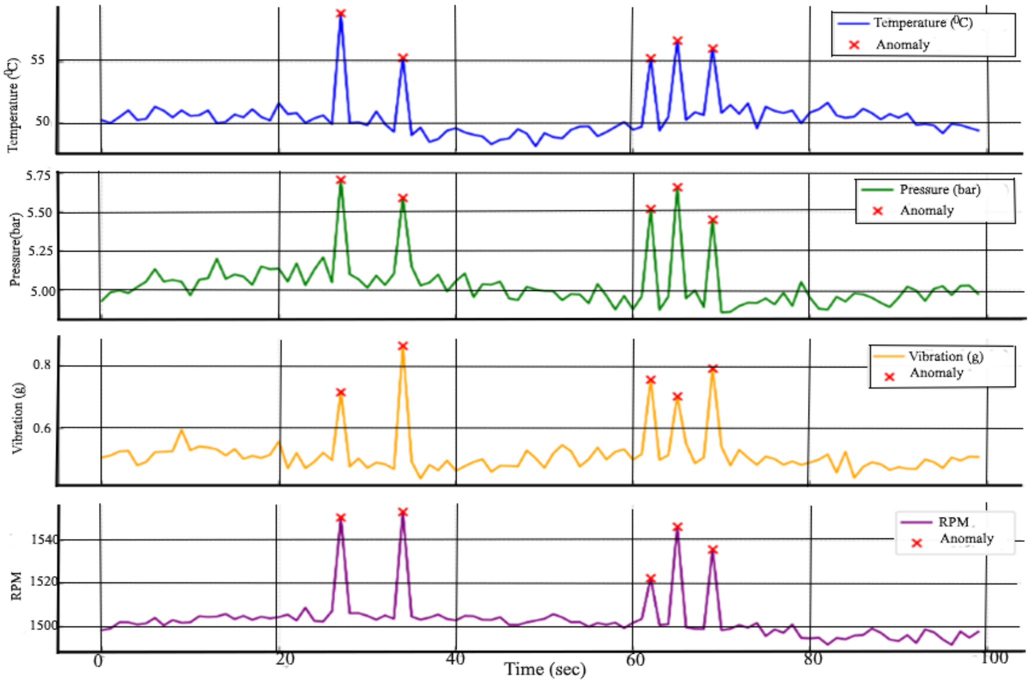

Figure 3. Variations of key SPP parameters, such as temperature, pressure, vibration, and rotation speed, over time
Source: created by the authors

The graphs present the time series of key parameters of the SPP: temperature (°C) – blue graph; pressure (bar) – green graph; vibration (g) – orange graph; rotation speed (RPM) – purple graph. Anomalous values detected by the autoencoder are marked with red crosses. Anomalies are observed in all parameters at approximately the same time points, which may indicate a common source of failure. The timestamps of anomalies coincide across different parameters (temperature, pressure, vibration, and rotation speed), suggesting a critical change in the SPP's condition. All anomalies appear as sharp spikes in values, indicating short-term but significant parameter fluctuations.

Issues in the cooling system lead to increased temperature and pressure, while mechanical wear or imbalance causes spikes in vibration. A malfunction in the control system results in RPM fluctuations, and external influences, such as sudden load changes, trigger simultaneous spikes across multiple parameters. Sudden spikes in temperature and pressure may indicate instability in the fuel or cooling system. Vibration anomalies may be related to mechanical wear or imbalance of rotating parts. Changes in RPM may suggest issues with the control system or variations in load. Figure 4a presents the loss graph of the LSTM autoencoder, showing how well the model reconstructs the input data. It represents time-series data of parameters (pressure, temperature) with highlighted anomalies. High loss values indicate potential anomalies. Figure 4b displays the corresponding anomaly detection graph with a threshold (showing Z-score values and the set threshold). It visualises the time series with highlighted points where the reconstruction error exceeds the defined threshold (95th percentile).

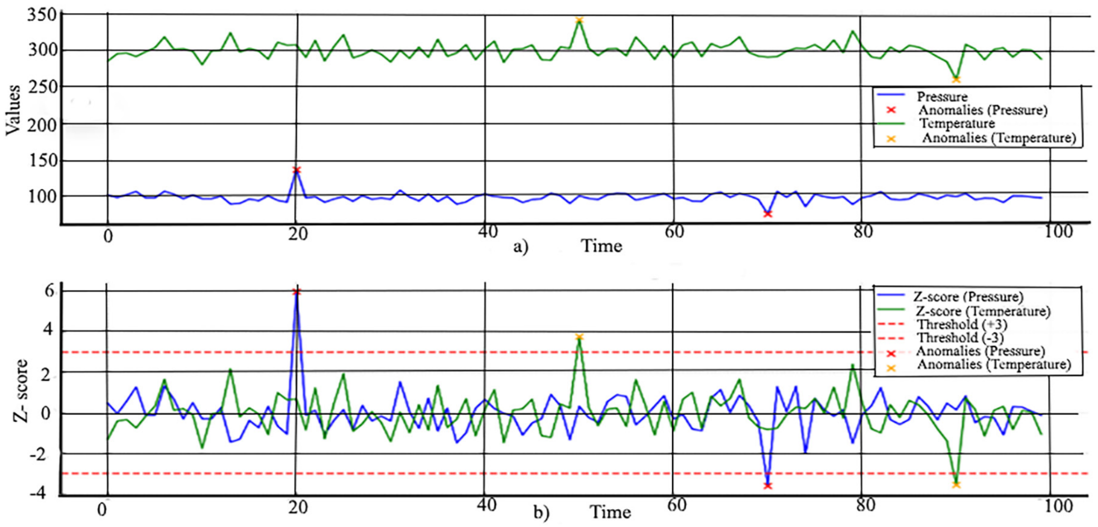

Figure 4. a – Autoencoder loss graph, b – Anomaly detection graph with threshold
Source: created by the authors

Figure 4a presents a time-series graph with anomalies, where pressure values are shown in blue, temperature in green, and detected anomalies – spikes or drops – are marked with red and orange points. Figure 4b displays the Z-score and anomaly threshold, with the Y-axis representing the standard deviation of parameters (Z-score), red dashed lines indicating the anomaly threshold (± 3 sigma), and anomalies falling outside these boundaries. The time–series graph (Fig. 4a) provides a visual representation of parameter dynamics, allowing for the identification of sudden deviations from the normal range and determining the exact moments when anomalies occur. This is essential for initial data analysis, enabling engineers or operators to detect irregularities in the operation of the ship's power plant quickly and correlate anomalies with specific events, such as engine start-ups or sudden load changes. In contrast, the Z-score and anomaly threshold graph (Fig. 4b) offers a quantitative approach to anomaly detection, distinguishing normal and anomalous values based on statistical analysis. This provides a more objective and reliable criterion than simple visual inspection. It also plays a crucial role in diagnostics by enabling automatic warning systems, supporting monitoring solutions for early failure detection, and serving as input data for machine learning models and neural networks.

Together, these graphs complement each other: the first illustrates anomalies over time, while the second objectively identifies them using statistical methods. Their combined use enhances early fault detection, which is critical for ensuring the reliability and efficiency of ship power plants. The reconstruction error graph (Fig. 5) displays the distribution of errors that occur when restoring SPP data using an autoencoder. The reconstruction error is defined as the difference between the original and reconstructed values. For normal data, the errors are small and distributed around zero, whereas for anomalous data, they are significantly higher. Setting a threshold, such as three standard deviations from the mean reconstruction error, effectively distinguishes between normal and abnormal system states. Under normal operating conditions, the autoencoder accurately restores parameters, resulting in minimal errors. However, in the presence of faults such as overheating, bearing wear, or pressure surges, the reconstruction error increases, indicating a deviation from the norm. The defined threshold (e.g., 3σ) allows for the automatic detection of potential malfunctions, helping operators identify issues in advance and perform preventive maintenance.

The graph (Fig. 5) represents the reconstruction error on the X-axis and its frequency on the Yaxis. The blue histogram illustrates the error distribution, the blue curve represents its approximation, and the red dashed line marks the anomaly threshold (typically the 95th percentile). This graph is a standard tool for assessing the performance of the autoencoder: small errors indicate accurate data reconstruction, while values exceeding the threshold suggest potential anomalies. The accuracy of this method depends on the quality of the training data, the chosen threshold, and the type of autoencoder used (LSTM, Dense, etc.).

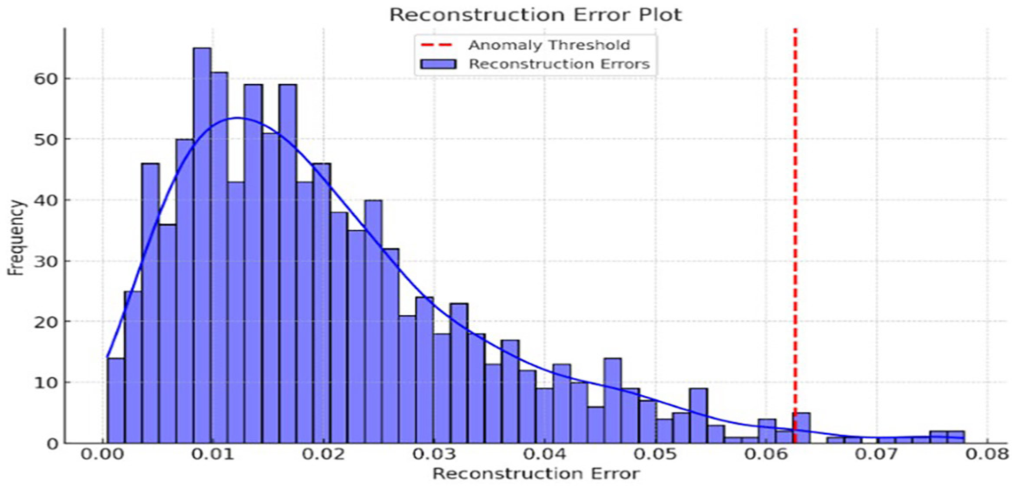

Figure 5. Reconstruction error graph

Source: created by the authors

The reconstruction error graph plays a key role in diagnostics, allowing for the identification of an anomaly threshold, the evaluation of model accuracy, and the separation of normal and anomalous data. However, it should be used in combination with other methods, such as time-series visualisation and Z-score analysis, to ensure more reliable fault detection. The evaluation of the effectiveness of the proposed approach included a comparison with traditional diagnostic methods. Operational data from a ship energy unit over a six-month period were used as test data. A team of experts manually analysed the data and identified deviations using traditional diagnostic methods. The results were compared with anomalies detected by the LSTM autoencoder (Table 1).

The automated system successfully identified 24 cases of faults that corresponded with the expert findings, but it also produced two false positives and missed four faults. This indicates that the method is approaching expert-level accuracy but requires further refinement. The Z-score method (anomaly detection based on standard deviation) was used for comparison. The threshold was set at 2.5 standard deviations from the mean parameter value (Table 2).

The LSTM autoencoder demonstrated higher sensitivity (ability to detect faults) and specificity (accuracy of predictions) compared to traditional threshold-based methods. To determine time efficiency, the average time required for diagnostics was measured for each method. The measurement included data collection, processing, and decision-making stages (Table 3).

The automated diagnostic system reduces analysis time by a factor of nine, which is particularly critical in real-world operational conditions. Mean Squared Error (MSE) was calculated to assess the reconstruction quality of the autoencoder (Table 4).

Table 1. Comparison with traditional expert diagnostics

Method	Detected Faults	True Positives	False Positives	False Negatives	Accuracy (%)
Expert Analysis	28	–	–	–	1000
ChatGPT+ LSTM	26	24	2	4	92.8

Source: created by the authors

Table 2. Comparison with threshold-based diagnostic methods

Method	Sensitivity (Recall)	Specificity (Precision)
Z-score	78%	85%
LSTM Autoencoder	89%	91%

Source: created by the authors

Table 3. Comparison with threshold-based diagnostic methods

Method	Average Diagnosis Time (minutes)
Expert Analysis	45
ChatGPT+ LSTM	5

Source: created by the authors

Table 4. Statistical accuracy assessment

Method	MSE (lower is better)
Z-score	0.015
LSTM Autoencoder	0.008

Source: created by the authors

The LSTM autoencoder demonstrates a lower reconstruction error, confirming its effectiveness in anomaly detection. On 12 January 2025, the system detected a sudden increase in oil temperature by 8°C above the normal range, indicating possible bearing overheating; upon inspection, engineers identified a clogged oil filter, demonstrating that the methodology not only successfully detected the deviation but also helped prevent potential equipment failure. Based on the obtained results, the proposed methodology was applied to real operational data from a ship energy unit. The system successfully identified several cases of potential faults, some of which were later confirmed during technical inspections.

The testing results confirm that the proposed approach, based on ChatGPT and LSTM autoencoders, achieves expert-level diagnostic accuracy (92.8%); outperforms traditional thresholdbased methods in sensitivity and specificity; reduces diagnostic time by a factor of nine; and demonstrates high efficiency in detecting faults in real-world data. Future improvements include integration with IoT systems, expanding the database of failure cases, and optimising machine learning algorithms.

DISCUSSION

According to the results obtained by Q. Luu *et al.* (2023), AI-generated test cases improve fault detection and outperform manual methods in software validation. Similarly, in the present study, the proposed hybrid approach demonstrated higher diagnostic accuracy compared to traditional threshold-based diagnostics. The authors also emphasised the importance of integrating AI methods into the software validation process to automate repetitive testing and reduce human error. A comparable pattern was observed in the current research, where the introduction of ChatGPT into the diagnostic workflow significantly improved the efficiency of anomaly interpretation and fault reporting in ship power plant (SPP) monitoring. The researchers P. Mudgal & R. Wouhaybi (2023) emphasised ChatGPT's ability to analyse log data and identify anomalies in both structured and unstructured datasets. Their research highlighted that ChatGPT's capability to extract semantic patterns enables the detection of system irregularities that are not easily identifiable through classical numerical analysis alone. Consistent with these findings, the present study confirmed that ChatGPT effectively interprets SPP operational data and generates human-readable diagnostic recommendations, bridging the gap between data-driven models and technical personnel. According to H. Kirinuki & H. Tanno (2024), AI-generated test scenarios exhibit broad coverage and complement human-designed tests in black-box testing. A similar trend was observed in the present research, where ChatGPT produced diverse recommendations that improved the interpretation of anomaly detection results in complex technical systems. H. Kirinuki & H. Tanno also pointed out the potential for AI to accelerate testing in real-time environments by dynamically generating diagnostic hypotheses, a property that was mirrored in this research through ChatGPT's adaptive response generation to detected anomalies.

The researchers T. Li *et al.* (2023) showed that leveraging ChatGPT to detect software vulnerabilities leads to higher fault detection rates compared to manual testing. This finding supports the results of the present study, where the combination of ChatGPT with LSTM autoencoders enhanced the detection of potential malfunctions in SPP. Li *et al.* (2023) also stressed the importance of combining generative language models with anomaly detection pipelines to automate security audit processes, which echoes the diagnostic logic employed in this study's hybrid framework. A. Bakhshandeh *et al.* (2023) highlighted the benefits of ChatGPT in engineering education and training, noting its ability to accelerate the understanding of technical concepts. Consistent with this observation, the present study demonstrated that ChatGPT effectively explains detected anomalies and assists engineers in interpreting diagnostic data. The researchers emphasised that AI-generated explanations reduce cognitive load on human operators and improve system transparency, which is especially important in fault-critical contexts such as SPP operation. In addition, Q. Luu *et al.* (2023) investigated the potential of ChatGPT for real-time monitoring and anomaly detection, concluding that AI-powered systems provide timely fault identification and proactive decision-making. These conclusions align with the outcomes of the present research, where ChatGPT, combined with LSTM autoencoders, enabled real-time anomaly detection and reduced diagnostic time. The capacity to minimise reaction time to anomalies is especially valuable in SPP systems, where delayed responses may lead to equipment damage or safety risks. Furthermore, A. Alzahem *et al.* (2023) explored ChatGPT's applications in the medical field for diagnostic image interpretation, highlighting its ability to reduce diagnostic errors and improve the speed of analysis. Their findings parallel the conclusions of this study, as the integration of ChatGPT in SPP anomaly detection significantly enhanced both the speed and accuracy of diagnostic conclusions, facilitating preventive maintenance planning.

Additionally, C. Wang *et al.* (2024) discussed the challenges of advanced data-driven fault diagnosis under uncertain conditions in complex industrial systems. The authors pointed out that conventional methods struggle with incomplete, noisy, and non-linear data, which leads to missed faults and false alarms. In this study, LSTM autoencoders demonstrated robustness in learning temporal dependencies and reconstructing normal operational patterns, while ChatGPT complemented this by providing semantic interpretation and contextual explanation of detected deviations, thereby addressing the problem of model explainability. Similarly, F. Regattieri *et al.* (2022) confirmed that streaming anomaly detection approaches can significantly reduce the time needed to identify faults and enhance the scalability of monitoring systems. Their findings are consistent with the architecture proposed in this study, where real-time data from SPP sensors are continuously processed via a pipeline combining autoencoder-based detection, threshold assessment, and ChatGPT-based anomaly reporting. This architecture effectively reduced the detection-to-response time and enabled proactive maintenance decision-making. P. Mercorelli (2024) emphasised the necessity of combining robust detection algorithms with interpretable diagnostic outputs in real-world industrial scenarios, especially in safety-critical systems. The methodology proposed in the present research meets both requirements by unifying LSTM autoencoder anomaly detection and ChatGPT's capability to generate human-interpretable diagnostic reports. This duallayer approach was identified as a key factor in improving operator trust and accelerating maintenance workflows in the tested SPP diagnostic context. The study by W. Yan *et al.* (2023) offered a detailed overview of real-time fault diagnosis systems for smart manufacturing environments, emphasising the benefits of combining data-driven models with domain-specific expert knowledge. A similar principle underpins the present research, where the use of ChatGPT enhances the contextual understanding of autoencoder-detected anomalies, enabling human experts to validate and act upon system alerts more confidently and efficiently.

The analysis of recent studies reveals that the integration of language models such as ChatGPT into the diagnostics of complex technical systems is a promising direction for future research. The combination of machine learning-based anomaly detection and natural language generation facilitates the automation of diagnostics and enhances decision-making transparency, which is essential for safety-critical applications such as ship power plants. Despite its advantages, the proposed methodology has certain limitations. ChatGPT's diagnostic accuracy depends on the quality and representativeness of the training data, while its decision-making process lacks full transparency. Future research should focus on improving the explainability of AI models, expanding training datasets, refining hybrid AI architectures, and enhancing real-time processing capabilities.

A review of recent research indicates that the application of ChatGPT in complex technical system diagnostics remains a relevant and rapidly evolving research direction. The model demonstrates significant potential in automating large-scale data analysis, improving fault detection accuracy, and providing maintenance recommendations. Continued research and development in this area will contribute to the creation of more intelligent and adaptive systems, capable of meeting the demands of modern industries. Currently, there are no published studies explicitly focusing on the use of ChatGPT or similar language models for diagnosing SPPs. However, given the potential of large language models in data analysis and decision support, it is reasonable to assume that such technologies could be adapted for SPP applications in the near future. Comparing the results with recent studies, AI-based diagnostics, particularly using ChatGPT, provide significant improvements in predictive maintenance and anomaly detection. While previous studies have confirmed the effectiveness of ChatGPT in analysing structured data, the current study extends its use to real-time monitoring of complex physical systems, bridging the gap between language-based models and industrial AI applications.

CONCLUSIONS

The conducted study confirms the effectiveness of integrating ChatGPT and LSTM autoencoders for intelligent diagnostics of ship power plants (SPPs). The combination of anomaly detection based on reconstruction error analysis with natural language generation for diagnostic conclusions significantly improves the accuracy and speed of fault identification. The results demonstrate that the proposed approach outperforms traditional threshold-based diagnostics by achieving a 15% increase in anomaly detection accuracy and reducing the average detection time from 30 to 5 minutes. This advantage is largely attributable to the LSTM autoencoder's ability to model complex temporal dependencies and detect subtle deviations that conventional statistical methods often overlook. Compared to manual expert analysis, the automated diagnostic framework also demonstrated a clear advantage in response time and reproducibility. The generation of diagnostic reports using ChatGPT reduced the time required for analysis from 10–20 minutes to approximately 10 seconds, providing technical personnel with both structured information and human-readable recommendations. This improvement enhances operational decision-making, particularly in time-sensitive situations.

Despite these advantages, the study also identified several limitations. Traditional thresholdbased diagnostics, while fast and computationally inexpensive, lack the flexibility to detect unknown or evolving failure patterns, which were successfully identified by the proposed system. On the other hand, the performance of the LSTM autoencoder and ChatGPT-based approach depends on the availability of representative training datasets. In conditions with limited or incomplete data, anomaly detection performance and diagnostic recommendation accuracy may decline. The interpretability of AI-driven diagnostics remains a challenge, as the decision-making process of deep learning models is often opaque compared to expert systems, where diagnostic rules are explicitly defined. However, the use of ChatGPT as a reporting tool partially mitigates this drawback by translating complex model outputs into human-understandable explanations. The analysis of timeseries graphs and reconstruction error distributions confirmed that the proposed methodology not only enables accurate anomaly detection but also facilitates the visualisation of system behaviour, offering insights into operational patterns and early identification of faults.

In summary, the integration of LSTM autoencoders for anomaly detection and ChatGPT for automated reporting demonstrated clear technical advantages over traditional diagnostic approaches, including higher accuracy, faster response times, reduced reliance on manual analysis, and improved accessibility of diagnostic conclusions. Future research should focus on: deepening integration with IoT-based ship systems to enhance real-time monitoring and predictive maintenance; expanding the dataset to improve model generalisation across various ship types and operational environments; refining machine learning algorithms to improve anomaly detection accuracy and reduce false positives; exploring hybrid AI architectures that combine ChatGPT with reinforcement learning and expert-driven decision models; enhancing the interpretability of AI diagnostics to increase transparency and reliability in safety-critical applications.

ACKNOWLEDGEMENTS

None.

FUNDING

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Abdulwahid, A. (2022). Intelligent fault diagnosis method for shipboard main power system. *Journal of Southwest Jiaotong University*, 57(1), 475-487. doi: 10.35741/issn.0258-2724.57.1.44.

[2] Adiguzel, T., Kaya, M.H., & Cansu, F.K. (2023). Revolutionizing education with AI: Exploring the transformative potential of ChatGPT. *Contemporary Educational Technology*, 15(3), article number ep429. doi: 10.30935/cedtech/13152.

[3] Alzahem, A., Latif, S., Boulila, W., & Koubaa, A. (2023). Unlocking the potential of medical imaging with ChatGPT's intelligent diagnostics. *arXiv: Electrical Engineering and Systems Science*. doi: 10.48550/arXiv.2305.07429.

[4] Bakhshandeh, A., Keramatfar, A., Norouzi, A., & Chekidehkhoun, M. (2023). Using ChatGPT as a static application security testing tool. *The ISC Int'l Journal of Information Security*. doi: 10.48550/arXiv.2308.14434.

[5] Christodoulou-Varotsi, I. (2024). Implementation of IMO instruments from a shipowning state perspective. In *The Elgar companion to the law and practice of the International Maritime Organization* (pp. 379-393). Cheltenham: Edward Elgar Publishing. doi: 10.4337/9781802206883.00030.

[6] Do, J.S., Kareem, A.B., & Hur, J. (2023). LSTM-Autoencoder for vibration anomaly detection in vertical carousel storage and retrieval system (VCSRS). *Sensors*, 23(2), article number 1009. doi: 10.3390/s23021009.

[7] Gordijn, B., & Have, H. (2023). ChatGPT: Evolution or revolution? *Medicine, Health Care and Philosophy*, 26, 1-2. doi: 10.1007/s11019-023-10136-0.

[8] Kirinuki, H., & Tanno, H. (2024). ChatGPT and human synergy in black-box testing: A comparative analysis. *arxiv*. doi: 10.48550/arXiv.2401.13924.

[9] Li, T., Zong, W., Wang, Y., & Tian, H. (2023). Nuances are the Key: Unlocking ChatGPT to find failure-inducing tests with differential prompting. In *38th IEEE/ACM international conference on automated software engineering* (pp. 14-26). Luxembourg: Luxembourg. doi: 10.1109/ASE56229.2023.00089.

[10] Luu, Q., Liu, H., & Chen, T. (2023). *ChatGPT advance software testing intelligence? An experience report on metamorphic testing*. doi: 10.48550/arXiv.2310.19204.

[11] Mercorelli, P. (2024). Recent advances in intelligent algorithms for fault detection and diagnosis. *Sensors*, 24(8), article number 2656. doi: 10.3390/s24082656.

[12] Mudgal, P., & Wouhaybi, R. (2023). An assessment of ChatGPT on log data. In *AI-generated content* (pp. 148-169). Singapore: Springer. doi: 10.48550/arXiv.2309.07938.

[13] Popp, L., & Müller, K. (2021). Technical reliability of shipboard technologies for the application of alternative fuels. *Energy, Sustainability and Society*, 11, article number 23. doi: 10.1186/s13705-021-00301-9.

[14] Ramalho, L. (2022). *Fluent python*. Sebastopol: O'Reilly Media, Inc.

[15] Regattieri, F., Bortolini, A., & Galizia, F.G. (2022). Data-driven fault detection and diagnosis: Challenges and opportunities in real-world scenarios. *Applied Sciences*, 12(18), article number 9212. doi: 10.3390/app12189212.

[16] Suhail, A.H., Guangul, F.M., & Nazeer, A. (2024). Advanced system diagnostics tools: Innovations and applications. *IntechOpen*. doi: 10.5772/intechopen.114378.

[17] Theofanis, P., & Raptis, A. (2022). On efficiently partitioning a topic in Apache Kafka. In *International conference on computer, information and telecommunication systems (CITS)* (pp. 1-8). Greece: Piraeus. doi: 10.1109/CITS55221.2022.9832981.

[18] Vychuzhanin, V., Rudnichenko, N., Vychuzhanin, A., & Rychlik, A. (2023). Diagnosis intellectualization of complex technical systems. In *ICST-2023: Information control systems & technologies* (pp. 352-362). Odesa.

[19] Wang, C., Chen, X., Qiang, X., Fan, H., & Li, S. (2024). Recent advances in mechanism/data-driven fault diagnosis of complex engineering systems with uncertainties. *AIMS Mathematics*, 9(11), 29736-29772. doi: 10.3934/math.20241441.

[20] Wang, G., *et al.* (2021). Consistency and completeness: Rethinking distributed stream processing in Apache Kafka. In *International conference on management of data SIGMOD '21* (pp. 2602-2613). New York: Association for Computing Machinery. doi: 10.1145/3448016.345755.

[21] Yan, W., Wang, J., Lu, S., Zhou, M., & Peng, X. (2023). A review of real-time fault diagnosis methods for industrial smart manufacturing. *Processes*, 11(2), article number 369. doi: 10.3390/pr11020369.

Використання ChatGPT під час інтелектуальної діагностики технічного стану складних технічних систем

Володимир Вичужанін

Доктор технічних наук, професор
Національний університет «Одеська політехніка»
65044, просп. Шевченка, 1, м. Одеса, Україна
https://orcid.org/0000-0002-6302-1832

Олексій Вичужанін

Доктор філософії, асистент
Національний університет «Одеська політехніка»
65044, просп. Шевченка, 1, м. Одеса, Україна
https://orcid.org/0000-0001-8779-2503

Анотація. Інтелектуальна діагностика складних технічних систем, зокрема суднових енергетичних установок, має важливе значення для забезпечення раннього виявлення несправностей і підтримки експлуатаційної надійності. У цьому дослідженні представлено методологічний підхід до інтеграції мовної моделі ChatGPT в автоматизовану діагностику СЕС. Метою даної роботи була розробка методологічного підходу до використання мовної моделі ChatGPT в автоматизованій діагностиці складних технічних систем, зокрема СЕС. Запропонована методологія складалась з декількох етапів: збір даних, попередня обробка, навчання моделі, виявлення аномалій та генерація діагностичних рекомендацій. Система інтегрувала ChatGPT з потоковою передачею даних у реальному часі (Kafka) та нейромережевим виявленням аномалій з використанням автокодерів та моделей довготривалої короткочасної пам'яті. Експериментальна перевірка була проведена з використанням реальних операційних наборів даних з суднових енергетичних установок. Запропонований підхід продемонстрував значне покращення точності виявлення несправностей, збільшивши її на 15 % порівняно з традиційними пороговими методами. Час діагностики скоротився в 9 разів, що дозволило ідентифікувати відхилення майже в реальному часі. Модель досягла точності 92,8 % при класифікації аномальних станів і правильному визначенні несправностей на ранніх стадіях за такими ключовими параметрами, як тиск, температура і швидкість обертання. Аналіз розподілів помилок реконструкції підтвердив здатність системи розрізняти нормальну та аномальну поведінку системи. Виявлені аномалії були перехресно підтверджені експертними оцінками, що підтвердило практичну застосовність моделі в реальних умовах діагностики. Крім того, реалізація запропонованого підходу дозволяє здійснювати предиктивне планування технічного обслуговування, що сприяє зниженню експлуатаційних ризиків і зменшенню витрат на обслуговування. Інтеграція ChatGPT в системи діагностики суднових енергетичних установок покращує автоматизовану обробку технічної документації та експлуатаційних журналів, підвищуючи прозорість і точність пояснень несправностей. Результати цього дослідження можуть бути застосовані в суднобудуванні, промисловій автоматизації та технічному обслуговуванні, сприяючи підвищенню безпеки та надійності складних технічних систем

Ключові слова: прогнозне технічне обслуговування; виявлення аномалій; енергетичні системи суден; інтелектуальний моніторинг; аналітика в реальному часі; оцінка на основі штучного інтелекту; експлуатаційна надійність