INTERNATIONAL JOURNAL OF COMPUTER INTEGRATED MANUFACTURING
https://doi.org/10.1080/0951192X.2026.2657824

RESEARCH ARTICLE

Autoencoder-driven fault detection in drilling machines: a study of hybrid deep learning models

Amin Karimi Dastgerdi[a], Paolo Mercorelli[a] and Hamidreza Nemati[b]

[a]Institute for Production Technology and Systems, Leuphana University of Lueneburg, Lueneburg, Germany; [b]College Lane, School of Physics Engineering and Computer Science (SPECS), University of Hertfordshire, Hatfield, UK

ABSTRACT
Detecting faults promptly and accurately in manufacturing is crucial for maintaining efficiency, productivity, and product quality. This research explores the potential of using autoencoders, including models such as Long Short-Term Memory (LSTM) networks, Bidirectional Long Short-Term Memory (BiLSTM), WaveNet, and hybrid models that combine 1D Convolutional Neural Networks (1D CNN) with BiLSTM layers, for real-time fault detection in drilling operations. The autoencoders were trained on data from normal operational conditions to learn standard patterns and then tested on datasets containing fault instances. Results show that autoencoders with BiLSTM and LSTM layers excelled at capturing temporal dependencies. Hybrid models, such as WaveNet-BiLSTM and CNN-BiLSTM, which integrate LSTM and convolutional layers, significantly enhanced fault detection performance, achieving perfect accuracy, precision, recall, and F1 scores with just 10% of normal training samples. The models' effectiveness was also evaluated using shorter time windows, resulting in an accuracy of 98%. This advancement facilitates the integration of such models into intelligent manufacturing systems for proactive maintenance and operational optimization, potentially reducing downtime and maintenance costs while maintaining high product quality.

ARTICLE HISTORY
Received 24 October 2024
Accepted 27 March 2026

KEYWORDS
Fault detection; tool condition monitoring; smart manufacturing; deep learning; autoencoders

1. Introduction

The rise of intelligent manufacturing under Industry 4.0 has transformed how industrial processes are monitored and controlled, underscoring the critical importance of reliable online fault detection. In the machinery industry, particularly in drilling operations, the inability to promptly identify faults can result in significant downtime, increased maintenance costs, and a decline in product quality (Achouch et al. 2022). Consequently, reliable fault detection systems are paramount for ensuring operational efficiency and maintaining high manufacturing standards.

Across production drilling lines, faults such as missing tools, tool wear or breakage, and drivetrain anomalies drive scrap, rework, and costly downtime (Finke et al. 2021); industry case studies report machine-learning monitoring for missing-tool detection in drilling cells and related machining contexts. For example, drilling tool faults have been addressed with ensemble and deep models in industrial datasets (Dastgerdi, Lange, and Mercorelli 2022), while acoustic and vibration pipelines for drill-failure detection achieve high accuracy but typically require high-rate sensors and careful calibration (Tran, Truong Pham, and Lundgren 2022; Yu et al. 2022; X. Zhang et al. 2024). Similar pressures appear in other sectors – petrochemical process monitoring on the edge (Santo et al. 2023), steel manufacturing quality control (Abdullahi et al. 2020), wind-turbine condition monitoring (Jiang et al. 2017), and additive manufacturing (Scheffel, Augusto Fröhlich, and Silvestri 2021) – where reliability must be balanced against sensor cost, labelling burden, and changing operating regimes. These cases motivate label-efficient, sensor-light approaches that integrate with existing signals (such as spindle power) and remain robust under drift.

Autoencoders, a class of artificial neural networks specifically designed for unsupervised learning, have recently emerged as a powerful tool in machine learning, especially for anomaly detection tasks. In the context of fault detection, autoencoders are particularly valuable because they learn normal operational patterns and flag deviations via reconstruction error (Sakurada and Yairi 2014).

The versatility and effectiveness of autoencoders are further demonstrated by their wide applications across diverse fields. In image processing, they are employed

CONTACT Hamidreza Nemati H.Nemati@herts.ac.uk College Lane, School of Physics Engineering and Computer Science (SPECS), University of Hertfordshire, Hatfield, UK

for noise reduction and image compression (Kumbhar 2023). In natural language processing, autoencoders assist in tasks such as language translation and sentiment analysis by learning compact representations of textual data (Bahdanau, Cho, and Bengio 2014). In industrial and manufacturing environments, autoencoders play a pivotal role in predictive maintenance and fault detection (Skog 2022), facilitating continuous machine monitoring by identifying potential issues at early stages and optimizing maintenance schedules.

Traditional fault detection methods, such as statistical process control and basic threshold-based techniques, often struggle to capture complex patterns and temporal dependencies inherent in sensor data (Abid, Khan, and Iqbal 2021). Recent advancements in deep learning techniques, particularly autoencoders, have shown significant promise for future applications. Autoencoders have demonstrated their ability to identify intricate patterns and features, offering considerable potential for anomaly and fault detection (Cacciarelli and Kulahci 2022; Jang et al. 2021; Qian et al. 2022).

Although supervised machine learning has achieved notable success in fault detection, it often depends on fault-labeled data that are costly or infeasible to obtain. Furthermore, traditional fault detection in drilling machines – including systems historically used by MARPOSS (ARTIS) – relies on threshold-based monitoring of variables such as spindle power or current and extensive expert tuning; these schemes can be brittle when faults are unseen or when process conditions drift. To address these limitations, this study proposes a fully data-driven, unsupervised framework that detects faults using only normal operation data. This avoids the expense and risk of collecting fault labels, improves cost-effectiveness, and facilitates integration with existing monitoring systems. To capture both local signal patterns and long-range temporal dependencies in drilling power signals, in this study architectures that combine convolutional layers (local features) with recurrent or dilated temporal models (long-range dynamics) (Lawal et al. 2021; Pan and Wu 2023; Schuster and Paliwal 1997; Van Den Oord et al. 2016) are considered. In this regard, this study represents two hybrid autoencoder frameworks that combine Long Short-Term Memory networks, 1D Convolutional Neural Networks, and WaveNet architectures to detect faults at earlier stages, which is crucial for reducing operational damage and costs. Moreover, a robust, nonparametric thresholding scheme based on reconstruction-error Z-scores and the median absolute deviation (MAD) is employed, which adapts to operating-regime shifts and resists outliers, strengthening performance in real-world conditions.

The remainder of this paper is organized as follows. Section 2 reviews related work and the state of practice. Sectioon 3 details the methodology, including the autoencoder pipeline, reconstruction error computation, and dynamic thresholding. Section 4 describes the data collection, preprocessing, and implementation setup. Section 5 reports the experimental results across architectures, window lengths, and training fractions. Section 6 discusses the findings and deployment implications. Section 7 concludes the study, outlines limitations and presents directions for future work.

2. Literature review

2.1. Challenges in fault detection

Despite significant advancements in machine learning – based fault detection, several practical challenges persist, particularly in industrial environments. Supervised learning methods, while highly accurate, rely heavily on the availability of labeled fault data. In many manufacturing contexts, fault events are rare, expensive, or dangerous to simulate, making it impractical to collect large, representative datasets for supervised training. Moreover, supervised models often require periodic retraining as machines, tools, or operational conditions evolve, further increasing maintenance complexity and costs.

While autoencoders, particularly deep learning – based ones, are powerful tools for fault detection in rotating machinery (H. Li et al. 2025; Zachariades and Xavier 2025), several challenges must also be addressed. For instance, a key difficulty lies in developing fault detection models that are reliable and adaptable enough to handle the fluctuations and disturbances present in real-world industrial data (Cao and Yunusa-Kaltungo 2021). Additionally, the performance of autoencoders can degrade significantly when confronted with noise, irregularities, or subtle and rare faults in operational data (Majumdar 2018). In scenarios where data exhibit complex, nonlinear dependencies across different time scales, traditional autoencoders may fail to accurately reconstruct the input signals (Deng and Hooi 2021). Furthermore, autoencoders can sometimes generalize too well, reconstructing abnormal inputs with low error, which reduces anomaly detection sensitivity (Gong et al. 2019). Another challenge concerns the interpretability of the learned representations, which is crucial for effective maintenance decision-making (Christoph 2020).

These challenges highlight the importance of selecting appropriate model architectures tailored to the specific characteristics of the data under analysis. Moreover, there remains a critical need for fault detection approaches that minimize dependency on specialized sensors, extensive preprocessing, and labeled fault data, while maintaining robustness against operational variability. Accordingly, a key need is for label-efficient, sensor-light methods that remain stable under regime shifts and noise, while minimizing manual calibration and preprocessing.

This study addresses these gaps by developing a spindle power – based, unsupervised fault detection framework using deep hybrid autoencoder architectures. The proposed method operates directly on raw sensor data, applies dynamic statistical thresholding, and requires only normal operation records for training, enabling scalable, real-time, and cost-effective monitoring in modern manufacturing environments.

Specifically, this research focuses on overcoming the challenge of insufficient fault data, which is often difficult or costly to generate. The developed framework detects faults solely based on training with normal process data. Additionally, the study investigates the capability of the proposed autoencoder models to detect faults earlier by reducing the time window size, thereby allowing faults to be identified before the end of the process and helping to prevent major damage and associated costs.

2.2. Supervised learning in fault detection

Supervised learning is a branch of machine learning where an algorithm is trained using labeled data, meaning each training example is associated with a known output label. The primary objective is for the algorithm to learn the mapping from inputs to output labels, enabling accurate predictions for unseen data. Prior supervised approaches vary mainly by sensing modality and network family; here we compare them on accuracy, data requirements, sensor/preprocessing cost, and robustness to changing conditions, in order to derive the problem this paper addresses.

Supervised machine learning algorithms have become increasingly popular for fault detection due to their ability to recognize complex patterns within data. Techniques such as Support Vector Machines (SVM), Decision Trees (DT), and Neural Networks (NN) have been widely applied in manufacturing processes. Prior work generally examines these techniques within three sensing pipelines – power/current, acoustic, and vibration/force. For instance, in power-signal pipelines, Dastgerdi, Lange, and Mercorelli (2022) achieved significant improvements in process monitoring and missing tool fault detection by employing nonlinear ensemble models using power signal datasets from 1 mm drilling tools. Similarly, Zhu, Yan, and Lu (2020) introduced a fault diagnosis method for disc slitting machines based on a combination of Wavelet Packet Transform (WPT) and SVM classifiers, achieving a classification accuracy of 95.6%. Power thresholds are inexpensive and easy to integrate, but rely on expert limit setting and are brittle under unseen faults or process drift; generalizing across tools and materials typically requires manual retuning.

Several other studies have focused on acoustic pipelines leveraging deep learning and hybrid models. For example, X. Zhang et al. (2024) used acoustic signals processed through a CNN to detect drill bit anomalies, reaching 95% accuracy. In another study, Tran, Truong Pham, and Lundgren (2022) applied CNN and attention-based LSTM networks on log-Mel spectrograms for real-world drill failure detection, achieving accuracies between 92.62% and 97.47%. Although acoustic methods using CNNs/LSTMs report strong drill-failure accuracy, they typically require high-rate microphones, careful calibration, and stable acoustic environments; performance can degrade with background noise and line changes.

Among vibration/force approaches, Hernandez-Matamoros et al. (2023) proposed a semi-supervised approach for early stuck pipe detection in drilling operations, outperforming purely supervised methods. In another research, Yu et al. (2022) developed 1D and 2D CNN models based on vibration signals transformed using Short-Time Fourier Transform (STFT), achieving an accuracy of 97.33%. Other notable contributions include a machine-learning framework for reliable fault notification in petrochemical processes using decision trees and Gaussian naïve Bayes (Santo et al. 2023) and a data-driven fault-detection method based on shifts in the probability distributions of residual signals in dynamic systems (Q. Wang et al. 2023). Traditional machine learning approaches have also been extended through optimization techniques. In another research, Rajakannu et al. (2024) employed an ANN with wavelet packet – transformed vibration features to achieve 98.05% accuracy. Furthermore, Dai and Huang (2024) utilized a supervised CWT-SqueezeNet pipeline for classifying nine fault types with an accuracy of 97.77%. In another study, Abdullahi et al. (2020) compared Logistic Regression, Naive Bayes, and SVM models for fault detection in steel manufacturing, finding that Logistic Regression offered the highest precision while SVM achieved the highest recall. Vibration/force approaches with time – frequency transforms and supervised CNN/SVM models also achieve high accuracy, but depend on additional sensors and handcrafted or heavy preprocessing, and they need fault labels to train or retune.

Further, optimization techniques have been used to enhance performance of supervised learning models.

For example, Van, Tang Hoang, and Jun Kang (2020) integrated Particle Swarm Optimization with least squares SVM for bearing fault diagnosis. In another study, X. Li et al. (2020) proposed a PSO-SVM framework for fault detection in high-voltage circuit breakers. In another research, Fan et al. (2020) introduced an optimized SVM using a self-regulating particle swarm algorithm for bearing fault detection. A modified Grey Wolf Optimization algorithm is applied by Huang et al. (2020) to enhance SVM performance for dissolved gas analysis in transformers. In another study, Y.-P. Zhao et al. (2020) developed a weighted one-class SVM approach for turboshaft engine fault detection.

While these supervised methods demonstrate strong classification performance, their reliance on labeled fault data poses a significant limitation. In many industrial applications, fault data are rare, costly, or unsafe to generate. Consequently, there is a growing need for reliable unsupervised solutions capable of detecting faults based solely on normal operational data.

2.3. Autoencoders in fault detection

Autoencoders have been a cornerstone of unsupervised learning since their introduction by Hinton and Zemel (1993). Initially designed to address the problem of dimensionality reduction, they offered a nonlinear alternative to traditional methods such as Principal Component Analysis (PCA) (Pearson 1901). Over the decades, autoencoders have undergone significant advancements, evolving from simple architectures to complex models capable of handling diverse data types. These developments have demonstrated their potential to capture essential data features while reducing dimensionality.

Further innovations include Variational Autoencoders (VAEs) introduced by Kingma and Welling (2013), which gained popularity in generative modeling, and Denoising Autoencoders (DAEs) introduced by Vincent et al. (2008), designed to learn robust features insensitive to noise and input variations.

The advent of deep learning revolutionized autoencoders. Deep autoencoders, comprising multiple hidden layers, were shown to be more effective at learning hierarchical data representations. Convolutional Autoencoders (CAEs), first introduced by Lee (2010), incorporated convolutional layers to capture spatial hierarchies, while 1D Convolutional Neural Network (1D CNN) autoencoders were developed for sequential data processing (Ren et al. 2019). For example, Senjoba et al. (2025) demonstrated the effectiveness of a CAE-based approach for anomaly detection in rotary percussion drilling, achieving an F1 score of 0.9412 and an AUC of 0.9893.

Convolutional neural networks have also been applied extensively for fault diagnosis. For instance, Zhou et al. (2020) tackled data imbalance in rotating machinery diagnosis using CNNs. Furthermore, K. Zhang et al. (2020) proposed a compact CNN with multiscale feature extraction for mechanical fault diagnosis. In another study, Yongbo et al. (2020) and Chen et al. (2020) demonstrated the use of CNNs in diagnosing faults from infrared thermal images and vibration signals, respectively. In another research, Y. Liu et al. (2020) achieved high robustness in fault detection using energy spectrum matrices and hierarchical adaptive CNNs. A wide-structured Deep Neural Network (DNN) is introduced by Hoang et al. (2021) integrating multiple sensor signals for bearing fault diagnosis.

These advancements paved the way for the broader application of autoencoders across domains such as image denoising, compression, natural language processing, and anomaly detection. Autoencoders have been effectively applied for fault detection in various industrial applications, often combined with clustering or ensemble learning methods, as seen in studies by Vincent et al. (2008) and Amruthnath and Gupta (2018). In industrial production systems, Windmann and Westerhold (2022) demonstrated the tuning of an LSTM autoencoder to achieve an F1 score of 0.8. Similarly, Iannace, Ciaburro, and Trematerra (2019) showcased a neural network approach for detecting UAV propeller faults, achieving an accuracy of 0.97.

Recent innovations continue to push the boundaries. Denoising autoencoders and attention-based deep convolutional networks have also been applied to multivariate anomaly detection (W. Liu et al. 2024). A combination of deep residual networks (ResNet) and autoencoders was proposed to enhance feature extraction and fault detection (Jiang et al. 2017; J. Wang, Li, and Zhang 2024). However, many pipelines still depend on heavy time – frequency preprocessing or specialized sensors, which can raise deployment cost; this reinforces the value of unsupervised models that operate on widely available power signals.

Specifically in rotating machinery, autoencoders have shown potential in detecting faults such as gear wear and rotor anomalies (Yan and Zhao 2024). Their primary advantage lies in their ability to automatically extract meaningful features directly from raw sensor data without extensive manual preprocessing (Toma, Piltan, and Kim 2021). Researchers have proposed various enhancements, including stacked autoencoders (Vincent et al. 2010) and adversarial training techniques (Dumoulin et al. 2016), to further improve detection performance.

Although autoencoders have demonstrated significant promise for fault detection in industrial settings, several practical challenges remain. Many approaches still rely on complex preprocessing steps, such as time-frequency transformations or wavelet decompositions. Others require specialized hardware, such as high-sample-rate microphones for acoustic monitoring or dedicated force sensors for mechanical signal capture. Additionally, methods based on audio analysis often necessitate manual calibration and controlled experimental environments to maintain performance. These factors can complicate large-scale deployment and limit practical scalability. Thus, there is a growing need for autoencoder-based approaches that minimize dependency on specialized hardware, reduce preprocessing complexity, and operate effectively using raw signals under real-world industrial conditions.

2.4. Rationale for model selection

The primary objective of this study is to investigate the effectiveness of hybrid deep autoencoder models combining Long Short-Term Memory networks (LSTM), Bidirectional Long Short-Term Memory networks (BiLSTM), 1D Convolutional Neural Networks (1D CNN), and WaveNet (WN) compared to single models such as standard Neural Networks (NN), LSTM, and BiLSTM for capturing data dynamics and accurately detecting faults during drilling operations.

BiLSTM networks have demonstrated superior performance over traditional LSTM models and alternatives such as Transformers in capturing temporal dependencies, particularly in ultra-long sequences, making them more effective for highly irregular or noisy time-series data (Tang and Zhang 2022). Their ability to process information bidirectionally – both forward and backward through time – enhances their capacity to capture complex patterns necessary for accurate fault detection (Schuster and Paliwal 1997). However, BiLSTM models may struggle with capturing nonlinear relationships in highly irregular data, suggesting the need for hybrid approaches that integrate convolutional layers to enhance feature extraction (Malczewski and Czubak 2021; R. Zhao, Lei, and Zhao 2024).

WaveNet architectures are designed to handle high-frequency time-series data by leveraging stacks of dilated causal convolutions, enabling the capture of long-range dependencies without requiring recurrent connections (Gong et al. 2019; Van Den Oord et al. 2016). This hierarchical convolutional structure allows WaveNet to model both short- and long-term patterns effectively. Furthermore, WaveNet models are capable of incremental training with new data, making them suitable for dynamically changing environments (Saurav et al. 2018). Their convolutional nature also provides robustness against input noise (Ghosh and Kataria 2020). However, limitations remain when dealing with significant nonlinearities or when sequential structure is weak, suggesting that combining WaveNet with BiLSTM may enhance overall model performance.

1D CNNs are highly effective at extracting local and global features from sequential data without strong assumptions about the input structure (Lawal et al. 2021). Their architecture – comprising convolutional, pooling, and flattening layers – enables them to detect both small and large-scale patterns. Early layers with smaller kernels capture fine-grained features, while deeper layers with larger kernels summarize broader trends (Y. Liu et al. 2021). 1D CNNs have shown robustness against transformations, scaling, and noise, and have been successfully applied to real-time fault diagnosis in vibration-based anomaly detection (Gharghory 2021; H. Wang, Liu, and Ai 2022). Combining 1D CNNs with recurrent models like LSTM or BiLSTM further enhances the model's ability to capture both local spatial and long-term temporal dependencies (Pan and Wu 2023).

Based on these advantages, hybrid models combining 1D CNN with BiLSTM layers were selected. In these models, the 1D CNN layers extract local features from the time-series input, while the BiLSTM layers model long-term dependencies. This integration allows the hybrid architectures to simultaneously detect local anomalies and broader temporal trends, enhancing overall fault detection capabilities.

Each autoencoder model employed in this study follows an encoder – bottleneck – decoder structure. Neural Networks (NN) serve as a baseline to evaluate the performance of simple feedforward architectures. LSTM and BiLSTM models are incorporated due to their proficiency in capturing long-term dependencies in time-series data, critical for monitoring drilling operations. 1D CNN layers are utilized for extracting intricate local features, while WaveNet models, known for their capacity to capture hierarchical temporal patterns, are integrated into the hybrid architectures to enhance reconstruction performance. The characteristics of the employed autoencoders are summarized in Table 1.

3. Methodology

This section outlines our end-to-end procedure for proposed unsupervised fault detection approach during drilling operations. Fault detection in this research relies entirely on analyzing the reconstruction error generated by trained autoencoder models and is fully data-driven. In this study a standard unsupervised fault detection pipeline in which autoencoders learn normal machine behavior and flag deviations at inference via

Table 1. Characteristics of the employed autoencoders.

	Structure				
Autoencoder	Encoder	Decoder	AF	Loss	Padding
NN	(128, 64, 32)	(64, 86)	tanh	MAPE	same
CNN	(128, 64, 32, 16)	(32, 64, 128)	relu	MSE	same
LSTM	(256, 64, 64, 32)	(64, 64, 256)	tanh	MAE	same
BiLSTM	(128, 64)	(128)	relu	MAPE	same
1D-CNN-BiLSTM	C (64, 32)-B (128, 64, 32)	B (64, 128)-C (32, 64)	tanh	MAPE	same
WN_BILSTM	B (64, 32)-RB (7×32)	C (32)-B (32, 64)	relu	MAPE	same

reconstruction error is employed. The methodology proceeds in brief as follows: (i) data collection and preprocessing, (ii) model fitting, (iii) calculating reconstruction errors, (iv) threshold optimization, and (v) fault detection and monitoring. The overall framework of the proposed fault detection approach is illustrated in Figure 1.

3.1. *Data collection and preprocessing*

Firstly, sensor data on spindle active power were collected in the form of JSON files using a GEMTP (Generic Equipment Monitoring and Telemetry Platform) module (without an additional filter) from drilling machines to a MQTT broker, located at the Egestorf branch of Marposs Monitoring Solutions GmbH (ARTIS) in Egestorf, Germany. MQTT is an efficient publish – subscribe protocol that enables real-time data exchange between IoT devices involved in the assembly process. Each device can publish data to designated topics and subscribe to receive messages from those topics, facilitating real-time monitoring and control.

The GEMTP module, when installed within an electrical cabinet, functions as an effective standalone device for tool condition monitoring. Marposs Hall sensors were used to measure the power consumed by the spindle during machining. The GEMTP module supports three distinct monitoring strategies and can manage up to 127 cutting cycles, each with four types of configurable limits (e.g. tool breakage or missing tool, overload, and tool wear) per cycle.

Spindle active power was selected for analysis due to its demonstrated effectiveness in fault detection, as shown in a previous study conducted by the authors (Dastgerdi, Lange, and Mercorelli 2022). The dataset includes both normal operations and malfunctioning processes across 86 time-stamp intervals. Data collection was performed over a sufficiently long period to capture a wide range of operational conditions and faulty occurrences. After data acquisition, preprocessing steps, including cleaning and Z-score normalization, were applied to ensure that all features contributed equally during model training. This normalization helps mitigate the effects of differing feature scales. Each data segment was labeled as either normal or faulty depending on the presence of faults. To evaluate model performance under realistic conditions, an imbalanced dataset was used, reflecting real-world industrial environments where faults are rare events.

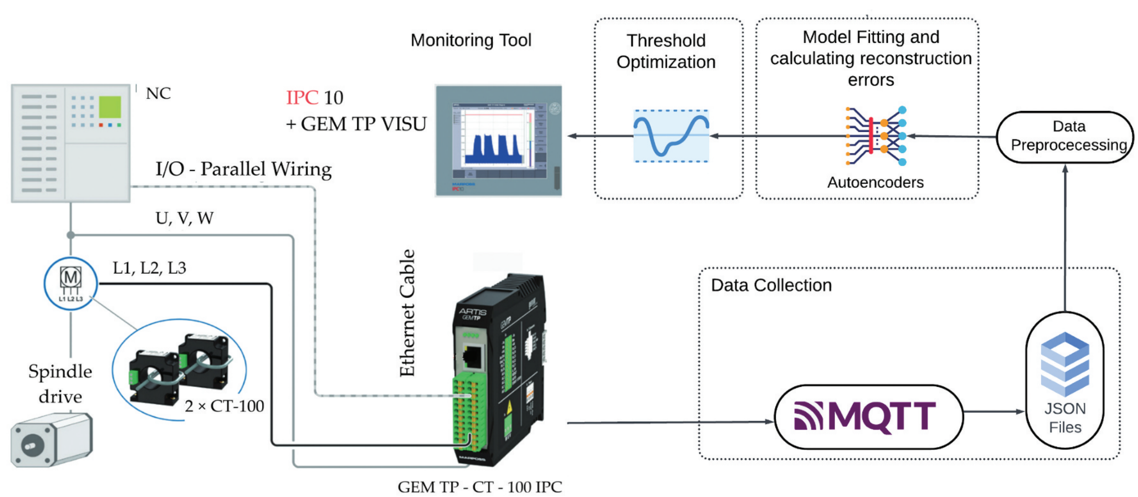

Figure 1. Schematic view of the proposed monitoring system.

3.2. Model fitting

After preprocessing and normalizing the collected data, the autoencoder models are trained exclusively on normal operating data, enabling them to learn a compact representation of healthy machine behavior. The proposed autoencoder architecture in this study, referred to as WN_BILSTM, is a hybrid neural network designed to efficiently capture and reconstruct sequential data by combining the strengths of convolutional layers and BiLSTM layers. It follows a traditional encoder – bottleneck – decoder structure. The encoder, which compresses the input data into a lower-dimensional latent representation, begins with BiLSTM layers that process the input by capturing long-range dependencies and temporal dynamics. The structure and detailed configuration of the proposed WN_BILSTM autoencoder are illustrated in Figure 2.

3.3. Calculating reconstruction errors

Once the models are trained, they are employed to generate reconstruction errors on new sensor data. Fault detection is then performed based on the analysis of these reconstruction errors. A significant deviation between the input data and its reconstruction indicates the presence of an anomaly.

This study employs a systematic approach that combines reconstruction error analysis with Z-score normalization to determine the optimal threshold for each model, thereby optimizing fault detection performance. Each model is first used to predict outputs for both normal and faulty data samples. The reconstruction error is computed by measuring the mean absolute difference between the original input data and the reconstructed output. This error serves as the primary indicator for identifying anomalies.

$$\textit{Mean Absolute Difference} = \frac{1}{N}\sum_{i=1}^{N}|x_i - \widehat{x}_i|, \quad (1)$$

The combined reconstruction errors from both normal and faulty data were concatenated, and the median and Median Absolute Deviation (MAD) were computed. MAD is especially useful when the data may contain outliers, as it provides a more robust estimate of variability compared to the standard deviation.

$$\textit{Median Absolute Deviation} = median(|x_i - median(x)|), \quad (2)$$

Which x_i represents each individual reconstruction error and $median(x)$ is the median of all reconstruction errors. To standardize the reconstruction errors and make them more comparable across different datasets, the errors were transformed into Z-scores.

$$Z = 0.6745 \times \frac{\textit{Reconstruction Error} - \textit{Median}}{\textit{Median Absolute Deviation}}, \quad (3)$$

The constant 0.6745 normalizes the MAD to be comparable to the standard deviation. This step ensures that the reconstruction errors are normalized relative to the distribution of errors across both normal and faulty data.

3.4. Threshold optimization

To detect irregularities and issues during drilling operations using autoencoder models, it is crucial to determine an optimal threshold that accurately differentiates between normal and faulty conditions. Selecting appropriate thresholds ensures not only statistical robustness but also practical effectiveness in distinguishing normal behavior from anomalies in drilling operations. Following the method proposed by Hundman et al. (2018), a nonparametric unsupervised approach is employed to dynamically calculate the

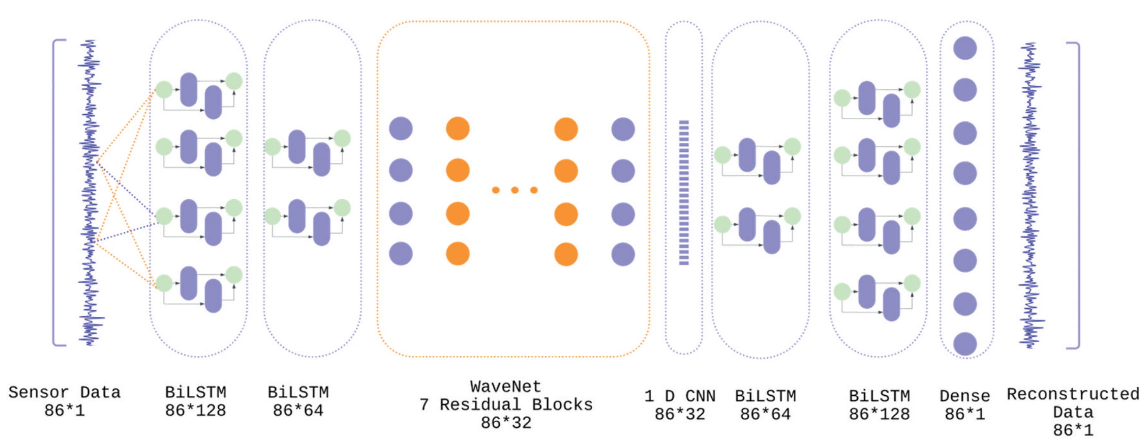

Figure 2. Structure of the proposed WN-BiLSTM autoencoder.

optimal threshold based on the statistical properties of the Z-scores.

The process begins by calculating the mean $\mu(es)$ and standard deviation $\sigma(es)$ of the Z scores. Using these values, a set of thresholds was established by spacing 100 values, between $\mu(es) + \sigma(es)$ and $\mu(es) +10 * \sigma(es)$. This range is chosen to ensure that the threshold is sensitive enough to detect anomalies while minimizing false positives. For each candidate threshold, its effectiveness is evaluated based on the reduction in mean and standard deviation of the Z-scores after applying the threshold. These reductions indicate how well the threshold separates normal behavior from anomalies. Additionally, a penalty is applied that accounts for the number and sequence of detected anomalies, to prevent the model from being overly sensitive. The penalty is proportional to both the number of errors exceeding the threshold and the continuity of these errors.

A scoring function then combines the reduction in mean and standard deviation with the penalty term, favoring thresholds that achieve significant reductions in error metrics while minimizing excessive anomaly detection. The score is calculated as follows:

$$Score(\varepsilon) = \left(\frac{\Delta\mu(\mathrm{es})}{\mu(es)} + \frac{\Delta\sigma(\mathrm{es})}{\sigma(es)}\right) \times \left(len(ea)^2 + len(Eseq)^2\right)^{-1}, \tag{4}$$

In this formula $\Delta\mu(\mathrm{es})$ represents the reduction in mean error when values above the threshold are excluded. $\Delta\sigma(\mathrm{es})$ represents the reduction in standard deviation when values above the threshold are excluded. $len(ea)$ is the number of errors above the threshold and $len(Eseq)$ is the number of consecutive sequences of these errors.

For each threshold under consideration predictions were made by categorizing data points as anomalous if their Z score exceeded the threshold. The threshold that achieves the highest score is selected as the optimal threshold for the model. Once the optimal threshold for a model was determined it could be utilized in real world applications to identify faults. This method ensures that the chosen threshold is not only statistically sound but also practically effective in distinguishing normal behavior from anomalies in drilling operations. This consistent pattern suggests that there may not be a need to repeat the process of selecting thresholds in instances enabling fault detection without reliance on labeled data. Implementation details (hardware, libraries, callbacks, and training schedules), along with data preparation and splits, are provided in Implementation.

3.5. Evaluation metrics and protocol

To assess the performance of the anomaly detection system based on autoencoders, standard evaluation metrics derived from the confusion matrix were employed. These metrics – accuracy, precision, recall, F1 score, specificity, and AUC-ROC – provide comprehensive insights into the models' effectiveness.

(1) **Precision**: Precision is crucial in scenarios where the cost of false positives is high. In fault detection, a high precision value indicates that most samples predicted as faults are indeed true faults. Precision is calculated as:

$$Precision = \frac{t_p}{t_p + f_p} \tag{5}$$

(2) **Recall**: Recall is important in situations where the cost of false negatives is significant. In the context of fault detection, a high recall rate means that the majority of actual faults are correctly identified. Recall is defined as:

$$Recall = \frac{t_p}{t_p + f_n} \tag{6}$$

(3) **F1 Score**: The F1 score represents the harmonic mean of precision and recall, providing a balanced measure of a model's accuracy, particularly when dealing with imbalanced data:

$$\mathrm{F1} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{7}$$

(4) **Specificity** This represents the percentage of normal samples that have been correctly identified.

$$Specificity = \frac{t_n}{t_n + f_p} \tag{8}$$

(5) **AUC-ROC**: The Area Under the Receiver Operating Characteristic Curve is a crucial metric for evaluating the performance of a binary classification model, particularly in cases where the dataset is imbalanced. The ROC curve plots the true positive rate (recall) against the false positive rate (1-specificity) at various threshold settings, providing a comprehensive view of the model's performance across different decision boundaries. The AUC, or the area under this curve, summarizes the model's ability to distinguish between classes.

4. Implementation

This section presents the details of the implementation process. All experiments were conducted on Google Colab using a T4 GPU, with Python version 3.10.12, Keras version 3.4.1, and Plotly version 5.15.0 for result visualization.

4.1. Model implementation and training

The collected dataset consists of normal (healthy operation) samples and faulty samples, explicitly labeled based on the presence of known fault events within defined intervals. Specifically, the dataset comprises 624 samples labeled as normal operations and 52 samples labeled as faulty, collected over various operational scenarios and fault occurrences to ensure comprehensive analysis. Detailed data collection and preprocessing procedures are provided in Section 3.1. To prepare the data for modeling, several preprocessing steps were undertaken:

- **Segmentation**: Continuous operational spindle power data was segmented into shorter fixed-length intervals (time windows) of 32, 64, and 86 timestamps to analyze model effectiveness under different temporal resolutions.
- **Normalization**: Data were normalized using the Z-score normalization method. Each feature was standardized to have a mean of zero and a unit standard deviation, ensuring that all data points contributed equally to the learning process, thereby improving the training efficiency and detection sensitivity of the autoencoder models.
- **Training and Testing Split**: Following segmentation and normalization, the normal dataset was further split into training and testing subsets using varying proportions (0.05, 0.1, 0.2, 0.3, and 0.4) to identify the minimum amount of training data required for effective fault detection. Importantly, only normal samples were used for training the autoencoder models, whereas both normal and faulty samples were reserved for evaluating model performance during testing.

The final dataset consists of 52 faulty samples and 624 normal samples. Faulty samples were reserved exclusively for evaluation and were not used during training. The normal samples were divided into different subsets for training and testing, with training set sizes of 5%, 10%, 20%, 30%, and 40% of the total normal data. The remaining normal data were used for testing.

This approach was adopted to investigate the minimum amount of training data necessary for the models to achieve effective fault detection performance. Additionally, in each experiment, 20% of the training set was allocated as a validation set to monitor performance and prevent overfitting. The exact distribution of samples used for training, validation, and testing is summarized in Table 2.

For most of the models, the Mean Absolute Percentage Error (MAPE) was used as the loss function during training. However, for the LSTM model, the Mean Absolute Error (MAE) was employed, and for the CNN model, the Mean Squared Error (MSE) was used, as these configurations produced superior results for their respective architectures. The definitions of the MAPE, MAE, and MSE functions are as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - r_i)^2, \tag{9}$$

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{x_i - r_i}{x_i}\right| \tag{10}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - r_i|, \tag{11}$$

Where x_i is the value of the input sensor data, r_i is the value of the reconstructed data by autoencoders and n is the number of processes.

The training process in this study involves iteratively adjusting the model weights over multiple epochs to minimize the reconstruction errors, with a focus on enhancing the models' ability to accurately regenerate normal operational states. To ensure reproducibility and robustness, the implementation and training of the selected models were performed in ten separate experiments.

To keep computational costs as low as possible, hyperparameters were tuned manually through trial-and-error optimization. In environments with advanced computational resources, more sophisticated optimization methods, such as metaheuristic algorithms, could be considered for future work.

To ensure optimal performance and efficient training of the autoencoders, early stopping and adaptive learning rate adjustment strategies were incorporated. These callbacks are essential for preventing overfitting, enhancing model robustness, and dynamically adjusting the learning rate throughout the training process.

Table 2. Number of Train, validation and test samples for different experiments.

Train portion [%]	# Train samples (only normal)	# Validation samples (only normal)	# Test samples	
			Normal	Faulty
5	25	6	593	52
10	50	12	562	52
20	100	25	499	52
30	150	37	437	52
40	200	50	374	52

Early stopping was applied to monitor the validation loss during training, with a patience parameter set to 10 epochs. Training was halted if the validation loss did not improve over 10 consecutive epochs, with the model weights reverted to those achieving the lowest validation loss, thereby preventing overfitting and improving generalization to unseen data.

Additionally, the 'ReduceLROnPlateau' callback was utilized to adjust the learning rate based on the validation loss. If the validation loss remained stagnant for 5 epochs, the learning rate was reduced by a factor of 0.5, with a minimum allowable learning rate set to 1e-6. This adjustment helped refine model convergence and avoid entrapment in suboptimal minima, contributing to more stable and effective training.

The autoencoder models were configured using the Adam optimizer and trained for up to 200 epochs with a batch size of 32. Training progress was monitored by observing reductions in reconstruction error across epochs. A consistent decrease in loss over time indicated that the models successfully assimilated the normal operational patterns.

4.2. Reconstruction errors and threshold

The trained autoencoders were evaluated on the testing dataset, which included both normal and faulty operations. In this phase, the autoencoders first reconstructed the normal test data. The reconstructed sequences were then compared to the original sequences to compute reconstruction errors.

Reconstruction errors were calculated by taking the Mean Absolute Error (MAE) between the original and reconstructed sequences for both the normal and faulty datasets. This metric provides a quantitative measure of how accurately the autoencoder can recreate the input data. Typically, lower reconstruction errors indicate better reconstruction quality, as expected for normal data. Conversely, higher reconstruction errors are anticipated for faulty data, since the autoencoder, trained exclusively on normal patterns, cannot accurately reconstruct unseen faults.

A threshold was subsequently determined based on the distribution of reconstruction errors, enabling the distinction between normal and faulty samples. This step represents the only supervised phase of the process, where labeled data was used solely for selecting the optimal threshold value. Data points with reconstruction errors exceeding the threshold were classified as faulty.

Figures 3 and 4 illustrate the reconstructed operations for both normal and faulty processes using the WN_BILSTM model, alongside their respective original operations across an entire process consisting of 86 time stamps.

To visually represent and compare the performance of the autoencoders in terms of reconstruction errors, scatter plots were generated to illustrate the differences

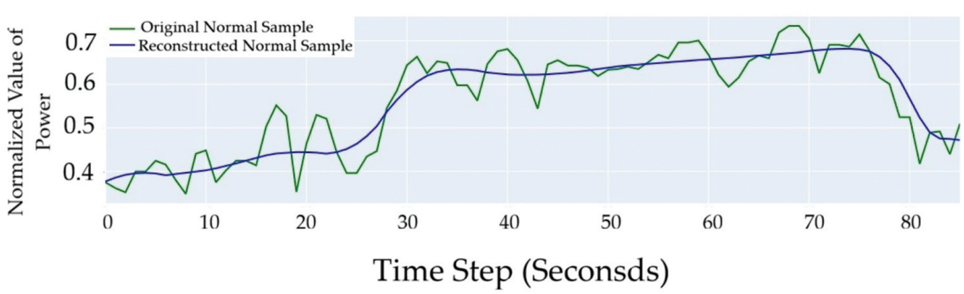

Figure 3. Demonstration of a normal sample and its reconstruction using WN_BILSTM.

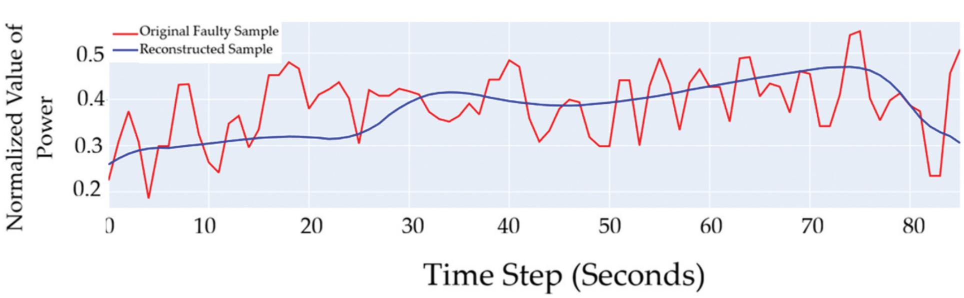

Figure 4. Demonstration of a faulty sample and its reconstruction using WN-BiLSTM.

in error values between normal and faulty data. These visualizations facilitate the identification of patterns in error distributions, highlighting how well the autoencoders can differentiate between faulty and normal data based on reconstruction errors.

Typically, normal data exhibit lower reconstruction error values, while faulty data display higher error distributions. This discrepancy forms the basis for fault detection, as processes with elevated reconstruction errors can be pinpointed as anomalies. Figure 5 illustrates the scatter plot visualizations of reconstruction errors and the corresponding identified anomalies, providing additional insights into the model's performance.

5. Results

This section reports the empirical performance of the unsupervised fault-detection framework across six autoencoder families (NN, CNN, LSTM, BiLSTM, CNN – BiLSTM, and WN – BiLSTM). First The performance of the models is evaluated using the metrics defined in Section 3.5, and then the architectures, data efficiency, and sequence-length effects is compared. Figures 6–8 provide a clear summary of evaluation metrics for the implemented models, across 10 experiments using three different time window sizes (32, 64, and 86). These figures illustrate how each model performs with different sizes of training data (ranging from 5% to 40% of available normal samples).

In Figure 6 (time window = 32), the proposed hybrid architecture consistently outperforms other models across key evaluation metrics achieving near-perfect precision and strong generalization. While CNN-BiLSTM shows slightly higher training times, it maintains competitive accuracy, especially in data-scarce settings. Unlike models such as CNN and NN, which suffer significant performance degradation at lower data proportions, CNN-BiLSTM retains strong metrics, indicating effective learning and robust generalization. The model's ability to maintain high precision without sacrificing recall suggests a good balance that avoids overfitting and underfitting.

In Figure 7 (time window = 64), CNN-BiLSTM continues to demonstrate superior and stable performance, with minimal variation across different training sizes. LSTM shows improved recall and F1 scores in this setting, highlighting its strength in capturing moderately extended temporal patterns. WN-BiLSTM also performs reliably across metrics, particularly excelling in recall and specificity. Both hybrid models significantly outperform simpler architectures like CNN and NN, confirming the advantage of combining convolutional and recurrent structures for fault detection.

As shown in Figure 8 (time window = 86), all models generally benefit from the longer sequence input; however, the hybrid models achieve the highest performance, particularly with smaller training sets (<20%). CNN-BiLSTM maintains top scores in most metrics, demonstrating its ability to model both temporal and long-term dependencies effectively. WN-BiLSTM again proves robust – especially in recall – making it highly suitable for applications where missing a fault is costly. Although CNN-BiLSTM requires more training time, its superior predictive performance justifies the computational cost. In contrast, shallow models such as CNN and

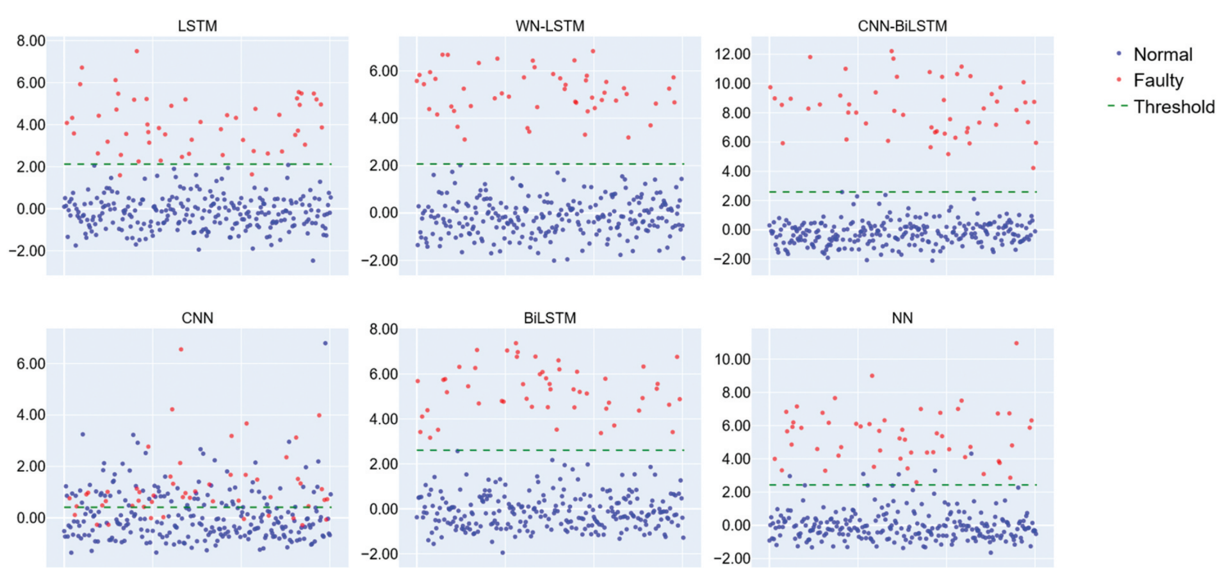

Figure 5. Autoencoders ability to detect faults based on reconstruction error.

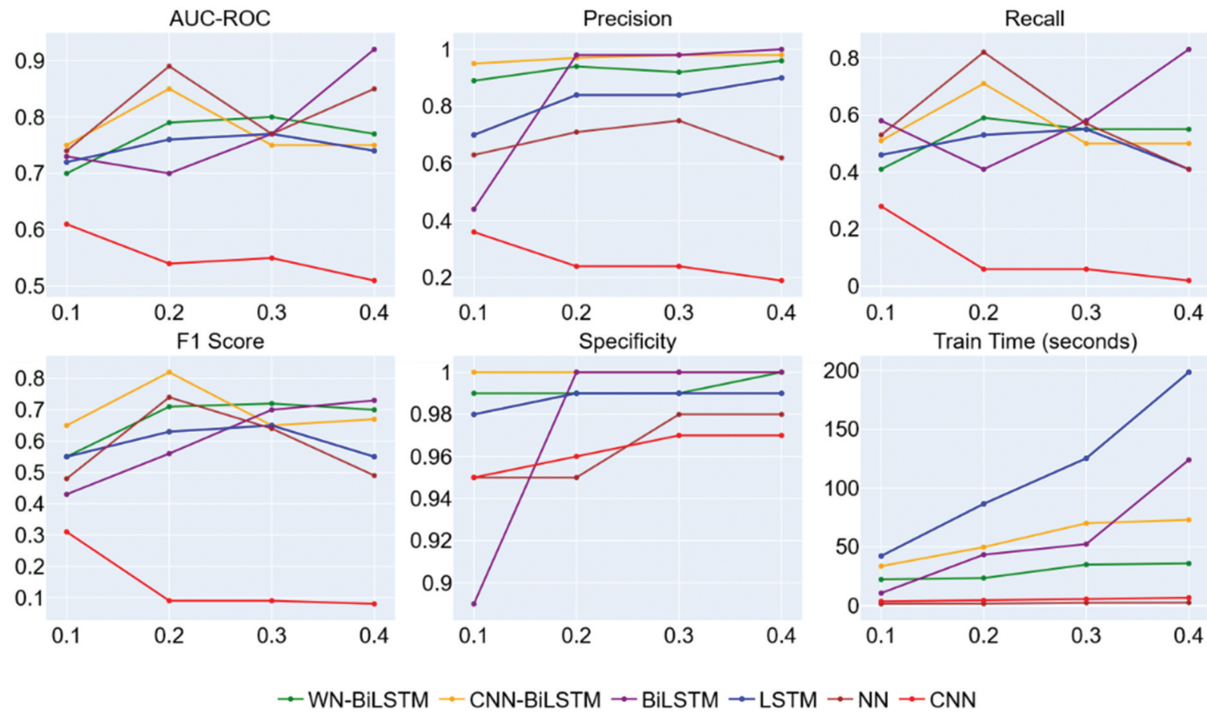

Figure 6. Mean performance results of different autoencoders over 10 experiments with time windows of 32 using different sizes of training samples.

NN exhibit unstable performance, reinforcing their limitations in complex temporal-spatial learning scenarios.

Overall, the results confirm that the hybrid models consistently outperform single-architecture baselines across various input lengths and training sizes. CNN-BiLSTM is particularly effective with shorter time windows and smaller training sets, while WN-BiLSTM shows its strength with longer sequences and larger

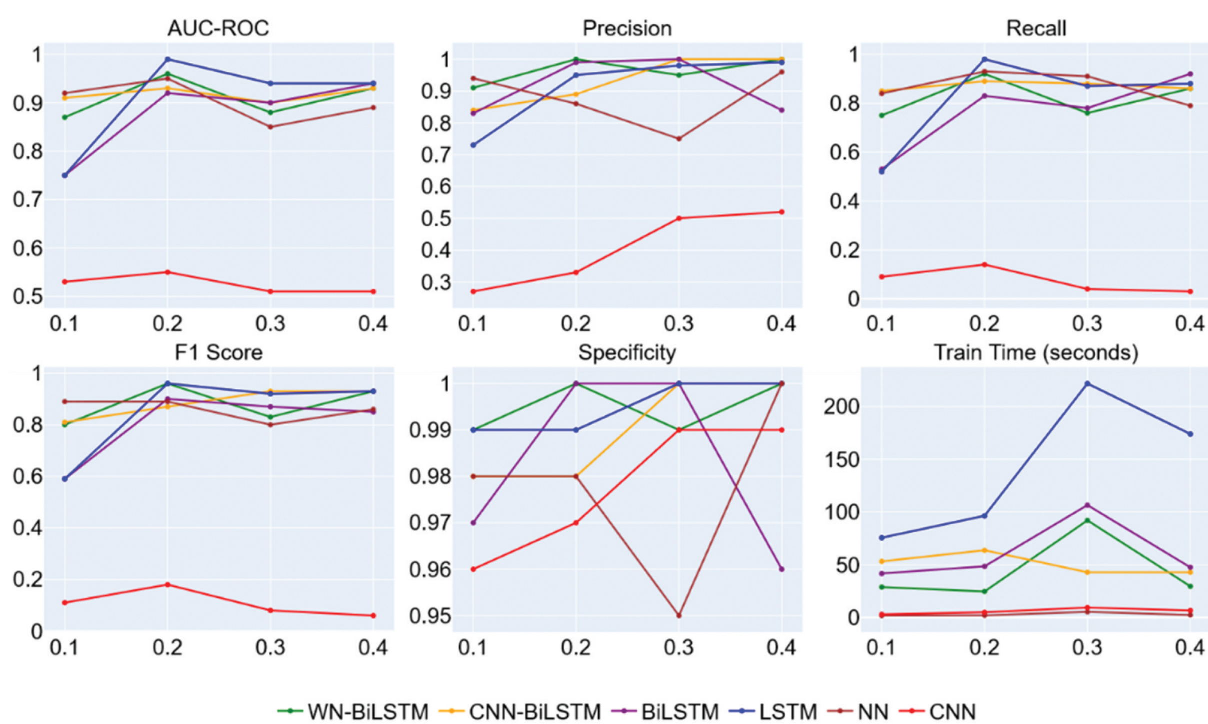

Figure 7. Mean performance results of different autoencoders over 10 experiments with time windows of 64 using different sizes of training samples.

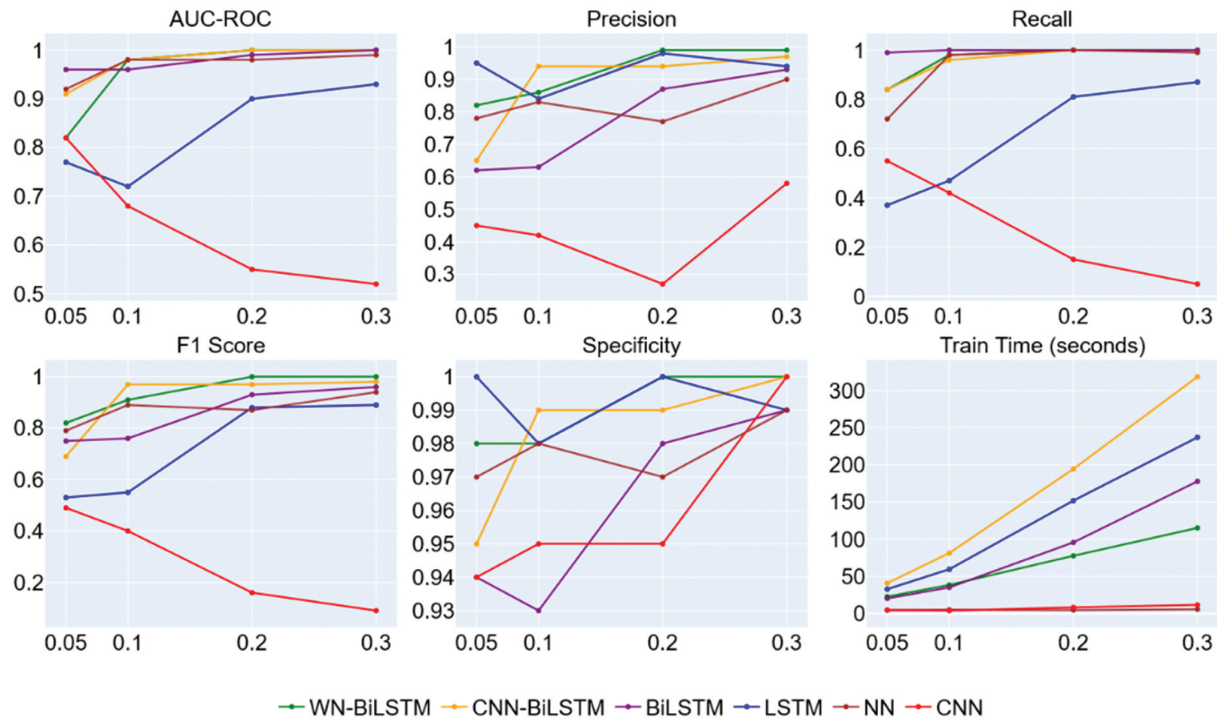

Figure 8. Mean performance results of different autoencoders over 10 experiments with time windows of 86 using different sizes of training samples.

datasets. Additionally, WN-BiLSTM's lower training time further enhances its practical value in time-sensitive deployment environments. In our experiments, CNN-BiLSTM performed best with shorter windows (size 64), while WN-BiLSTM remained effective even with only 62 normal training samples (~10% of 624 normal training samples), indicating complementary strengths for early detection and low-data regimes.

6. Discussion

In the realm of manufacturing the effectiveness of drilling procedures holds importance as it directly impacts both the quality of production and operational expenses. The detection of faults plays a role in upholding this effectiveness by averting machine harm minimizing downtime and guaranteeing a standard of product quality. Effective fault detection in drilling must also balance early detection, low false-alarm rates, and stability under changing conditions. Recognizing faults at a stage the autoencoder can enable maintenance, lessening downtime and averting expensive machine breakdowns. This utilization has the potential to notably boost the efficiency of manufacturing operations. In practice, three sensing paradigms dominate: acoustic, vibration, and force/torque/power signals. Acoustic and vibration pipelines often depend on high-rate sensors and time – frequency transforms, whereas power-based monitoring leverages signals already available on most machines. The experimental results across multiple architectures and training scenarios in this study provide several meaningful insights into the effectiveness of deep autoencoder models for fault detection in drilling operations.

6.1. Model performance and comparison

The results in Figures 6–8 indicate that hybrid models, particularly CNN-BiLSTM and WN-BiLSTM, consistently outperform simpler architectures such as LSTM and BiLSTM in terms of accuracy, F1-score, and recall. The CNN-BiLSTM model showed superior performance for smaller window sizes (e.g. 32 and 64), making it suitable for early-stage fault detection with limited data. In contrast, the WN-BiLSTM model excelled with longer input sequences (window size = 86), suggesting its capacity to capture long-range temporal dependencies more effectively.

6.2. Comparison with traditional fault detection approaches

Traditional fault detection approaches in drilling machines (including the one which was used to be utilized in MARPOSS – ARTIS) involve threshold-based monitoring of variables such as spindle power or current. These methods typically require extensive technician experience, especially in the face of unseen fault types or varying process conditions. In contrast, the autoencoder-based approach in this study is fully data-driven and unsupervised. It requires only normal operating data for training and can generalize to previously unseen fault events. This is particularly advantageous in industrial settings, where labeled fault data are scarce, and faults are rare or unpredictable. Moreover, the CNN and WaveNet components allow the model to learn both local signal fluctuations and longer-term operational trends directly from raw time-series input, reducing dependency on manual signal processing.

6.3. Comparison with state of the art

A wide range of recent research has explored fault detection and drill condition monitoring using deep learning, often relying on vibration or acoustic signals and supervised learning frameworks. This section compares our proposed unsupervised spindle power – based autoencoder models with state-of-the-art methods across several data modalities and architectures.

Although studies using acoustic signals achieved reasonable accuracy, their approaches rely on high-sample-rate audio acquisition, manual calibration, and sound-specific hardware (Tran, Truong Pham, and Lundgren 2022; X. Zhang et al. 2024). In contrast, our method leverages spindle power signals already available in most drilling machines, enabling seamless integration into existing monitoring infrastructure with no additional sensing setup.

Vibration- and force-based methods have demonstrated strong classification accuracy using supervised learning or CNN-based architectures (Dai and Huang 2024; Rajakannu et al. 2024; Scheffel, Augusto Fröhlich, and Silvestri 2021; Senjoba et al. 2025; Yu et al. 2022; Zhu, Yan, and Lu 2020). However, they typically require complex preprocessing, handcrafted feature extraction, or multi-sensor configurations. These approaches, while effective, involve significant configuration and dependence on labeled fault data.

In contrast, our unsupervised hybrid autoencoder models (CNN-BiLSTM and WaveNet-BiLSTM) operate directly on raw spindle power signals and achieved F1-scores up to 1.0 using as little as 5–10% of normal operation data – with no need for fault annotations or handcrafted features. The simplicity of deployment, statistical robustness via MAD-based Z-score thresholding, and adaptability to unseen fault patterns make our method particularly suited for real-time, scalable monitoring in modern manufacturing.

6.4. Implications for manufacturing practice

The proposed approach has direct applicability to smart manufacturing environments, particularly under Industry 4.0 paradigms. The ability to achieve high fault detection accuracy using only a small amount of normal data (as little as 5–10%) implies that the system can be deployed quickly, even in data-sparse scenarios such as new production lines or machines. Furthermore, since the models were trained on spindle power – a variable readily available on most drilling machines – the method is highly scalable and non-intrusive. The architecture can be integrated into existing monitoring systems such as ARTIS’s (MARPOSS) GEMTP platform, enabling real-time fault detection and predictive maintenance. This can help reduce unscheduled downtime, extend tool life, and lower overall maintenance costs.

7. Conclusion

This study presented a deep learning-based fault detection framework for drilling operations using autoencoders trained on spindle power signals. By exploring multiple architectures – including NN, CNN, LSTM, BiLSTM, CNN-BiLSTM, and WaveNet-BiLSTM – the research demonstrated:

- The effectiveness of unsupervised models in detecting machine faults using only normal operating data.
- Two hybrid autoencoder models (CNN-BiLSTM and WaveNet-BiLSTM) which combine local and long-range temporal modeling, outperforming single autoencoder models.
- The impact of time window size and training data proportion on model performance, identifying CNN-BiLSTM as optimal for smaller windows and WaveNet-BiLSTM for longer sequences.
- A robust MAD-based dynamic threshold to adapt sensitivity across operating regimes.
- The proposed model requires only spindle power data, making it non-intrusive and easily deployable in industrial settings using existing sensor infrastructure.
- High fault detection accuracy with as little as 5–10% of total normal-operation dataset (for training) can be achieved which enables rapid deployment in production environments with limited historical data, supporting cost-effective integration with existing monitoring systems.

7.1. Limitations

While the use of hybrid deep autoencoders for fault detection in drilling and rotating equipment is promising, this study highlights several limitations. Notably, the models' dependence on high-quality training data presents challenges, particularly in environments where data may be noisy or scarce. Although the models' prediction latency is acceptable, the computational demands of training may limit their applicability in resource-constrained settings. Scaling these models to different types of tools, machinery, or operational environments would require retraining, further increasing deployment complexity.

Deploying these models in environments with limited computational resources (e.g. older or lower-powered industrial hardware) may necessitate specialized hardware such as GPUs or TPUs, which is not always feasible. Moreover, this study focused solely on the discrimination between normal and faulty operations, without addressing the classification of different fault types that may occur during drilling processes. An inherent challenge of deep learning models – particularly autoencoders – is their 'black-box' nature, which complicates the interpretability of decisions. This lack of transparency may limit adoption, especially in safety-critical industrial applications where stakeholders require a clear understanding of the decision-making process.

7.2. Future works

The challenges outlined above emphasize the importance of optimizing models not only for accuracy but also for speed, scalability, and interpretability to enhance their practical viability in intelligent manufacturing systems. Future studies could focus on expanding the dataset to include a wider range of drilling scenarios, machine types, and materials, thereby assessing model robustness and generalization capabilities.

Additionally, efforts could be directed toward integrating fault-type classification, where an anomaly is first detected and subsequently categorized into specific fault types. Hybrid approaches that combine data-driven models with physics-based representations (e.g. analytical models of cutting force or torque) could also be explored to improve interpretability and provide actionable insights into fault causes, thereby enhancing trust and adoption in industrial settings.

Furthermore, the development of advanced autoencoder architectures optimized for faster training and reduced computational demands should be considered, making the deployment of these models more practical across diverse operational conditions. Finally, integrating autoencoder-based fault detection with real-time monitoring systems could greatly enhance practicality, enabling automated, continuous fault detection and paving the way for more efficient and resilient industrial operations.

Author contributions

CRediT: **Amin Karimi Dastgerdi:** Conceptualization, Data curation, Formal analysis, Methodology, Software, Visualization, Writing – original draft; **Paolo Mercorelli:** Conceptualization, Project administration, Supervision, Validation, Writing – review & editing; **Hamidreza Nemati:** Formal analysis, Methodology, Project administration, Supervision, Validation, Writing – review & editing.

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

Abdullahi, A., N. A. Samsudin, M. R. Ibrahim, M. S. Aripin, S. K. A. Khalid, and Z. A. Othman. 2020. "Towards IR4.0 Implementation in E-Manufacturing: Artificial Intelligence Application in Steel Plate Fault Detection." *Indonesian Journal of Electrical Engineering and Computer Science* 20 (1): 430–436. https://doi.org/10.11591/ijeecs.v20.i1.pp430-436.

Abid, A., M. T. Khan, and J. Iqbal. 2021. "A Review on Fault Detection and Diagnosis Techniques: Basics and Beyond." *Artificial Intelligence Review* 54 (5): 3639–3664. https://doi.org/10.1007/s10462-020-09934-2.

Achouch, M., M. Dimitrova, K. Ziane, S. S. Karganroudi, R. Dhouib, H. Ibrahim, and M. Adda. 2022. "On Predictive Maintenance in Industry 4.0: Overview, Models, and Challenges." *Applied Sciences* 12 (16): 8081. https://doi.org/10.3390/app12168081.

Amruthnath, N., and T. Gupta. 2018. "A Research Study on Unsupervised Machine Learning Algorithms for Early Fault Detection in Predictive Maintenance." *Paper presented at the 2018 5th international conference on industrial engineering and applications (ICIEA)*, Singapore, Singapore.

Bahdanau, D., K. Cho, and Y. Bengio. 2014. "Neural Machine Translation by Jointly Learning to Align and Translate." *arXiv Preprint arXiv: 1409.0473*. https://doi.org/10.48550/arXiv.1409.0473.

Cacciarelli, D., and M. Kulahci. 2022. "A Novel Fault Detection and Diagnosis Approach Based on Orthogonal Autoencoders." *Computers & Chemical Engineering* 163:107853. https://doi.org/10.1016/j.compchemeng.2022.107853.

Cao, R., and A. Yunusa-Kaltungo. 2021. "An Automated Data Fusion-Based Gear Faults Classification Framework in Rotating Machines." *Sensors* 21 (9): 2957. https://doi.org/10.3390/s21092957.

Chen, Z., A. Mauricio, W. Li, and K. Gryllias. 2020. "A Deep Learning Method for Bearing Fault Diagnosis Based on

Cyclic Spectral Coherence and Convolutional Neural Networks." *Mechanical Systems and Signal Processing* 140:106683. https://doi.org/10.1016/j.ymssp.2020.106683.

Christoph, M. 2020. *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. Leanpub.

Dai, M., and Z. Huang. 2024. "Research on Fault Diagnosis of Drilling Pump Fluid End Based on Time-Frequency Analysis and Convolutional Neural Network." *Processes* 12 (9): 1929. https://doi.org/10.3390/pr12091929.

Dastgerdi, A. K., D. Lange, and P. Mercorelli. 2022. "Faulty Process Detection Using Machine Learning Techniques." *Paper presented at the Congress on Smart Computing Technologies*, New Delhi, India.

Deng, A., and B. Hooi. 2021. "Graph Neural Network-Based Anomaly Detection in Multivariate Time Series." *Paper presented at the Proceedings of the AAAI conference on artificial intelligence*, Virtual conference.

Dumoulin, V., I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. 2016. "Adversarially Learned Inference." *arXiv Preprint arXiv: 1606.00704*. https://doi.org/10.48550/arXiv.1606.00704.

Fan, Y., C. Zhang, Y. Xue, J. Wang, and F. Gu. 2020. "A Bearing Fault Diagnosis Using a Support Vector Machine Optimised by the Self-Regulating Particle Swarm." *Shock and Vibration* 2020 (1): 9096852. https://doi.org/10.1155/2020/9096852.

Finke, T., M. Krämer, A. Morandini, A. Mück, and I. Oleksiyuk. 2021. "Autoencoders for Unsupervised Anomaly Detection in High Energy Physics." *Journal of High Energy Physics* 2021 (6): 1–32. https://doi.org/10.1007/JHEP06(2021)161.

Gharghory, S. M. 2021. "A Hybrid Model of Bidirectional Long-Short Term Memory and CNN for Multivariate Time Series Classification of Remote Sensing Data." *Journal of Computer Science* 17 (9): 789–802. https://doi.org/10.3844/jcssp.2021.789.802.

Ghosh, S., and V. Kataria. 2020. "Multivariate Time Series Unsupervised Anomaly Detection and Diagnosis in 5G Networks. *Technical Disclosure Commons*. Defensive Publications Series, Art. 3873. https://www.tdcommons.org/dpubs_series/3873

Gong, D., L. Liu, V. Le, B. Saha, M. Reda Mansour, S. Venkatesh, and A. van den Hengel. 2019. "Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection." *Paper presented at the Proceedings of the IEEE/CVF international conference on computer vision*, Seoul, Korea.

Hernandez-Matamoros, A., K. Sugawara, T. Kaneko, R. Wada, and M. Ozaki. 2023. "Semi-Supervised Approach for Early Stuck Sign Detection in Drilling Operations." *ArXiv Preprint arXiv: 2302.11135*. https://doi.org/10.48550/arXiv.2302.11135

Hinton, G. E., and R. Zemel. 1993. "Autoencoders, Minimum Description Length and Helmholtz Free Energy. *Advances in Neural Information Processing Systems*6: 3–10. https://dl.acm.org/doi/10.5555/2987189.2987190.

Hoang, D. T., X. T. Tran, M. Van, and H. J. Kang. 2021. "A Deep Neural Network-Based Feature Fusion for Bearing Fault Diagnosis." *Sensors* 21 (1): 244. https://doi.org/10.3390/s21010244.

Huang, X., X. Huang, B. Wang, and Z. Xie. 2020. "Fault Diagnosis of Transformer Based on Modified Grey Wolf Optimization Algorithm and Support Vector Machine." *IEEJ Transactions on Electrical and Electronic Engineering* 15 (3): 409–417. https://doi.org/10.1002/tee.23069.

Hundman, K., V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom. 2018. "Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding." *Paper presented at the Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, London, United Kingdom.

Iannace, G., G. Ciaburro, and A. Trematerra. 2019. "Fault Diagnosis for UAV Blades Using Artificial Neural Network." *Robotics* 8 (3): 59. https://doi.org/10.3390/robotics8030059.

Jang, K., S. Hong, M. Kim, J. Na, and I. Moon. 2021. "Adversarial Autoencoder Based Feature Learning for Fault Detection in Industrial Processes." *IEEE Transactions on Industrial Informatics* 18 (2): 827–834. https://doi.org/10.1109/TII.2021.3078414.

Jiang, G., P. Xie, H. He, and J. Yan. 2017. "Wind Turbine Fault Detection Using a Denoising Autoencoder with Temporal Information." *IEEE/ASME Transactions on Mechatronics* 23 (1): 89–100. https://doi.org/10.1109/TMECH.2017.2759301.

Kingma, D. P., and M. Welling. 2013. "Auto-Encoding Variational Bayes." *ArXiv Preprint ArXiv: 1312.6114*. https://doi.org/10.48550/arXiv.1312.6114.

Kumbhar, M. F. 2023. "Image Denoising Using Autoencoders." In *Artificial Intelligence and Knowledge Processing*, edited by K. Hemachandran, R. Villamarin Rodriguez, U. Subramaniam, and V. E. Balas, 137–144. CRC Press.

Lawal, A., S. Rehman, L. M. Alhems, and M. Mahbub Alam. 2021. "Wind Speed Prediction Using Hybrid 1D CNN and BLSTM Network." *IEEE Access* 9:156672–156679. https://doi.org/10.1109/ACCESS.2021.3129883.

Lee, H. 2010. *Unsupervised Feature Learning via Sparse Hierarchical Representations*. Stanford University.

Li, H., C. Yang, B. Han, and X. Cao. 2025. "IBL-AE: An Interpretable Base Learning Autoencoder for Intelligent Fault Diagnosis of Rotating Machinery." *Knowledge-Based Systems*: 114687. https://doi.org/10.1016/j.knosys.2025.114687.

Li, X., S. Wu, X. Li, H. Yuan, and D. Zhao. 2020. "Particle Swarm Optimization-Support Vector Machine Model for Machinery Fault Diagnoses in High-Voltage Circuit Breakers." *Chinese Journal of Mechanical Engineering* 33 (1): 1–10. https://doi.org/10.1186/s10033-019-0428-5.

Liu, W., L. Yan, N. Ma, G. Wang, X. Ma, P. Liu, and R. Tang. 2024. "Unsupervised Deep Anomaly Detection for Industrial Multivariate Time Series Data." *Applied Sciences* 14 (2): 774. https://doi.org/10.3390/app14020774.

Liu, Y., Y. Yang, T. Feng, Y. Sun, and X. Zhang. 2020. "Research on Rotating Machinery Fault Diagnosis Method Based on Energy Spectrum Matrix and Adaptive Convolutional Neural Network." *Processes* 9 (1): 69. https://doi.org/10.3390/pr9010069.

Liu, Y., W. Zhao, S. Chen, and T. Ye. 2021. "Mapping Crop Rotation by Using Deeply Synergistic Optical and SAR Time Series." *Remote Sensing* 13 (20): 4160. https://doi.org/10.3390/rs13204160.

Majumdar, A. 2018. "Blind Denoising Autoencoder." *IEEE Transactions on Neural Networks and Learning Systems* 30 (1): 312–317. https://doi.org/10.1109/TNNLS.2018.2838679.

Malczewski, J., and W. Czubak. 2021. "Hybrid Convolutional Neural Networks Based Framework for Skimmed Milk Powder Price Forecasting." *Sustainability* 13 (7): 3699. https://doi.org/10.3390/su13073699.

Pan, S.-T., and H.-J. Wu. 2023. "Performance Improvement of Speech Emotion Recognition Systems by Combining 1D

CNN and LSTM with Data Augmentation." *Electronics* 12 (11): 2436. https://doi.org/10.3390/electronics12112436.

Pearson, K. 1901. "LIII. OnItempstart> Lines and Planes of Closest Fit to Systems of Points in Space." *The London, Edinburgh & Dublin Philosophical Magazine & Journal of Science* 2 (11): 559–572. https://doi.org/10.1080/14786440109462720.

Qian, J., Z. Song, Y. Yao, Z. Zhu, and X. Zhang. 2022. "A Review on Autoencoder Based Representation Learning for Fault Detection and Diagnosis in Industrial Processes." *Chemometrics and Intelligent Laboratory Systems* 231:104711. https://doi.org/10.1016/j.chemolab.2022.104711.

Rajakannu, A., K. P. Ramachandran, K. Vijayalakshmi, and A. V. Sri Rajkavin. 2024. "Condition Monitoring of CNC Drill Bit for the Manufacturing Sector Using Wavelet Analysis and Artificial Neural Network Based on Feedforward Multilayer Perceptron (MLP) *SSRG International Journal of Electrical and Electronics Engineering* 11 (12): 61–75. https://doi.org/10.14445/23488379/IJEEE-V11I12P106.

Ren, H., B. Xu, Y. Wang, C. Yi, C. Huang, X. Kou, T. Xing, M. Yang, J. Tong, and Q. Zhang. 2019. "Time-Series Anomaly Detection Service at Microsoft." *Paper presented at the Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, Anchorage, Alaska, USA.

Sakurada, M., and T. Yairi. 2014. "Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction." *Paper presented at the Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, Gold Coast, QLD, Australia.

Santo, Y., R. Immich, B. L. Dalmazo, and A. Riker. 2023. "Fault Detection on the Edge and Adaptive Communication for State of Alert in Industrial Internet of Things." *Sensors* 23 (7): 3544. https://doi.org/10.3390/s23073544.

Saurav, S., P. Malhotra, T. V. Vishnu, N. Gugulothu, L. Vig, P. Agarwal, and G. Shroff. 2018. "Online Anomaly Detection with Concept Drift Adaptation Using Recurrent Neural Networks." *Paper presented at the Proceedings of the acm india joint international conference on data science and management of data*, Goa, India.

Scheffel, R. M., A. Augusto Fröhlich, and M. Silvestri. 2021. "Automated Fault Detection for Additive Manufacturing Using Vibration Sensors." *International Journal of Computer Integrated Manufacturing* 34 (5): 500–514. https://doi.org/10.1080/0951192X.2021.1901316.

Schuster, M., and K. K. Paliwal. 1997. "Bidirectional Recurrent Neural Networks." *IEEE Transactions on Signal Processing* 45 (11): 2673–2681. https://doi.org/10.1109/78.650093.

Senjoba, L., H. Ikeda, H. Toriya, T. Adachi, and Y. Kawamura. 2025. "Utilizing Convolutional Autoencoders for Anomaly Detection and Drill Bit Damage Assessment in Rotary Percussion Drilling." *Journal of Sustainable Mining* 24 (2): 282–298. https://doi.org/10.46873/2300-3960.1456.

Skog, W. 2022. "Anomaly Detection in Predictive Maintenance." https://aaltodoc.aalto.fi/items/516f5499-5749-4abb-b511-e0357542ad9b.

Tang, P., and X. Zhang. 2022. "Features Fusion Framework for Multimodal Irregular Time-Series Events." *Paper presented at the Pacific Rim International Conference on Artificial Intelligence*, Shanghai, China.

Toma, R. N., F. Piltan, and J.-M. Kim. 2021. "A Deep Autoencoder-Based Convolution Neural Network Framework for Bearing Fault Classification in Induction Motors." *Sensors* 21 (24): 8453. https://doi.org/10.3390/s21248453.

Tran, T., N. Truong Pham, and J. Lundgren. 2022. "A Deep Learning Approach for Detecting Drill Bit Failures from a Small Sound Dataset." *Scientific Reports* 12 (1): 9623. https://doi.org/10.1038/s41598-022-13237-7.

Van, M., D. Tang Hoang, and H. Jun Kang. 2020. "Bearing Fault Diagnosis Using a Particle Swarm Optimization-Least Squares Wavelet Support Vector Machine Classifier." *Sensors* 20 (12): 3422. https://doi.org/10.3390/s20123422.

Van Den Oord, A., S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. 2016. "Wavenet: A Generative Model for Raw Audio." *arXiv Preprint arXiv: 1609.03499* 12. https://doi.org/10.48550/arXiv.1609.03499.

Vincent, P., H. Larochelle, Y. Bengio, and P.-A. Manzagol. 2008. "Extracting and Composing Robust Features with Denoising Autoencoders." *Paper presented at the Proceedings of the 25th international conference on Machine learning*, Helsinki, Finland.

Vincent, P., H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou. 2010. "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion." *The Journal of Machine Learning Research* 11 (12):3371–3408. https://doi.org/10.5555/1756006.1953039.

Wang, H., Z. Liu, and T. Ai. 2022. "Long-Range Dependencies Learning Based on Non-Local 1D-Convolutional Neural Network for Rolling Bearing Fault Diagnosis." *Journal of Dynamics, Monitoring and Diagnostics* 1 (3): 148–159. https://doi.org/10.37965/jdmd.2022.53.

Wang, J., X. Li, and Y. Zhang. 2024. "A Series Fault Arc Detection Method Based on Denoising Autoencoder and Deep Residual Network." *Frontiers in Energy Research* 12:1341281. https://doi.org/10.3389/fenrg.2024.1341281.

Wang, Q., B. Peng, P. Xie, and C. Cheng. 2023. "A Novel Data-Driven Fault Detection Method Based on Stable Kernel Representation for Dynamic Systems." *Sensors* 23 (13): 5891. https://doi.org/10.3390/s23135891.

Windmann, S., and T. Westerhold. 2022. "Online Fault Detection in Industrial Production Systems Based on a Regularized LSTM Autoencoder." Research Square. https://doi.org/10.21203/rs.3.rs-1662363/v1.

Yan, R., and Z. Zhao. 2024. *Deep Neural Networks-Enabled Intelligent Fault Diagnosis of Mechanical Systems*. CRC Press.

Yongbo, L. I., D. U. Xiaoqiang, W. A. N. Fangyi, W. A. N. G. Xianzhi, and Y. U. Huangchao. 2020. "Rotating Machinery Fault Diagnosis Based on Convolutional Neural Network and Infrared Thermal Imaging." *Chinese Journal of Aeronautics* 33 (2): 427–438.

Yu, Y., Q. Liu, B. Siew Han, and W. Zhou. 2022. "Application of Convolutional Neural Network to Defect Diagnosis of Drill Bits." *Applied Sciences* 12 (21): 10799. https://doi.org/10.3390/app122110799.

Zachariades, C., and V. Xavier. 2025. "A Review of Artificial Intelligence Techniques in Fault Diagnosis of Electric Machines." *Sensors (Basel, Switzerland)* 25 (16): 5128. https://doi.org/10.3390/s25165128.

Zhang, K., J. Chen, T. Zhang, and Z. Zhou. 2020. "A Compact Convolutional Neural Network Augmented With Multiscale Feature Extraction of Acquired Monitoring Data for Mechanical Intelligent Fault Diagnosis." *Journal of*

Manufacturing Systems 55:273–284. https://doi.org/10.1016/j.jmsy.2020.04.016.

Zhang, X., F. Wang, W. Zhao, and S. Yang. 2024. “Research on Drill Bit Sound Anomaly Detection Method Based on Deep Learning.” *Paper presented at the Ninth International Symposium on Advances in Electrical, Electronics, and Computer Engineering (ISAEECE 2024)*, Changchun, China.

Zhao, R., Z. Lei, and Z. Zhao. 2024. “Research on the Application of Deep Learning Techniques in Stock Market Prediction and Investment Decision-Making in Financial Management.” *Frontiers in Energy Research* 12:1376677. https://doi.org/10.3389/fenrg.2024.1376677.

Zhao, Y.-P., G. Huang, Q.-K. Hu, and B. Li. 2020. “An Improved Weighted One Class Support Vector Machine for Turboshaft Engine Fault Detection.” *Engineering Applications of Artificial Intelligence* 94:103796. https://doi.org/10.1016/j.engappai.2020.103796.

Zhou, Q., Y. Li, Y. Tian, and L. Jiang. 2020. “A Novel Method Based on Nonlinear Auto-Regression Neural Network and Convolutional Neural Network for Imbalanced Fault Diagnosis of Rotating Machinery.” *Measurement* 161:107880. https://doi.org/10.1016/j.measurement.2020.107880.

Zhu, Y., Q. Yan, and J. Lu. 2020. “Fault Diagnosis Method for Disc Slitting Machine Based on Wavelet Packet Transform and Support Vector Machine.” *International Journal of Computer Integrated Manufacturing* 33 (10–11): 1118–1128. https://doi.org/10.1080/0951192X.2020.1795927.