

Deep Learning Techniques for Automated Engineering Fault Detection

Dr. Ghasem Alipoor

Assistant Professor, Hamedan University of Technology, Iran

Abstract

Engineering systems are becoming increasingly complex, interconnected, and dependent on continuous operational performance, creating a growing need for reliable and automated fault detection mechanisms. Conventional fault detection approaches often depend on predefined thresholds, manually engineered features, physical inspections, or expert interpretation, which may be insufficient for identifying subtle and evolving faults in complex technical environments. Deep learning provides an alternative approach by enabling computational models to automatically learn hierarchical representations from large volumes of sensor measurements, images, vibration signals, acoustic data, and operational records. This paper examines the application of deep learning techniques for automated engineering fault detection, with particular attention to convolutional neural networks, recurrent neural networks, autoencoders, long short-term memory models, and hybrid architectures. A conceptual analytical methodology is employed to examine how deep learning supports data preprocessing, feature representation, fault classification, anomaly detection, and predictive maintenance workflows.

The analysis indicates that deep learning can improve automated fault identification by learning complex nonlinear relationships and temporal or spatial patterns that may be difficult to capture through conventional statistical techniques. However, model effectiveness depends strongly on data quality, fault-label availability, class imbalance, sensor reliability, computational resources, and the ability to generalize across different operating conditions. The study further examines challenges involving interpretability, false alarms, domain shifts, cybersecurity, model deployment, and computational complexity. Future research opportunities include explainable deep learning, self-supervised learning, multimodal fault diagnosis, edge-based inference, digital twins, federated learning, and adaptive models capable of continuously responding to changing engineering conditions. The study concludes that deep learning can provide a powerful foundation for automated engineering fault detection when model development is integrated with reliable sensing, domain knowledge, robust validation, and appropriate deployment strategies.

Keywords: Deep Learning; Fault Detection; Engineering Systems; Automated Diagnosis; Predictive Maintenance; Anomaly Detection; Convolutional Neural Networks; Recurrent Neural Networks; Industrial Intelligence; Condition Monitoring.

1. Introduction

Modern engineering systems operate under increasingly demanding conditions involving high utilization, complex interactions, variable operating environments, and large quantities of continuously generated data. Industrial machinery, electrical equipment, manufacturing systems, transportation

infrastructure, and energy assets rely on interconnected components whose failure can reduce productivity, increase maintenance costs, and create safety risks. Traditional fault detection methods often depend on predefined thresholds, mathematical models, scheduled inspections, or expert interpretation. Although these approaches remain useful, they may have difficulty identifying complex fault signatures that develop gradually or occur under changing operating conditions.

The development of industrial sensing technologies has created new opportunities for data-driven fault detection. Sensors can continuously capture vibration, temperature, pressure, acoustic emissions, current, voltage, rotational speed, and other operational parameters. These data streams contain information about the condition of engineering components and can potentially reveal abnormal behavior before complete failure occurs. However, the increasing volume and complexity of sensor data create challenges for conventional analytical methods. Deep learning is particularly relevant because it can learn nonlinear representations directly from high-dimensional engineering data.

Deep learning techniques can support both supervised and unsupervised fault detection. In supervised applications, models are trained using labeled examples representing normal and faulty operating conditions. Classification architectures can then identify fault categories when new measurements are received. In unsupervised or semi-supervised settings, models can learn representations of normal behavior and detect deviations as anomalies. Autoencoders, recurrent architectures, and other representation-learning techniques are therefore valuable for engineering environments where comprehensive fault labels are difficult to obtain.

The temporal characteristics of engineering data also make deep learning useful for fault detection. Equipment behavior often changes over time, and early-stage faults may appear as subtle temporal patterns rather than isolated abnormal measurements. Recurrent neural networks and long short-term memory architectures can model sequential dependencies, while convolutional neural networks can extract patterns from transformed signals, images, or spatial sensor arrangements. Hybrid models can combine multiple forms of information to improve fault recognition under complex operational conditions.

Despite the potential of deep learning, automated fault detection cannot be treated as a purely computational problem. Engineering environments involve sensor noise, operating variability, component aging, changing loads, limited fault examples, and differences between laboratory and real-world conditions. A model that performs well under one operating condition may generate false alarms or miss faults when the environment changes. Consequently, reliable fault detection requires careful data preparation, model validation, domain knowledge, interpretability, and continuous monitoring. This paper examines these factors and evaluates the role of deep learning in developing robust automated engineering fault detection systems.

2. Objectives

The primary objective of this study is to examine the application of deep learning techniques for automated fault detection in engineering systems. The study evaluates how neural architectures can process engineering data, identify abnormal patterns, classify fault conditions, and support condition-monitoring activities while considering reliability, computational efficiency, and practical deployment requirements.

2.1 Evaluation of Deep Learning-Based Fault Detection

The first objective is to examine the capabilities of deep learning models for identifying complex engineering faults. Convolutional neural networks, recurrent neural networks, long short-term memory networks, and autoencoders are considered according to their ability to learn spatial, temporal, and nonlinear patterns from engineering data.

The objective also includes assessing how automated feature learning can reduce dependence on manually designed diagnostic indicators. Traditional fault detection often requires experts to identify relevant frequency-domain, time-domain, or statistical features. Deep learning can potentially learn these representations directly from raw or minimally processed measurements.

2.2 Assessment of Detection Reliability and Deployment

The second objective is to assess the factors affecting the reliability of deep learning-based fault detection. Data quality, class imbalance, sensor noise, operating variability, model generalization, and false-alarm rates are considered important factors influencing practical performance.

The study also evaluates deployment requirements for automated engineering environments. Computational resources, inference latency, edge processing, model updating, interpretability, and cybersecurity must be considered when deep learning systems are integrated with operational engineering infrastructure. Effective deployment therefore requires coordination between machine learning models and engineering systems.

3. Literature Review

3.1 Deep Learning for Condition Monitoring

Deep learning has become an important approach for condition monitoring because it can learn hierarchical representations from complex engineering signals. LeCun et al. established the broader significance of deep learning for representation learning, while Goodfellow et al. provided a comprehensive foundation for deep neural architectures and optimization. These developments created opportunities for applying neural models to engineering diagnostics.

In condition-monitoring applications, deep learning can process vibration signals, acoustic measurements, electrical signals, thermal images, and other sensor data. CNN-based approaches have demonstrated particular usefulness when engineering signals are transformed into representations such as spectrograms or time-frequency maps. This enables models originally designed for image processing to identify structural patterns associated with equipment faults.

3.2 Temporal Deep Learning for Fault Diagnosis

Many engineering faults evolve over time, making sequential modeling important for automated detection. Recurrent neural networks can process sequences of measurements and learn relationships between previous and current operating states. LSTM architectures were developed to address limitations associated with learning long-term dependencies in recurrent networks and have subsequently been applied to various predictive and diagnostic tasks.

Temporal deep learning models can be particularly valuable when early-stage degradation produces weak signals that become more pronounced over time. Instead of analyzing each measurement independently, sequential architectures can identify changes in system behavior across multiple time steps. However, they require sufficient and representative sequential data for reliable model training.

3.3 Autoencoders and Anomaly Detection

Autoencoders provide an alternative strategy for engineering fault detection when labeled fault data are limited. These models learn to reconstruct input data and can identify abnormal observations when reconstruction errors become significantly different from those associated with normal operating behavior. This approach is attractive for industrial environments where actual failures are relatively rare.

Research on deep autoencoder-based diagnosis has shown that representation learning can support automated extraction of condition-related information. Nevertheless, defining suitable anomaly thresholds remains challenging because normal engineering behavior may vary substantially with load, temperature, speed, and other operating parameters. Effective anomaly detection therefore requires careful modeling of operational variability.

4. Methodology

4.1 Research Design

This study adopts a conceptual analytical research design for examining deep learning techniques used in automated engineering fault detection. The methodology integrates established research on deep neural networks, condition monitoring, anomaly detection, signal processing, and predictive maintenance. The focus is placed on identifying the relationships between model architectures, engineering data characteristics, fault-detection performance, and practical deployment requirements.

The research considers fault detection as a multi-stage process involving data acquisition, preprocessing, representation learning, model training, fault identification, and decision support. Both supervised and unsupervised approaches are considered because engineering environments differ substantially in the availability of labeled fault records. This framework enables comparison of different deep learning strategies without restricting the analysis to a particular engineering domain.

4.2 Engineering Data Preparation

Engineering fault detection depends heavily on the quality of the data used for model development. Sensor measurements may contain noise, missing observations, outliers, sampling inconsistencies, and variations caused by changing operating conditions. Data preprocessing may therefore involve filtering, normalization, segmentation, synchronization, and transformation into suitable model inputs. Time-domain and frequency-domain representations can be used depending on the nature of the engineering signal.

The methodology also considers data partitioning and validation. Training, validation, and testing datasets should represent different operating states and fault conditions to reduce the risk of misleading performance estimates. Particular attention is required for class imbalance because normal operating records are often much more numerous than actual fault observations. Appropriate sampling or weighting strategies can improve model learning under imbalanced conditions.

4.3 Deep Learning Model Assessment

The study assesses several major deep learning architectures according to their suitability for engineering fault detection. CNNs are considered for spatial and local feature extraction, recurrent models for sequential information, LSTM networks for long-term dependencies, and autoencoders for anomaly detection.

Key assessment dimensions include:

- Fault classification capability
- Anomaly detection performance
- Temporal pattern recognition
- Automated feature extraction
- Model generalization
- Computational requirements
- Detection latency
- Interpretability
- Robustness to sensor noise
- Deployment flexibility

4.4 Performance and Deployment Evaluation

Model evaluation should consider multiple performance measures rather than accuracy alone. Precision, recall, F1-score, false-positive rate, false-negative rate, and detection latency can provide a more comprehensive understanding of fault-detection effectiveness. This is especially important when missed faults have considerably greater consequences than false alarms.

Deployment evaluation considers whether trained models can operate within practical engineering environments. Cloud-based inference can provide substantial computational capacity, while edge deployment can reduce communication latency and support real-time monitoring. The methodology therefore treats computational architecture as an important factor in determining the practical effectiveness of deep learning fault detection.

5. Analysis**5.1 Deep Learning Architectures for Fault Detection**

Different deep learning architectures provide different capabilities for processing engineering information. CNNs can identify localized patterns within signal representations, while recurrent models can analyze sequential relationships. LSTM networks provide improved capabilities for long-duration dependencies, and autoencoders can identify deviations from learned normal patterns.

The selection of architecture should therefore be based on the characteristics of the engineering data rather than general model popularity. Systems with strongly temporal behavior may benefit from recurrent architectures, whereas image-based inspection tasks can be better suited to CNNs. Hybrid architectures may provide advantages when both spatial and temporal information are important.

Table 1: Deep Learning Architectures for Automated Engineering Fault Detection

Deep Learning Technique	Primary Data Type	Main Detection Capability	Typical Application
CNN	Images and transformed signals	Spatial/local pattern recognition	Visual and vibration diagnosis
RNN	Sequential sensor	Temporal pattern	Dynamic equipment

Deep Learning Technique	Primary Data Type	Main Detection Capability	Typical Application
	data	learning	monitoring
LSTM	Long sequential records	Long-term dependency detection	Degradation monitoring
Autoencoder	Normal operating data	Anomaly detection	Unknown fault identification
CNN-LSTM	Spatial-temporal data	Combined feature learning	Complex machinery diagnosis
Deep Belief Network	High-dimensional features	Representation learning	Equipment classification
Transformer	Long-range sequences	Global dependency modeling	Large-scale sensor analytics

Interpretation: Table 1 demonstrates that no single deep learning architecture is universally optimal for engineering fault detection. CNNs are useful for spatial and signal-pattern recognition, while recurrent architectures are better suited to sequential measurements. Autoencoders are particularly useful when fault labels are limited. Hybrid and transformer-based architectures can integrate broader information but may require greater computational resources and larger training datasets.

5.2 Fault Detection Performance

The effectiveness of an automated fault detection system depends on its ability to distinguish genuine abnormal conditions from normal operational variability. High accuracy alone may be misleading when normal samples dominate the dataset. Precision and recall are therefore important for evaluating whether models correctly identify faults without generating excessive false alarms.

Model robustness is equally important. Engineering systems frequently operate under different loads, temperatures, speeds, and environmental conditions. A model trained under narrow conditions may not generalize effectively to new situations. Domain adaptation, data augmentation, transfer learning, and continuous model validation can help address these issues.

Table 2: Major Factors Influencing Deep Learning Fault Detection

Factor	Influence on Detection	Recommended Consideration
Data Quality	Determines reliability of learned patterns	Robust preprocessing
Fault Data Availability	Affects supervised model learning	Semi-supervised approaches
Class Imbalance	Can bias model predictions	Balanced training strategies
Sensor Noise	May obscure fault signatures	Filtering and denoising

Factor	Influence on Detection	Recommended Consideration
Operating Variability	Influences model generalization	Diverse training conditions
Model Complexity	Affects computational demand	Architecture optimization
False Alarms	Increase maintenance burden	Precision-oriented evaluation
Detection Latency	Affects real-time response	Edge or optimized inference
Interpretability	Influences engineering trust	Explainable AI methods

Interpretation: Table 2 indicates that model architecture is only one part of automated fault detection. Data quality, operating variability, fault availability, computational requirements, and interpretability can significantly influence practical performance. A technically sophisticated model may still provide unreliable results if trained on unrepresentative data. Therefore, engineering fault detection should be designed as an integrated data, model, and deployment system.

5.3 Automated Diagnostic Decision Support

Deep learning can extend fault detection beyond simple abnormality identification by supporting automated diagnostic classification. Once a fault pattern is recognized, the system can associate the detected condition with a likely component or failure category. This information can support maintenance prioritization and reduce the time required for manual diagnosis.

Integration with maintenance management systems can further increase operational value. Fault detection outputs can trigger inspection requests, maintenance alerts, or additional data collection. However, automated decisions should incorporate appropriate confidence measures because incorrect diagnoses can result in unnecessary maintenance or missed failures.

6. Discussion

6.1 Automated Feature Learning

One of the major strengths of deep learning is its capacity to learn useful representations directly from engineering data. Conventional diagnostic approaches often depend on domain experts to select signal characteristics manually. Deep neural networks can instead identify hierarchical patterns through training, potentially capturing subtle relationships that are difficult to represent using manually engineered features.

However, automated feature learning does not eliminate the importance of engineering knowledge. Domain expertise remains necessary for selecting meaningful measurements, defining operational conditions, validating results, and interpreting model behavior.

6.2 Generalization Across Operating Conditions

A significant challenge in engineering fault detection is maintaining model performance when operational conditions change. Machines may operate at different speeds, loads, temperatures, or environmental conditions, and these variations can produce patterns that resemble faults. Deep learning

models trained on limited operating states may therefore generate false alarms or fail to identify genuine faults. Robust training datasets, transfer learning, domain adaptation, and continual learning can improve generalization. Validation under realistic operating conditions should remain an essential part of model development.

6.3 Integration with Engineering Maintenance

Deep learning fault detection can contribute to predictive maintenance by identifying abnormal patterns before equipment failure occurs. Early detection can enable maintenance teams to schedule inspections and interventions more effectively. Nevertheless, model outputs should not automatically replace engineering judgment in safety-critical environments.

A practical system should combine machine learning predictions with operational thresholds, historical maintenance information, physical knowledge, and expert validation. Such integration can improve trust and reduce the consequences of incorrect model predictions.

7. Challenges

7.1 Data Availability and Quality

Obtaining sufficiently representative fault data remains one of the most significant challenges. Real equipment failures may be rare, expensive, or unsafe to reproduce intentionally. As a result, datasets can contain large amounts of normal operational data but relatively few examples of actual faults.

Sensor failures and inconsistent measurement conditions can further reduce data quality. Data augmentation, simulation, transfer learning, and semi-supervised approaches may help address limited fault records, but their effectiveness must be validated against real engineering conditions.

7.2 Model Interpretability and Trust

Deep learning models can contain large numbers of parameters and may produce predictions that are difficult to explain. Engineering personnel may hesitate to rely on automated fault predictions when the reasoning behind an alert is unclear. Explainability is particularly important in safety-sensitive applications where decisions can affect equipment operation or human safety.

Interpretable feature representations, visualization techniques, attention mechanisms, and explainable AI methods can improve confidence, but explanations must remain technically meaningful rather than merely presenting simplified model outputs.

7.3 Deployment and Computational Constraints

Complex deep learning models can require significant computational resources, especially during training and sometimes during real-time inference. Industrial environments may also contain devices with limited processing capabilities or unreliable network connectivity. Cloud deployment provides scalable computing resources but may introduce latency and connectivity dependencies.

Edge inference can reduce response time but requires model compression and hardware optimization. Consequently, deployment architecture must be selected according to the application's latency, reliability, and computational requirements.

8. Future Research

8.1 Explainable and Trustworthy Deep Learning

Future research should focus on developing explainable deep learning models that provide technically meaningful evidence for detected faults. Instead of reporting only a fault category or probability score, models could identify the sensor patterns, time intervals, or equipment components contributing to a prediction.

Such explanations could help engineers validate model outputs and distinguish genuine faults from abnormal but acceptable operating conditions. Research should also investigate uncertainty estimation and confidence-aware decision mechanisms for safety-critical engineering applications.

8.2 Multimodal and Self-Supervised Learning

Engineering systems increasingly generate multiple forms of data, including vibration, temperature, acoustic, visual, electrical, and operational measurements. Future research can investigate multimodal deep learning architectures capable of combining these heterogeneous data sources.

Self-supervised learning is also promising because it can exploit large quantities of unlabeled engineering data before fine-tuning models with limited fault records. These approaches may reduce dependence on expensive manual labeling while improving representation quality and fault-detection robustness.

8.3 Edge Intelligence and Digital Twins

Future fault detection systems may increasingly combine deep learning with edge computing and digital twin technologies. Edge-based models can perform real-time analysis close to engineering equipment, reducing communication delays and enabling faster responses.

Digital twins can provide virtual representations of physical systems that support simulation and comparison between expected and observed behavior. Integrating these technologies could enable adaptive fault detection systems that continuously compare real equipment states with digitally modeled operating conditions.

9. Limitations

This study is based primarily on conceptual analysis and established research rather than a newly collected engineering fault dataset or experimental implementation. Therefore, the discussion identifies potential capabilities and challenges of deep learning fault detection but does not provide direct experimental evidence for the performance of a particular neural architecture under a specific engineering environment.

The effectiveness of deep learning models can vary substantially according to sensor configuration, equipment type, fault severity, operating conditions, dataset size, and model architecture. Consequently, the analytical findings should not be interpreted as evidence that one technique will consistently outperform all alternatives. Experimental benchmarking across multiple engineering domains is required for stronger generalization.

10. Conclusion

Deep learning provides a powerful approach for automating fault detection across increasingly complex engineering systems. CNNs, recurrent networks, LSTM models, autoencoders, hybrid architectures, and

emerging transformer-based models can learn meaningful patterns from sensor measurements, images, vibration signals, and operational sequences. Their ability to perform automated representation learning can reduce dependence on manually engineered diagnostic features and support earlier identification of abnormal equipment behavior.

The analysis demonstrates that reliable fault detection requires more than selecting an advanced neural architecture. Data quality, fault availability, class imbalance, sensor noise, operating variability, computational requirements, and model interpretability all influence practical performance. Engineering organizations should therefore develop fault detection solutions through integrated data management, model validation, domain expertise, and deployment planning. Particular attention should be given to false negatives and false positives because both can produce operational and economic consequences.

Future engineering fault detection systems are expected to become increasingly intelligent, distributed, and adaptive. Explainable AI, self-supervised learning, multimodal sensing, edge intelligence, federated learning, and digital twins can strengthen the ability of automated systems to recognize changing fault patterns. When these technologies are combined with reliable engineering knowledge and appropriate validation processes, deep learning can become an important foundation for safer, more efficient, and more proactive engineering maintenance and monitoring.

References

1. Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., & Nandi, A. K. (2020). Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mechanical Systems and Signal Processing*, 138, 106587. <https://doi.org/10.1016/j.ymssp.2019.106587>
2. Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, 115, 213–237. <https://doi.org/10.1016/j.ymssp.2018.05.050>
3. Zhang, W., Li, C., Peng, G., Chen, Y., & Zhang, Z. (2018). A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load. *Mechanical Systems and Signal Processing*, 100, 439–453. <https://doi.org/10.1016/j.ymssp.2017.06.022>
4. Wen, L., Li, X., Gao, L., & Zhang, Y. (2018). A new convolutional neural network-based data-driven fault diagnosis method. *IEEE Transactions on Industrial Electronics*, 65(7), 5990–5998. <https://doi.org/10.1109/TIE.2017.2774777>
5. Ince, T., Kiranyaz, S., Eren, L., Askar, M., & Gabbouj, M. (2016). Real-time motor fault detection by 1-D convolutional neural networks. *IEEE Transactions on Industrial Electronics*, 63(11), 7067–7075. <https://doi.org/10.1109/TIE.2016.2582729>
6. Janssens, O., Slavkovikj, V., Vervisch, B., Stockman, K., Locufier, M., Verstockt, S., Van de Walle, R., & Van Hoecke, S. (2016). Convolutional neural network based fault detection for rotating machinery. *Journal of Sound and Vibration*, 377, 331–345. <https://doi.org/10.1016/j.jsv.2016.05.027>
7. Chen, Z., Li, W., & Pu, X. (2019). A deep learning approach for intelligent fault diagnosis of rotating machinery using vibration signals. *IEEE Access*, 7, 153898–153909.
8. Zhao, B., Zhang, X., Li, H., & Yang, J. (2020). Intelligent fault diagnosis of rotating machinery based on deep learning and transfer learning. *IEEE Access*, 8, 172132–172143.

9. Zhang, S., Su, L., Gu, J., Li, K., Zhou, L., & Pecht, M. (2023). Rotating machinery fault detection and diagnosis based on deep domain adaptation: A survey. *Chinese Journal of Aeronautics*, 36(1), 45–74. <https://doi.org/10.1016/j.cja.2021.10.006>
10. Tama, B. A., Vania, M., Lee, S., & Lim, S. (2023). Recent advances in the application of deep learning for fault diagnosis of rotating machinery using vibration signals. *Artificial Intelligence Review*, 56(5), 4667–4709. <https://doi.org/10.1007/s10462-022-10293-3>
11. Zhu, Z., Lei, Y., Qi, G., Chai, Y., Mazur, N., An, Y., & Huang, X. (2023). A review of the application of deep learning in intelligent fault diagnosis of rotating machinery. *Measurement*, 206, 112346. <https://doi.org/10.1016/j.measurement.2022.112346>
12. Das, O., Das, D. B., & Birant, D. (2023). Machine learning for fault analysis in rotating machinery: A comprehensive review. *Heliyon*, 9(6), e17584. <https://doi.org/10.1016/j.heliyon.2023.e17584>
13. Lei, Y., Li, N., Guo, L., Li, N., Yan, T., & Lin, J. (2018). A deep learning-based method for machinery health monitoring with big data. *IEEE Transactions on Industrial Informatics*, 14(5), 1876–1886. <https://doi.org/10.1109/TII.2017.2774128>
14. Eren, L., Ince, T., & Kiranyaz, S. (2019). A generic intelligent bearing fault diagnosis system using a single 1-D convolutional neural network. *IEEE Access*, 7, 2181–2191. <https://doi.org/10.1109/ACCESS.2018.2882029>
15. Li, X., Zhang, W., Ding, Q., & Sun, J.-Q. (2018). Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation. *Journal of Intelligent Manufacturing*, 29, 1–15.
16. Yang, B., Lei, Y., Jia, F., & Xing, S. (2020). A transfer learning-based deep learning method for machinery fault diagnosis under variable operating conditions. *Mechanical Systems and Signal Processing*, 139, 106625.
17. Su, H., Xiang, L., & Hu, A. (2024). Application of deep learning to fault diagnosis of rotating machineries. *Measurement Science and Technology*, 35(4), 042003. <https://doi.org/10.1088/1361-6501/ad1e20>
18. Misbah, I., Lee, C. K. M., & Keung, K. L. (2024). Fault diagnosis in rotating machines based on transfer learning: Literature review. *Knowledge-Based Systems*, 283, 111158. <https://doi.org/10.1016/j.knosys.2023.111158>
19. Matania, O., Dattner, I., Bortman, J., Kenett, R. S., & Parmet, Y. (2024). A systematic literature review of deep learning for vibration-based fault diagnosis of critical rotating machinery: Limitations and challenges. *Journal of Sound and Vibration*, 578, 118562. <https://doi.org/10.1016/j.jsv.2024.118562>
20. Neupane, D., Bouadjenek, M. R., Dazeley, R., & Aryal, S. (2024). Data-driven machinery fault detection: A comprehensive review. *arXiv*. <https://arxiv.org/abs/2405.18843>