**A Comparative Analysis of Machine Learning Algorithms for Predictive Maintenance.**

**ABHISHEK BAJAJ**

ASSISTANT PROFESSOR

ST. ANDREW’S INSTITUTE OF TECHNOLOGY AND MANAGEMENT, GURUGRAM.

abhishek.bajaj80@yahoo.com

ORCID iD: 0009-0004-9320-3356

**Abstract:**

Predictive Maintenance (PdM) has gained prominence as an intelligent maintenance strategy that leverages data-driven insights to anticipate equipment failures before they occur. With the advent of Industry 4.0 and the proliferation of Industrial Internet of Things (IIoT) technologies, Machine Learning(ML) has become a cornerstone for implementing PdM systems. This conceptual paper provides a comparative analysis of major ML algorithms commonly employed in PdM, including Decision Trees, Support Vector Machines, Artificial Neural Networks, K-Nearest Neighbors, and Ensemble Learning methods such as Random Forest and Gradient Boosting. The paper systematically compares these algorithms based on their theoretical strengths, interpretability, computational complexity, data requirements, and suitability for various industrial settings. Emphasis is placed on understanding the contexts in which each algorithm performs optimally, highlighting issues such as overfitting, model explainability, and scalability. The analysis also touches upon the integration of ML models with sensor data platforms and real-time monitoring systems. By presenting a structured conceptual comparison, this paper aims to guide decision-makers, researchers, and practitioners in selecting appropriate ML tools tailored to their maintenance environments. The study contributes to the growing body of knowledge in maintenance informatics and supports the strategic deployment of intelligent maintenance frameworks across industries.

**Keywords**: Predictive Maintenance (PdM), Machine Learning, Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN), Random Forest (RF).

**1. Introduction**

Predictive Maintenance (PdM) signifies a strategic shift from conventional reactive and time-based preventive maintenance toward condition-based, data-driven upkeep of industrial assets (Zhu et al. 2019). Enabled by the advent of Industry 4.0 and the Industrial Internet of Things (IIoT), PdM harnesses continuously collected sensor data such as vibration, temperature, and acoustic signals to predict equipment health and remaining useful life (RUL), thereby minimizing unexpected downtime and improving operational efficiency (Ucar et al. 2024; Cummins et al. 2024).

ML lies at the heart of modern PdM systems due to its ability to model complex, nonlinear patterns and generalize across diverse failure modes. Supervised techniques such as Support Vector Machines (SVM), Random Forests (RF), and Neural Networks have consistently demonstrated strong performance in fault detection and prognostics tasks (Maheshwari et al. 2024; Efeoglu & Tuna 2022). For instance, Efeoglu and Tuna (2022) reported that SVM with tailored kernel functions achieved up to 99 % accuracy on UCI PdM datasets, underscoring the method’s efficacy in high-dimensional feature spaces.

However, the effective deployment of ML in PdM faces several critical challenges. Real-world sensor data are often noisy, imbalanced, and scarce issues that degrade predictive reliability (Hansen et al. 2024; Assagaf et al. 2023). Deployment in industrial contexts also introduces requirements for real-time inference, high interpretability, and trustworthiness, especially in safety-critical settings (Cummins et al. 2024; Pashami et al. 2023). This has spurred research in explainable AI (XAI) within PdM, exemplified by frameworks assessing methods like SHAP, LIME, and Anchor across models such as RF, XGBoost, and feed-forward neural networks (Cummins et al. 2024; Pashami et al. 2023).

This conceptual paper offers a structured, theory-driven comparison of core ML algorithms Decision Trees, SVM, k-Nearest Neighbors, Artificial Neural Networks, Random Forest, and Gradient Boosting in the context of PdM. The comparative framework targets dimensions such as theoretical foundations, scalability, interpretability, data requirements, and system integration. By highlighting algorithmic strengths, trade-offs, and practical alignment with different maintenance environments, it aims to support researchers and practitioners in making informed decisions when adopting ML-driven PdM strategies.

**2. Overview of Predictive Maintenance**

Predictive Maintenance (PdM) is a proactive strategy that monitors the real-time condition of machinery using sensor data and analytics to forecast failures and schedule maintenance only when needed (Goriveau et al., 2016; WorkTrek Glossary 2025). By anticipating equipment degradation, PdM reduces downtime, lowers repair costs, extends asset lifespan, and provides optimal scheduling that maximizes productivity and safety (IBM, 2023). Maintenance methodologies have evolved significantly over time:

* Corrective (Reactive) Maintenance: Repairs are performed only post-failure, leading to unplanned downtime and costly disruptions.
* Preventive Maintenance: Scheduled at regular intervals to mitigate failure risk, but often results in unnecessary part replacements or maintenance when conditions are still optimal (Zhu et al., 2019).
* Predictive Maintenance: Activated when data-driven insights indicate impending failure, ensuring maintenance is both timely and necessary. This approach represents a natural progression in the industry 4.0 era toward smarter, more efficient maintenance (Zhu et al., 2019; Goriveau et al., 2016; WorkTrek Glossary 2025).

***Components of a PdM System***

A modern PdM system integrates several key elements:

* Data Acquisition: IoT-enabled sensors capture critical parameters—vibration, temperature, pressure, and acoustic signals in real time (Zhu et al., 2019; Susto, 2016).
* Data Processing & Analysis: Data pipelines clean, aggregate, and extract meaningful features. ML and statistical tools detect anomalies and estimate Remaining Useful Life (RUL) of components (Zhu et al., 2019; Susto, 2016).
* Decision Support & Maintenance Scheduling: The system translates analytics into actionable maintenance tasks alerting operators, triggering work orders, and optimizing parts inventory and labor allocation (Zhu et al., 2019; WorkTrek Glossary 2025).

**3. Key ML Algorithms for Predictive Maintenance**

***Decision Trees (DT)***

Decision trees use hierarchical, rule-based logic to split data into subsets based on feature values offering high interpretability and fast training times. Each decision node divides data through simple binary questions, and leaves represent final predictions. They require minimal data preprocessing, can handle both categorical and numerical features, and allow easy visualization of decision rules. In PdM, DTs are frequently used to classify equipment conditions and escalate maintenance alerts efficiently with transparent logic (Kaparthi & Bumblauskas 2020)

***Support Vector Machines (SVM)***

SVMs find the maximum-margin hyperplane that separates classes and can also leverage kernel functions to handle non-linear data in high-dimensional spaces. They excel with high-dimensional inputs, such as sensor arrays, and often generalize well with limited training samples IBM notes that SVMs tend to overfit less than decision trees and perform effectively with unstructured or high-dimensional data, although they may be computationally intensive (IBM 2024)

**Artificial Neural Networks (ANN)**

ANNs consist of interconnected layers of artificial neurons capable of learning complex, non-linear relationships making them well-suited for modeling intricate sensor data patterns. They exhibit strong performance and adaptability but are often criticized as “black-box” models with limited interpretability. This opacity hinders trust in safety-critical PdM applications unless explainability techniques (e.g., SHAP, LIME) are integrated (Cummins et al. 2024)

**K‑Nearest Neighbors (KNN)**

KNN is an instance-based, non-parametric algorithm that classifies new samples based on the majority label of their K nearest neighbors in feature space. It is simple, easy to understand, and requires no training per se. However, its performance suffers from high memory usage and latency during prediction, posing challenges for real-time PdM. KNN is most effective for small to medium datasets with low dimensionality (Vithi & Chibaya 2024).

**Random Forest (RF)**

Random Forest is an ensemble method that builds multiple decision trees using bootstrap sampling and random feature selection. This approach enhances accuracy and reduces overfitting. RFs are parallelizable and robust, blending interpretability (through feature importance measures) with strong predictive performance, making them highly suitable for real-world PdM (Kaparthi & Bumblauskas 2020)

**Gradient Boosting Machines (GBM)**

GBMs such as XGBoost or LightGBM build models sequentially where each new tree attempts to correct errors from prior ones. This “boosting” process yields high predictive accuracy, especially in complex datasets. However, GBMs are computationally demanding, require careful hyperparameter tuning, and are less interpretable without supplementary explainability frameworks (Cummins et al. 2024).

**Table 1- Summary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Interpretability | Complexity | Strengths | Drawbacks |
| DT | High | Low | Fast, clear logic | Prone to overfitting |
| SVM | Medium | Medium–High | Generalizes well, works with small data | Expensive training |
| ANN | Low | High | Models complex patterns | Black-box, needs trust |
| KNN | High | Low training / High inference | Simple, intuitive | Slow at scale |
| RF | Medium | Medium | Accurate, resistant to overfitting | Model size increases |
| GBM | Low | High | State-of-the-art performance | Harder to tune, interpret |

This conceptual comparison provides clarity on each algorithm's trade-offs. In the next section, we will analyze them in terms of scalability, data requirements, and suitability across different PdM environments.

**4. Comparative Analysis Framework**

***Theoretical Foundations***

Decision Trees (DT): Learn via greedy splits on features to maximize purity (e.g. Gini index); effectively model non-linear decision boundaries with simple logic.

* SVM: Optimize a maximum-margin hyperplane; kernel functions enable mapping to high-dimensional feature spaces.
* ANN: Learn hierarchical nonlinear representations through backpropagation and gradient descent across multiple layers.
* KNN: Lazy, instance-based learning; predictions depend on proximity in feature space to labeled examples.
* RF: Ensemble of decision trees trained on bootstrap samples and random feature subsets; reduce variance via aggregation.
* GBM: Sequentially build additive decision trees, correcting previous errors via gradient descent; powerful but complex. Insights derived from broad ML theory and PdM review studies

***Interpretability and Transparency***

* High Interpretability: DT and KNN – DTs visualize decision paths; KNN provides exemplar cases.
* Medium: SVM – support vectors hint at decision boundaries; linear kernels retain clarity.
* Lower: RF offers feature importance summaries but lacks simple decision logic.
* Low (Black-Box): ANN and GBM – necessitate XAI tools (SHAP, LIME) for insight, especially in safety-critical applications.

***Scalability and Complexity***

* Low Complexity: DT, KNN – manageable memory and training requirements, though KNN struggles in high-dimensional feature space at inference time.
* Medium: RF – parallelizable training; handles large datasets efficiently.
* High: SVM with non-linear kernels, GBM, and ANN – require intensive computation and memory, and careful hyperparameter tuning.

***Data Requirements***

* Light Requirements: DT, KNN, SVM perform acceptably on smaller, curated datasets.
* Heavy Requirements: RF benefits from larger datasets for better generalization; GBM and ANN require substantial data volumes to mitigate overfitting and achieve stable training.
* Impact of Data Quality: All methods suffer from noisy, imbalanced, or sparse data; ensemble and deep models show slightly greater noise resilience

***Real-time Applicability***

* Real-time Ready: DT and RF – fast inference, suitable for edge devices.
* Moderately Real-time: SVM (with optimized kernels) can manage real-time if tuned; KNN is expensive at inference.
* Less Suitable: GBM and ANN require high computational capacity and often run offline or via cloud systems.

***Maintenance Use Case Suitability***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case | DT | SVM | KNN | ANN | RF | GBM |
| Condition Monitoring (classifying normal vs. faulty signals) | ✅ | ✅ | ✅ | ✅ | ✅ | ✅ |
| Anomaly Detection (detecting unusual patterns) | ✅ | ✅ | ✅ | ✅ | ✅ | ✅ |
| RUL Estimation (estimating remaining life) | ❌ | ✅ | ❌ | ✅ | ✅ | ✅ |

DT/KNN: Effective for simpler tasks like binary fault classification.

SVM: Well-suited for classification and RUL with moderate data volumes.

ANN/RF/GBM: Flexible for all use-cases, especially capable of RUL estimation in complex sensor environments

***Framework Summary***

* Simplicity & Transparency: DT, KNN – minimal setup, high interpretability, suited to basic PdM tasks.
* Balanced Performance: SVM, RF – good trade-off of accuracy, interpretability, scalability; well-suited for real-time inference.
* High Performance (at cost): ANN, GBM – best accuracy and handling of complex tasks like RUL; require robust infrastructure and XAI support.

This conceptual framework guides algorithm selection tailored to PdM contexts—with machine selection depending on resource availability, task complexity, and transparency needs.

**5. Discussion**

In evaluating the strengths and weaknesses of each ML algorithm within predictive maintenance (PdM) scenarios, several nuanced trade‑offs emerge. **Decision Trees** offer highly interpretable, white‑box decision rules and require minimal preprocessing, contributing to rapid deployment in simple PdM tasks. However, they are prone to overfitting and structural instability, where slight data perturbations can drastically alter the model (GeeksforGeeks, 2025) In contrast, **Support Vector Machines (SVM)** excel at handling high-dimensional feature spaces and generalize robustly even with limited data. Yet, this strength comes at the expense of high computational load and complex kernel tuning, which impedes real-time deployment (N‑iX, 2023; Cureus, 2025) . Artificial Neural Networks (ANN) can model intricate non-linear sensor relationships and achieve top performance in tasks like RUL estimation. Nevertheless, they are often criticized for being “black-box” systems that demand extensive data and hyperparameter tuning, reducing transparency. K‑Nearest Neighbors (KNN) is straightforward and interpretable, making it suitable for small-scale, labelled datasets. However, inference time and memory usage grow prohibitively with dataset size, limiting its applicability in scalable, real-time PdM (Nature, 2024) The Random Forest (RF) ensemble method balances interpretability and predictive power, offering robust performance even with noisy inputs. Nonetheless, its interpretability is less intuitive, although feature‑importance metrics partially mitigate this concern (ScienceXcel, 2025). Finally, Gradient Boosting Machines (GBM) provide state‑of‑the‑art accuracy in complex failure detection and RUL forecasting, but they are resource-intensive and opaque without XAI tools

These findings emphasize critical trade-offs: simpler models (DT, KNN) afford speed and transparency at the cost of performance, while more complex models (ANN, GBM) demand higher computational resources and risk opaqueness. SVM and RF often emerge as balanced choices suitable for regulated industries requiring both reliability and interpretability. Yet, challenges persist in actual deployment: data quality remains a hurdle, with sparse failure events and noisy sensor streams complicating model training even though ensemble and neural methods show some resilience. Integration into legacy systems necessitates bridging interoperability gaps across outdated platforms. Moreover, workforce readiness and resistance to algorithmic systems underscore the need for augmented explainable solutions, ideally supported by conversational AI agents. Resource constraints pose further limitations; deep-learning models often require investments unlikely to be justified for smaller operations. Lastly, in high-stakes industries such as aviation or healthcare regulatory and safety demands dictate that black-box methods must be complemented with XAI techniques (e.g., SHAP, LIME) to achieve transparency and compliance.

**6. Conclusion**

This conceptual study provides a comparative analysis of prominent ML algorithms applied to Predictive Maintenance (PdM), aiming to assist researchers and practitioners in making informed decisions aligned with industrial priorities. The six algorithms reviewed Decision Trees (DT), Support Vector Machines (SVM), Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN), Random Forest (RF), and Gradient Boosting Machines (GBM) demonstrate distinct advantages and limitations when evaluated across interpretability, scalability, real-time applicability, and data demands. DT and KNN stand out for simplicity and transparency, making them suitable for quick deployments and technician-friendly systems, although they lack robustness in complex or high-dimensional scenarios. SVM offers strong generalization but struggles with scalability. RF provides a good balance of accuracy and interpretability, while GBM and ANN are top performers in predictive accuracy, particularly for tasks like Remaining Useful Life (RUL) estimation, albeit at the cost of explainability and computational overhead.

Future research should focus on developing hybrid ML systems that combine interpretability and predictive power, such as integrating rule-based models with deep learning. Emphasis should also be placed on real-time learning, explainable AI (XAI), and transfer learning to enhance adaptability across equipment types and sectors. As industries move toward Industry 5.0, combining human-centric AI with autonomous PdM systems will be a key frontier

**Recommendations:**

* For resource-constrained environments (e.g., small-scale manufacturers), DT and KNN offer easy deployment and fast inference.
* In regulated industries (e.g., healthcare, aviation), RF or SVM with explainability tools (like SHAP or LIME) ensure compliance and trust.
* For data-rich and performance-critical applications (e.g., smart factories, energy grids), GBM and ANN provide superior results, particularly when integrated with cloud-based or edge-computing platforms.

**References**

* Antoni Musril, H., Saludin, S., Firdaus, W., Usanto, S., Kundori, K., & Rahim, R. (2023). Using k-NN artificial intelligence for predictive maintenance in facility management. *International Journal of Engineering & Emerging Technology, 10*(6). https://doi.org/10.14445/23488379/IJEEE-V10I6P10
* Assagaf, I., Ga, J. L., Sukandi, A., Abdillah, A. A., & Arifin, S. (2023). *Machine Predictive Maintenance by Using Support Vector Machines*. Recent in Engineering Science and Technology.
* Benítez‑Peña, S., Blanquero, R., Carrizosa, E., & Ramírez‑Cobo, P. (2024). Cost‑sensitive probabilistic predictions for support vector machines. *European Journal of Operational Research, 314*(1), 268–279. https://doi.org/10.1016/j.ejor.2023.09.027
* Bertsimas, D., & Digalakis, V. (2023). Improving stability in decision tree models. *arXiv*. <https://arxiv.org/abs/2305.17299>
* Chen, T. (2021). Predictive maintenance method of equipment based on machine learning. *LinkedIn Pulse*. <https://www.linkedin.com/pulse/predictive-maintenance-method-equipment-based-machine-learning-chen/>
* Cummins, L., Sommers, A., Ramezani, S. B., et al. (2024). *Explainable Predictive Maintenance: A Survey of Current Methods, Challenges and Opportunities*. arXiv. [arxiv.org+3arxiv.org+3arxiv.org+3](https://arxiv.org/abs/2401.07871?utm_source=chatgpt.com)
* Efeoğlu, E., & Tuna, G. (2022). *ML for Predictive Maintenance: Support Vector Machines and Different Kernel Functions*. Journal of Machinery Manufacture and Reliability, 51(5), 447–456. [researchgate.net](https://www.researchgate.net/publication/364289083_Machine_Learning_for_Predictive_Maintenance_Support_Vector_Machines_and_Different_Kernel_Functions?utm_source=chatgpt.com)
* Geeks for Geeks. (2023). Feature importance with random forests. https://www.geeksforgeeks.org/feature-importance-with-random-forests/
* Geeks for Geeks. (2023). Overfitting in decision tree models. https://www.geeksforgeeks.org/overfitting-in-decision-tree-models/
* Goriveau, R., Medjaher, K., & Zerhouni, N. (2016). *Predictive maintenance: monitoring and prognostics*. ISTE Ltd & John Wiley & Sons. [worktrek.com](https://worktrek.com/glossary/predictive-maintenance-2/?utm_source=chatgpt.com)[arxiv.org+4arxiv.org+4researchgate.net+4](https://arxiv.org/html/1912.07383v2?utm_source=chatgpt.com)[brightlysoftware.com+3en.wikipedia.org+3arxiv.org+3](https://en.wikipedia.org/wiki/Predictive_maintenance?utm_source=chatgpt.com)
* Hansen, H. H., Kulahci, M., & Nielsen, B. F. (2024). *A primer on predictive maintenance: Potential benefits and practical challenges*. Quality Engineering. [bohrium.dp.tech](https://bohrium.dp.tech/paper/arxiv/864950044715909264?utm_source=chatgpt.com)
* IBM (2023). *What is Predictive Maintenance?* IBM Think. [ibm.com](https://www.ibm.com/think/topics/predictive-maintenance?utm_source=chatgpt.com)
* IBM. (2023). Support vector machines (SVM). *IBM Think*. https://www.ibm.com/cloud/learn/svm
* IBM. (2024). *Support vector machines*. IBM Think. [en.wikipedia.org+3ibm.com+3en.wikipedia.org+3](https://www.ibm.com/think/topics/support-vector-machine?utm_source=chatgpt.com)[mdpi.com](https://www.mdpi.com/2813-2203/3/4/28?utm_source=chatgpt.com)[en.wikipedia.org+3mdpi.com+3mdpi.com+3](https://www.mdpi.com/2076-3417/14/2/898?utm_source=chatgpt.com)[medium.com+2researchgate.net+2mdpi.com+2](https://www.researchgate.net/publication/353786557_Support_Vector_Machines_in_Big_Data_Classification_A_Systematic_Literature_Review?utm_source=chatgpt.com)
* Kaparthi, S., & Bumblauskas, D. (2020). *Designing predictive maintenance systems using decision tree‑based MLtechniques*. *International Journal of Quality & Reliability Management*. [researchgate.net](https://www.researchgate.net/publication/339361879_Designing_predictive_maintenance_systems_using_decision_tree-based_machine_learning_techniques?utm_source=chatgpt.com)
* Leiva, L. A., Anta, A. F., Mancuso, V., & Casari, P. (2019). A novel hyperparameter-free approach to decision tree construction that avoids overfitting by design. *arXiv*. <https://arxiv.org/abs/1906.11006>
* Maheshwari, S., Tiwari, S., Rai, S., & Singh, S. V. D. P. (2024). *Comprehensive Study Of Predictive Maintenance In Industries Using Classification Models And LSTM Model*. arXiv. [scholar.its.ac.id+8arxiv.org+8mdpi.com+8](https://arxiv.org/abs/2403.10259?utm_source=chatgpt.com)
* Medium. (2024a). Explaining feature importance by example of a random forest. <https://medium.com>
* Medium. (2024b). Unlocking the power of support vector machines (SVMs). <https://medium.com>
* Meegle. (2025a). Fine-tuning for support vector machines. <https://www.meegle.com>
* Meegle. (2025b). Neural networks in transportation. <https://www.meegle.com>
* Neptune.ai. (2021). The KNN algorithm – Explanation, opportunities, limitations. https://neptune.ai/blog/knn-algorithm
* N-iX. (2023). High-dimensional data classification and feature selection using support vector machines. *ScienceDirect*. <https://www.sciencedirect.com>
* Pagano, D. (2023). A predictive maintenance model using Long Short-Term Memory neural networks and Bayesian inference. *Decision Analytics Journal, 6*(7), 100174. https://doi.org/10.1016/j.dajour.2023.100174 [sciencedirect.com](https://www.sciencedirect.com/science/article/pii/S235291482100174X?utm_source=chatgpt.com)
* Pashami, S., Nowaczyk, S., Fan, Y., Jakubowski, J., et al. (2023). *Explainable Predictive Maintenance*. arXiv. [arxiv.org+1arxiv.org+1](https://arxiv.org/abs/2306.05120?utm_source=chatgpt.com)
* Prabhudesai, S. (2024). Explainable predictive maintenance of rotating machines using LIME, SHAP, PDP, ICE. *ResearchGate*. <https://www.researchgate.net/publication/369509874>
* StackOverflow. (2016). Feature importance of the predicted value in random forest. <https://stackoverflow.com/questions/31355059>
* Susto, G. A. (2016). *Predictive maintenance: implementation and trends*. Proceedings in… Journal.
* SynchroNet. (2025). What is SVM? Understanding support vector machines. <https://synchronet.net>
* Taşcı, B., Omar, A., & Ayvaz, S. (2023). Remaining useful lifetime prediction for predictive maintenance in manufacturing. *Computers & Industrial Engineering, 202*, 109566. https://doi.org/10.1016/j.cie.2023.109566
* Ucar, A., Karakose, M., & Kırımça, N. (2024). *Artificial Intelligence for Predictive Maintenance Applications: Key Components, Trustworthiness, and Future Trends*. Applied Sciences, 14(2), 898. [mdpi.com](https://www.mdpi.com/2076-3417/14/2/898?utm_source=chatgpt.com)
* Uçar, A., Karaköse, M., & Kırımça, N. (2024). Artificial intelligence for predictive maintenance applications: Key components, trustworthiness, and future trends. *Applied Sciences, 14*(2), 898. https://doi.org/10.3390/app14020898
* Vithi, N. L., & Chibaya, C. (2024). *Advancements in Predictive Maintenance: A Bibliometric Review of Diagnostic Models Using MLTechniques*. *Analytics*, 3(4), 493–507. [mdpi.com](https://www.mdpi.com/2813-2203/3/4/28?utm_source=chatgpt.com)
* Wikipedia contributors. (2025). *Decision tree learning*. Wikipedia. [en.wikipedia.org](https://en.wikipedia.org/wiki/Decision_tree_learning?utm_source=chatgpt.com)
* WorkTrek Glossary. (2025). *Predictive Maintenance,* [journals.sagepub.com+14worktrek.com+14researchgate.net+14](https://worktrek.com/glossary/predictive-maintenance-2/?utm_source=chatgpt.com)
* Zhu, T., Ran, Y., Zhou, X., & Wen, Y. (2019). *A Survey of Predictive Maintenance: Systems, Purposes and Approaches*. arXiv. [researchgate.net+15arxiv.org+15researchgate.net+15](https://arxiv.org/abs/1912.07383?utm_source=chatgpt.com)