POLITECNICO DI TORINO
Repository ISTITUZIONALE

Toward Autonomous LLM-Based AI Agents for Predictive Maintenance: State of the Art, Challenges, and Future Perspectives

Original
Toward Autonomous LLM-Based AI Agents for Predictive Maintenance: State of the Art, Challenges, and Future Perspectives / Di Maggio, L.G.. - In: APPLIED SCIENCES. - ISSN 2076-3417. - 15:21(2025). [10.3390/app152111515]

Availability:
This version is available at: 11583/3005359 since: 2025-11-24T09:23:39Z

Publisher:
MDPI

Published
DOI:10.3390/app152111515

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

13 October 2026

applied sciences

MDPI

Review

Toward Autonomous LLM-Based AI Agents for Predictive Maintenance: State of the Art, Challenges, and Future Perspectives

Luigi Gianpio Di Maggio

Dipartimento di Ingegneria Meccanica e Aerospaziale (DIMEAS), Politecnico di Torino, Corso Duca Degli Abruzzi 24, 10129 Torino, Italy; luigi.dimaggio@polito.it

Abstract

Recent advances in Large Language Models (LLMs) enable agentic systems that combine perception, reasoning, and action across the enitre Predictive Maintenance (PdM) lifecycle, including machine fault diagnosis. However, the literature on LLM-driven agents for PdM remains fragmented and lacks a unified view on contemporary frameworks such as Model Context Procotol. This paper reviews discriminative, generative, and LLM-based approaches for PdM and consolidates fragmented evidence on LLM-driven AI agents. Namely, it introduces agentic AI concepts for PdM and develops an analysis of potential applications, challenges, and risks in light of agency theory, while mapping drivers and barriers to adoption based on recent evidence from industry analysis. Findings indicate near-term value for information and decision-support agents, while higher autonomy needs stronger governance, benchmarks, and safety evidence.

Keywords: predictive maintenance; prognostics and health management; AI agents; model context protocol; large language models; fault diagnosis; condition monitoring; industrial internet of things

check for updates

Academic Editor: Alexandru Stanciu

Received: 3 October 2025
Revised: 23 October 2025
Accepted: 24 October 2025
Published: 28 October 2025

Citation: Di Maggio, L.G. Toward Autonomous LLM-Based AI Agents for Predictive Maintenance: State of the Art, Challenges, and Future Perspectives. *Appl. Sci.* **2025**, *15*, 11515. https://doi.org/10.3390/app152111515

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Predictive maintenance (PdM) plays a key role to anticipating failures and planning interventions based on the actual condition of assets, with the aim of reducing costs and increasing plant efficiency and availability [1,2]. This approach is particularly crucial for rotating machines, which support the operations of different manufacturing sectors. The evolution of the Industrial Internet of Things (IIoT) has made continuous streams of sensory and operating data available [3,4], enabling the use of machine learning techniques for condition monitoring [5], fault diagnosis [6], and RUL estimation [2,7]. In this context, rolling bearings are often the most informative observation nodes in rotating machinery, as highlighted by recent research, for example in the case of machine learning applied to bearing diagnosis [8,9] , modeling and simulation [10,11] and the translation of these analyses into domains other than those of laboratory [12].

Over the past decade, discriminative statistical learning [13,14] and deep learning techniques, such as Convolutional Neural Networks (CNNs) [15,16], Recurrent Neural Networks (RNNs) [17,18], hybrid approaches, and physics-informed models [19], have achieved significant results in the processing of vibrational and multivariate signals and transformer architectures have expanded the ability to capture long-range dependencies in operating data [20,21]. However, critical issues remain in terms of transferability between

different operating conditions and machinery, strong dependence on data availability, and limited interpretability [15].

In recent years, Generative AI has opened new opportunities for PdM, while also helping to mitigate some of the limitations of machine learning methods. Generative models such as Generative Adversarial Networks (GANs) [22], Variational Autoencoders (VAEs), and diffusion models [23] enabled data augmentation, mitigating the scarcity of failure examples and improving anomaly detection capabilities.

More recently, Large Language Models (LLMs) [24,25] have demonstrated advanced capabilities in language modeling and generalization tasks, including few-shot scenarios [26]. By virtue of these properties, they potentially enable functions useful to PdM such as consultation and explanation in natural language, integration of knowledge bases via Retrieval Augemented Generation (RAG) [27], and automation of technical documents, with early evidence of operational benefits in real-world contexts [4,28].

However, the orchestration of heterogeneous tools, permission management, and ensuring robustness with respect to out-of-distribution OOD data require an architecture that goes beyond isolated algorithms. This results in a structural gap in the literature because, since rapid advances in LLM-based agents in other domains, the literature on autonomous agents for PdM is fragmented, lacks shared benchmarks, and does not offer a unified architectural vision calibrated to industrial maintenance workflows. At the same time, the adoption of agentic systems raises questions of governance and alignment that are largely unexplored in PdM [29,30]. Filling this gap is timely and necessary to translate model-level advances into reliable and auditable PdM solutions.

This work offers four contributions:

1. it includes a summary of the state of the art on the use of AI in PdM, divided into discriminative approaches, Generative AI, and emerging LLM-based applications;
2. it introduces for the first time in this context the definition of an architectural framework for AI agents in PdM integrating current and state-of-the-art technologies such as Model Context Protocol (MCP);
3. it provides an analysis of potential applications with an assessment of possible risks and mitigation in light of the literature on agentic systems governance [29];
4. it proposes a discussion of the drivers and barriers to real adoption, informed by economic impact evidence and recent reports on the use and adoption of AI and Generative AI in companies [31], with practical implications for a roadmap for progressive introduction.

The rest of the article is organized as follows: Section 2 reviews the current state of the art in discriminative, generative, and LLM-based AI approaches for PdM and diagnostics; Section 3 presents a detailed analysis of state-of-the-art agent architectures; Section 4 discusses potential use cases, risks, and mitigation; Section 5 analyzes current drivers and barriers to real adoption, Section 6 provides hypotheses on future perspectives and roadmaps, whereas Section 7 provides conclusions.

2. State of the Art

Most of the literature currently investigating AI, machine learning [6], and deep learning methods [15,32,33] for PdM, condition monitroing [5] and fault diagnosis focuses primarily on analyzing fault detection and RUL estimation [2,7] using data-driven approaches. For this reason, it is first introduced the status of these discriminative approaches, aimed at identifying the health status of mechanical components and systems in a fairly deterministic manner, before introducing the topic of generative AI and agentic systems. Although these latter have found little space in the literature to date, these approaches are identified as trends of interest with particularly high potential in industrial setting.

2.1. Discriminative AI for Predictive Maintenance

Modern PdM systems use Industrial IIoT infrastructure for the mass collection of sensory data, including vibration signals, temperatures, pressures, and electrical quantities. This data-driven approach enables real-time monitoring of operating conditions and more accurate fault prediction, which is particularly relevant for rotating machines that form the backbone of many plants.

The role of artificial intelligence and machine learning techniques in the maintenance of rotating machines is central to key industrial sectors such as manufacturing, oil & gas, automotive, aerospace. Unexpected failures of these assets can have serious consequences such as prolonged production downtime, safety risks, and significant economic losses.

CNNs [15] have established themselves as the benchmark architecture for fault diagnosis in rotating machines thanks to their effectiveness in processing vibration signals and automatically extracting spatial features [34,35]. Fuzzy fusion approaches that combine multiple CNNs models fed by different signal representations (e.g., frequency domain or time-frequency domain transforms) have shown superior performance, particularly in bearing anomaly detection [34]. Also, CNNs variants optimized at the hyperparameter level by means of with Bayesian techniques have achieved impressive accuracies [35] in classifying defects on rotating machine test benches.The spread of CNN-based methods is partly due to their intrinsic ability to work end-to-end without requiring manual feature extraction. Examples of improved performance are found in the field of electric machines [36].

RNNs, and in particular Long Short-Term Memory (LSTM), excel at modeling the temporal dependencies typical of multivariate sensory data [17,18]. These models are particularly well suited to both predicting RUL and recognizing sequential failure patterns. Hybrid CNN–RNN architectures that combine local feature extraction and long-range dynamic modeling have demonstrated overall superior performance in PdM applications [17]. LSTM networks are then proposed for modeling temporal dependencies in data, making them suitable for studying the evolution of equipment failures over time [37]. However, challenges remain in handling very long sequences and computational complexity for real-time applications. Differently, transformer architectures [20] capture long-range dependencies without the limitations of recurrent networks by exploiting self-attention mechanisms [21,38]. In particular, hybrid CNN–Transformer models [39] have reported very high average accuracies across multiple operating conditions, confirming the ability of attention to enhance time–frequency or directly sequential representations in complex diagnostic tasks.

Despite the increase in deep learning-related research work, traditional machine learning algorithms continue to play a significant role in PdM, especially in contexts with limited data or when interpretability is essential. Support Vector Machines (SVM) [14] remain effective in fault classification tasks, especially when combined with adequate feature extraction [13,40]. They are particularly useful when datasets are small and when, in industrial settings, it is important to be able to interpret the decision boundary.

Ensemble methods, such as Random Forest and Gradient Boosting, offer robust performance across a wide range of fault diagnoses [41]. They also provide good readability through feature importance ranking and, in scenarios with little data, tend to be less prone to overfitting than deep learning approaches [41]. Fast training times are achievable through these approaches [9].

In the context of PdM, transfer learning [42] is crucial because it allows knowledge learned in source domains to be reused to improve performance in target domains where data is scarce or distributed differently. In this direction, He et al. [43] proposes a diagnosis that combines GANs and transfer learning to extract transferable features between conditions, Guo et al. [44] introduce a deep convolutional transfer learning network with

two modules for domain adaptation that drives the network to learn invariant features. Yang et al. [12] address the lab-to-field gap by proposing the Feature-based Transfer Neural Network for extracting transferable features from vibration data of bearings in the laboratory and in actual operation.

A growing line of research explores hybrid architectures that combine the strengths of different techniques, along with multi-modal strategies that use vibrational, thermal, and electrical signals for a more comprehensive diagnosis. In this context, the integration of Physics-Informed Neural Networks (PINN) with data-driven methods represents a promising trend [19], since by incorporating physical knowledge into learning algorithms, it can improve generalization capabilities and reduce the amount of data required. Shen et al. [45] propose a physics-informed deep learning approach that combines a threshold model with a CNN and a loss function that incorporates these constraints, validated on experimental data. Qin et al. [46] address sample imbalance in bearings with data generation based on digital twins and inverse PINNs, estimating physical dynamic parameters and synthesizing multi-condition fault signals that improve cross-working-condition diagnosis. Ni et al. [47] introduce a physics-informed residual network demonstrating superiority under non-stationary regimes. Lu et al. [48] present a physics-informed feature weighting method that extracts features robust to speed variation and Jia et al. [49] propose a physics-informed unsupervised domain adaptation framework resulting in more robust cross-machine diagnostic models.

In the field of few-shot learning for rotating machinery, Yu et al. [50] introduced a prototypical network with a multiscale wavelet module, Zhang et al. [51] propose a framework for bearing diagnosis which, with few examples, learns to recognize new failure scenarios and in case studies on artificial faults, it achieves improved accuracy compared to a Siamese network and shows good robustness even on real damage. Chang and Lin [52] present meta-learning method with adaptive learning rates and revised loss functions on two bearing datasets.

Some critical issues relating to training are also analyzed by means of Federated learning [53,54] which is aimed at training models in a distributed manner without sharing sensitive data. Federated approaches for fault diagnosis have been recently proposed by Ma et al. [55] and Zhang et al. [56].

The main critical issues concern the quality of training data, which is often limited [51] and unbalanced [22], resulting in a decline in performance on OOD data and the need to resort to unsupervised strategies [49,57], self-supervised anomaly detection [58] and deep learning unsupervised approaches [1]. To mitigate the risk of unreliable decisions in the presence of distribution shift, deep ensemble approaches estimate uncertainty and implement OOD data detection with threshold criteria, improving the reliability of diagnosis on unseen samples [59]. Also, Bayesian frameworks for diagnosis separate random uncertainty from epistemic uncertainty, enabling the recognition of OOD inputs and reducing erroneous decisions in the absence of explicit signals [60]. The scarcity of high-quality, standardized public datasets hinders reproducibility and makes it difficult to verify industrial transferability. There are also persistent problems of adaptation between different machines, sites, or operating conditions, while the poor interpretability [61,62] of black box machine learning models reduces operator confidence and slows adoption in critical production contexts. The framework for this summary analysis is shown in the Table 1.

Table 1. Discriminative AI for predictive maintenance and fault diagnosis.

Topic	Capabilities	Sources
CNN-based diagnosis	• End-to-end spatial feature extraction on vibration • Fuzzy fusion of multiple CNNs (time, frequency, time–frequency) • Bayesian-optimized CNN variants • Improved performances in electric machines	[15,34–36]
RNN, LSTM and CNN–RNN hybrids	• Temporal dependency modeling for RUL and sequential patterns • CNN–RNN hybrids couple local features and long-range dynamics • LSTM suited to equipment evolution over time • Challenges with very long sequences and real-time computational cost	[17,18,37]
Transformers and CNN–Transformer hybrids	• Self-attention captures long-range dependencies without recurrence limits • Hybrid CNN–Transformer models achieve high accuracies across operating conditions	[20,21,38,39]
Traditional ML, SVM	• Effective with proper feature extraction • Useful with small datasets and interpretable decision boundaries in industrial settings	[13,14,40]
Traditional ML, Ensembles	• Random Forest and Gradient Boosting robust across diagnoses • Feature-importance interpretability • Less overfitting with limited data • Fast training achievable	[9,41]
Transfer learning for domain shift	• Reusing knowledge across machines/conditions with scarce data • GAN used for domain adaptation • Deep convolutional transfer with domain adaptation • Lab-to-field transferable features	[12,42–44]
Physics-informed and hybrid approaches	• PINNs/CNNs with physics constraints improve generalization and data efficiency • Digital Twin and inverse PINN for synthetic fault signals (cross-condition diagnosis) • Physics-informed feature weighting robust to speed variation • Physics-informed models for cross-machine robustness	[19,45–49]
Few-shot and meta-learning	• Multiscale wavelet prototypical network (few-shot, cross-component) • Model-agnostic Meta-learning few-shot for bearings • Meta-learning with adaptive learning rates and improved losses	[50–52]
Federated learning in PdM	• Distributed training without sharing sensitive data • Recent FL approaches for fault diagnosis across machines	[53–56]
Key challenges (data and interpretability)	• Limited or imbalanced training data; generalization • Need for unsupervised/self-supervised anomaly detection • Lack of standardized public datasets (reproducibility) • Domain adaptation hurdles across machines/conditions • Black-box interpretability slows adoption in critical operations	[1,22,49,51,57,58,61,62]

2.2. Generative AI for Predictive Maintenance

The existing literature on Generative AI for PdM is mainly focused on the development and application of models based on GANs, VAEs, Diffusion Models, and transformer-based architectures [4,23,63,64]. Specifically, the integration of this kind of generative AI has fostered the use of virtual replicas that can simulate various failure scenarios, improve anomaly detection capabilities, and generate synthetic datasets to address data scarcity challenges in industrial environments [65]. Generally, this approach enables real-time simulation, data augmentation, and improved fault diagnosis accuracy across manufacturing, energy, healthcare, and transportation sectors [4,23].

The problem of limited data availability, especially with regard to damage data, is addressed in particular by generating synthetic data that can compensate for the absence of real and experimental data [66]. Examples of this type are mainly available in the field of bearing diagnosis [22,65,67–70], given the presence of benchmark datasets in the literature [8,71], but also in other fields such as ship machinery monitoring [72], where recent advances including GAN-based systems integrate Failure Mode and Effect Analysis (FMEA) methodologies to provide comprehensive frameworks for both failure detection and prescriptive maintenance recommendations [73,74].

VAE are also effective as they learn compressed representations of normal conditions and identify deviations indicative of anomalies, enabling early detection and unsupervised diagnosis. The use of VAEs, on the other hand, has proven to be effective in federated learning [75,76] applications by drastically reducing communication overhead [54]. Similarly, diffusion models have shown in recent studies [77] the possibility of generating synthetic data from installed systems that have not yet experienced failure. Some application of diffusion models [78] have been directed toward increasing the explainability [62] of fault diagnosis models.

Transformer architectures [20] represent the most recent evolution in the field of generative AI and LLMs [25] are built relying on such architectures. Generic transformer architectures are also used for deep reinforcement learning aimed at predicting RUL [79]. Then, foundation models and LLMs are emerging as enabling technologies in the domain of PdM as well. For instance, a first attempt to build a foundation models for fault diagnosis by leveraging self-supervised learning and generative pre-trained transformers has been recently proposed by Wang et al. [80].

2.3. LLMs and AI Agents for Predictive Maintenance

Pre-trained generative models specific to rotating machines have shown promising accuracy in diagnostic tasks and significant results in multi-class one-shot scenarios, suggesting the possibility of generalizing across datasets that are heterogeneous in terms of signal characteristics and operating conditions [80]. LLM and transformer-based architecture applications, include the ability to generate FMEA documents using LLMs with foundational models capable of generating most of the key content [81]. Deep generative models such as LLMs have also been proposed to provide explanations in natural language by using RAG [23,82]. The report proposed by Reddicharla and Ali [28] found a 20% increase in operational productivity after adopting LLM-basesd solutions in the field of oil & gas industry. Namely, maintenance and operations engineers could receive guidance on the most appropriate steps to take to identify the root cause, drawing on manuals, previous events, and current performance and enabling LLMs them to explore intervention logs with generative support to resolve operational issues more quickly. In the same direction, MaintAGT [83] has been proposed as a multimodal model that employs a specialized LLM and a multimodal module with Chain-of-Thought (CoT) [84] for diagnosis, trained on liter-

ature, manuals, and standards. The model outperforms generalist LLMs and approaching an ISO Level III analyst.

The application of LLMs for fault diagnosis and maintenance prescriptions and recommendations is evidenced by a growing body of literature. Studies related to these approaches span various applications, including aviation, railways, and rotating machinery, where certain prescriptive capabilities are highlighted through fine-tuning and prompt engineering. In the aeronautical field, recent studies [85] use LLMs to analyze telemetry and propulsion system operating data, extracting diagnostic and maintenance insights and supporting the identification of anomalies. In the railway sector, Wang and Li [86] examine the potential applications of LLMs in Prognostics and Health Management (PHM) processes (knowledge management, condition monitoring), also highlighting their limitations. For rotating machinery, Wang et al. propose RmGPT [80], a token-based generative model for diagnosis and prognosis that prove to be effective even in few-shot scenarios. Tao et al. [87] explore the use of LLMs for bearing fault diagnosis in order to improve their generalization capabilities of diagnosis models, while Qaid et al. [88] present a framework for adapting LLMs to numerical data for fault diagnosis. From a prescriptive perspective, He et al. [89] introduce a context-aware approach that combines LLM with tool calling and RAG, approaching an agent-based paradigm. In the offshore wind sector, an LLM-based agent has also been proposed to generate safer repair recommendations [90]; in line with this, Lukens et al. [91] evaluate the performance of LLMs in similar PHM tasks.

LLMs are also integrated with multimodal inputs including sensory signals, text logs, and images to enhance fault detection and decision-making. Integration with digital twins, in particular, enables real-time monitoring and PdM, improving interpretability and reasoning capabilities in complex industrial contexts.

In line with the integration between LLM and digital twin, Sun et al. [92] propose an LLM-driven multi-agent architecture in which perceptual agents merge multimodal data and decision-making agents interact with traceable mechanisms. Also the authors validated th study with an ablation study on a maintenance scenario. Zhang et al. [93] develop a multimodal expert system for transformers, fine-tuned on visual language models, capable of real-time monitoring and generating maintenance strategies through human-machine dialogue. In the railway sector, Ferdousi et al. [94] present DefectTwin, an multimodal and multi-model pipeline that, integrated into a digital twin, analyzes visible and unseen defects from images and provides usable responses on consumer devices with good performance even in zero-shot. On the more general side of fault detection & diagnosis, Alsaif et al. [95] introduce a framework based on a large multimodal model (GPT-4-preview), with synthetic data augmentation generated by LLM and particular attention to safety aspects in industrial scenarios. Focusing on specialized visual language models for PHM tasks, Kumar et al. [96] propose Diagnostics-LLaVA to interpret images of industrial equipment and provide recommendations, showing improvements over state-of-the-art open-source models. To leverage operational knowledge in Computerized Maintenance Management Systems (CMMS), Bengtsson et al. [97] adopt a hybrid approach combining language processing, domain ontologies, and LLM to structure the natural language fields of maintenance logs, enabling near real-time analysis (e.g., detection of recurring faults) and knowledge sharing across multiple sites. Finally, Wang and Li [98] outline a solution in which an LLM is enhanced by an industry knowledge base with text embedding and vector retrieval, achieving more accurate, specific, and relevant responses to real-world cases than generalist LLMs.

The literature addresses the practical challenges of deploying LLMs in industrial contexts, including computational resource constraints, data protection, and integration across the edge–fog–cloud supply chain. Reference architectures and methodological frameworks

guide model training, evaluation, and deployment, emphasizing the role of Large Language Model Operations (LLMOps) practices and hybrid processing to ensure scalable and secure applications. In the context of the IoT, Kok et al. [99] outline the roles of LLMs along the entire edge–fog–cloud chain, highlighting how contextual reasoning enables advanced decisions and proposing a system model for IoT for condition monitoring and PdM. Mar et al. [100] introduce a reference architecture for the industrial integration of LLMs covering data preparation, training, evaluation, deployment, and prompt engineering. On the edge front, Zheng et al. [101] offer a review of the entire lifecycle of edge LLMs and on-device applications in personal, enterprise, and industrial scenarios, while Friha et al. [102] summarize LLM and Edge Intelligence architectures, compare optimization techniques for resource-constrained devices, and analyze vulnerabilities and countermeasures in depth, as well as principles of responsible development. Finally, focusing on microcontrolled platforms, Dashdamirli [103] explores LLM applications for real-time control, adaptive decisions, and sensory integration in industrial automation, discussing key implementation challenges and possible solutions.

In the field of autonomous industrial control with LLM agents, Vyas and Mercangöz [104] propose an operator–validator–reprompter agent architecture with validation and re-prompting mechanisms that allow the agent to recover from errors and adapt to unexpected disturbances. A case study of thermal control on a microcontroller illustrates its feasibility. Wang et al. [105] develop an automation system based on LLM agents by integrating LangChain [106] for real-time analysis aimed at optimizing lines, improving energy efficiency, and reducing emissions, outlining an application path in production scenarios. Xia et al. [107] present an end-to-end framework in which an agent system, structured prompting, and an event-driven information model feed LLM inference with real-time events on different semantic levels, enabling production planning and operations control.

The evidence summarized in Table 2 shows the developments in LLM-based systems for predictive and prescriptive maintenance and agentic systems. However, there are no structured studies indicating the possibilities offered by autonomous agents, whose analysis is becoming increasingly in-depth for various fields of application.

Table 2. Overview of LLMs and AI Agents in Predictive Maintenance.

Topic	Capabilities	Sources
Rotating machinery foundation model	• High diagnostic accuracy • Strong multi-class one-shot performance • Potential cross-dataset generalization	[80]
FMEA drafting with LLMs	• Automatic generation of most key FMEA content	[81]
Enterprise deployment (oil & gas)	• Reported +20% operational productivity • Root-cause guidance from manuals, past events, current performance • Interactive exploration of maintenance logs	[28]
Multimodal agent	• Specialized LLM + multimodal module with CoT • Trained on literature, manuals, standards • Outperforms generalist LLMs; near ISO Level III analyst	[83,84]
Sectoral applications: aeronautics; railways; rotating machinery	• Aircraft propulsion diagnostic insights • Railways: PHM scope and limits • Rotating machinery, bearing generalization & numeric-data adaptation	[80,85–88]

Table 2. *Cont.*

Topic	Capabilities	Sources
Context-aware, tool-calling agent (prescriptive)	• LLM, tool calling and information retrieval • Towards agent-based recommendations	[89]
Offshore wind agent; PHM evaluation	• LLM-based agent for safer repair recommendations • Evaluation of LLM performance in PHM tasks	[90,91]
Digital Twin, multimodal agents	• LLM-driven multi-agent • Multimodal expert for transformers • Digital Twin pipeline with zero-shot on device	[92–94]
Multimodal LLM framework and safety	• GPT-4–based framework • Synthetic augmentation with LLMs • Safety considerations for industrial deployment	[95]
Specialized vision language models for PHM	• Image-based diagnostics and repair recommendations	[96]
CMMS knowledge extraction; Information retreval	• Structuring free-text maintenance logs • Near real-time pattern mining • Vector-based retrieval outperform generalist LLMs	[97,98]
LLMOps and edge–fog–cloud integration	• Edge LLM lifecycle (on-device) • Edge Intelligence architectures & security • Microcontroller scenarios	[99–103]
Autonomous industrial control with LLM agents	• Operator–validator–reprompter architecture • LangChain integration • Event-driven, structured prompting for planning	[104–107]

3. From LLM Tools to Agentic Systems: Architectures, Tooling, and MCP

In order to analyze the potential offered by autonomous AI agents in the context of PdM, it is first necessary to define their structure and provide detailed information on the current technology stack on which applications can be based. This section describes this and other aspects.

An AI agent can be defined as a system or program capable of autonomously performing tasks on behalf of a user or another system, planning the workflow and using available tools to achieve set objectives in a self-determined manner [29,108–110]. Such agents may include natural language processing [25], problem solving, interaction with external environments through sensors and actuators and decision-making. Although the objective is set by humans, the agent autonomously chooses the best actions to take. According to the recent survey of Xi et al. [108], the historical roots actually stem from the field of philosophy, where, in a general sense, an agent is defined as an entity endowed with the capacity to act, while the term agency indicates the exercise or manifestation of that capacity. In a narrower sense, agency usually refers to the execution of intentional actions; consequently, the term agent describes entities that possess desires, beliefs, intentions, and the capacity to act. Nowadays, although agents are not yet fully autonomous, their development is rapidly progressing in this direction [29]. Recent studies [111] have shown that by integrating techniques such as Monte Carlo Tree Search and self-critique mechanisms, AI agents can acquire considerable autonomy and advanced reasoning in complex task environments [112,113].

Over the past two years, LLMs [25,114,115] have demonstrated remarkable emerging capabilities and gained widespread popularity, with researchers beginning to exploit them to build AI agents. In particular, LLMs are used as the main component or controller of these agents, expanding their perceptual and action capabilities through strategies such as multimodal perception and the use of tools. Thanks to techniques such as CoT [84] and problem decomposition, LLM-based agents can exhibit reasoning and planning capabilities comparable to those of symbolic agents. Given the particularly rapid developments in

the field of LLMs, the agents that are becoming most widespread nowadays are mainly LLM-based.

Among the main features of agentic systems currently being investigated and developped, it emerges that an AI agent does not simply respond, but plans the path to follow, breaking down complex objectives into smaller tasks and coordinating the use of tools and subcomponents, acquiring information and assessing whether to handle the request directly or involve a human operator. The AI agent also uses external tools, which today can consist of databases, Application Programming Interfaces (APIs), or even other agents to fill information gaps and correct its plan based on the results obtained.

Recent studies further discussed main concerns about AI agents. According to Kolt [29], AI agents inherit from LLMs already known risks such as hallucinations, bias, discrimination, production of toxic content, environmental damage, and leakage of sensitive data, but they also introduce new and more serious threats, as they are not limited to providing instructions but can perform real actions. This increases the potential for damage from cyber fraud to erroneous financial transactions, to physical damage in robotic systems. Additional risks include unwanted autonomous capabilities, behaviors emerging from interactions between agents (e.g., market collusion), and loss of control and transparency as agent networks become more complex. A central issue is the alignment problem [29,116] which means that AI agents tend to optimize the objectives assigned to them, but if these are incomplete or imprecise, they can produce undesirable effects, especially in scenarios not anticipated by designers. Important aspects are often omitted from the objective and in these cases, the system may maximize measurable elements while neglecting or even compromising those that are not measurable or difficult to quantify.

3.1. Architecture of AI Agents

The typical structure of an LLM-based AI agent is based on some fundamental components [109,110]:

- the model is the brain of the agent and is typically a LLM capable of understanding context, planning, reasoning, and making decisions. The model coordinates the workflow, decides what actions to take, and adapts to any unforeseen events;
- tools are external components, such as APIs, databases, software or hardware services, which allow data to be acquired, interact with third-party systems, and perform operations in the digital or physical world. The ability to connect to multiple tools and use them in sequence is one of the elements that differentiate AI agents from simple prompt-based language models;
- instructions define how it should behave, what tools it can use, under what conditions to interrupt an operation, and how to handle exceptions. Clear and comprehensive instructions make the agent's behavior more predictable and robust.
- memory allows the agent to retain its state and context between different execution phases. It can include short-term memory, useful for maintaining logical flow in a single session, and long-term memory, for learning from past interactions and adapting over time;
- the reasoning engine develops strategies, plans sequences of actions, and breaks down complex objectives into more manageable sub-objectives. This is typially accompanied by a reflection mechanism, whereby the agent evaluates intermediate results and corrects its strategy in the event of errors or inefficiencies;
- guardrails [110] establish operational boundaries and safety rules. Also, they prevent risky behavior (e.g., avoid sending sensitive data to unauthorized services and manage exceptions and failures by anticipating scenarios in which the agent should not act autonomously). They can be implemented at different levels. Namely, providing

initial instructions and constraints, filtering and validating the actions or responses generated or directly limiting the use of certain tools or commands.

These components can be organized into single-agent architectures, where a single agent tackles tasks sequentially, or multi-agent architectures, where the problem is divided into several parallel tasks, each handled by a separate agent with its own tools [109]. While offering considerable flexibility, this model has operational and structural limitations. As highlighted by recent studies [117,118], the use of tools is fragmented. Each agent must be configured individually to access the necessary tools, with APIs connections often managed manually and lacking common standards. Added to this is the dependence on proprietary integrations, where access to external resources and services is restricted to plugins or interfaces specific to a given platform, thus reducing the portability of the system. Security is also a critical issue, as connections to external systems increase attack surfaces and require permission controls that are often implemented on an ad hoc basis. Finally, dynamic orchestration is hindered by the absence of a standard mechanism for identifying and using tools in real time, adapting to new contexts or tasks requires manual development and configuration. These issues are further amplified as the number of tools increases or when agents operate in complex collaborative environments, such as in multi-agent architectures with shared resources.

3.2. Model Context Protocol

The critical issues of agent architectures have highlighted the need for a standardized solution that allows agents to access tools and resources in a unified manner and operate in multi-agent scenarios without redundant or incompatible integrations [117,118].

To this purpose, the MCP was introduced by Anthropic at the end of 2024 [117,118]. MCP is an open protocol which aims at standardizing the process with which applications provide context to LLMs [117,118]. MCP allows AI agents to autonomously identify, select, and orchestrate the most appropriate tools based on the operating context [118] and do not rely on predefined tool mappings. The protocol also integrates human-in-the-loop (HITL) mechanisms, allowing users to provide additional data or approve certain actions when necessary. The unification of interfaces promotes the development of AI applications, increasing their flexibility in managing complex workflows. Since its introduction, MCP has spread rapidly, going from a niche solution to a key element in the development of AI-native applications [118].

MCP adopts a three-role architecture with host, client, and server that standardizes the discovery and use of tools, data, and prompts by AI agents [117,118]. An example architecture is reported in Figure 1. The host consists in the AI application that coordinates and manages one or multiple MCP clients. The client maintains a connection to an MCP server and obtains context from an MCP server for the MCP host to use. The server exposes three main capabilities further complemented by cross-cutting utilities:

- tools that invoke of services, APIs and external operations;
- resources that can be represented by structured and unstructured data from local files, databases, or cloud platforms;
- prompts that are reusable templates and workflows that optimize responses and standardize repetitive tasks.

In the traditional tool-based model, the agent depends on the Software Development Kit (SDK) and application wrappers to invoke external tools; this involves tight coupling with the chosen framework, non-standardized permission management, and ad hoc context exchange. MCP introduces a standard interface layer between agents and resources (tools, data, extensions), explicitly separating discovery and implementation of capabilities, context exchange and prompts that can be standardized as well. This open protocol reduces

lock-in and duplication of integrations, improves observability, and makes multi-agent and multi-tool composition repeatable in heterogeneous environments.

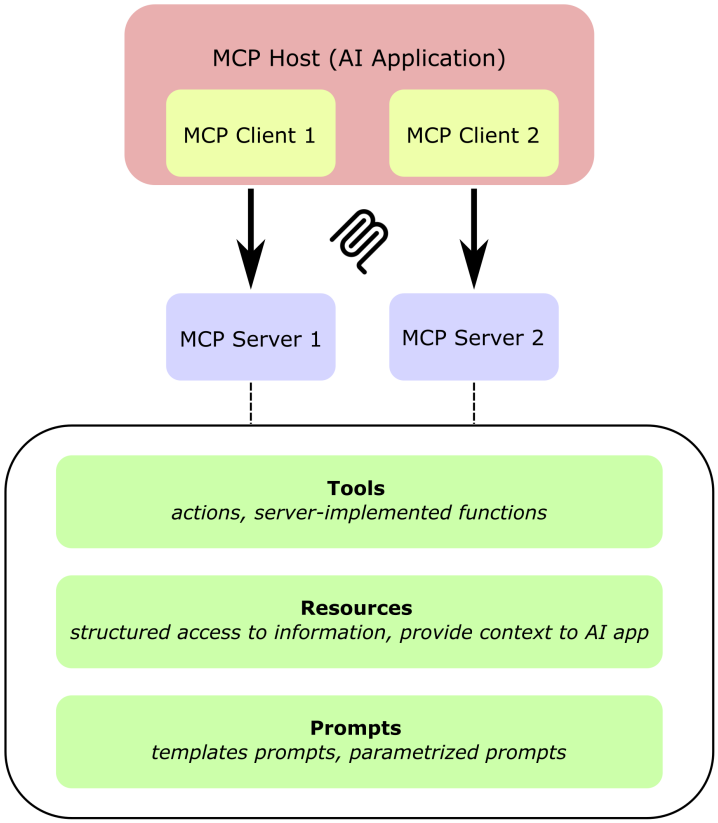

Figure 1. Model Context Protocol (MCP): example architecture.

3.3. Tools for Developing AI Agents

The current landscape of tools for developing AI agents is characterized by several solutions that provide essential infrastructure. Among them, LangChain [106] is presented as a composition framework that enables LLM applications through modular components combined in chains; in this paradigm, interaction with models occurs by directly calling APIs, without training agents from scratch. Then, AgentGym [119] is a generic interaction platform for LLM-based agents built on HTTP services, whereas AutoGen [120] provides a flexible environment in which customizable agents can collaborate with each other and with human operators and tools. AgentVerse [121] simulates human-inspired problem-solving processes, with the ability to dynamically adjust team composition. Development frameworks also include OpenAI Agents SDK, which is the official Software Development Kit for building agentic applications with a reduced set of primitives. Microsoft AutoGen [120] is an open framework for multi-agent applications, with collaboration between agents and humans, and LlamaIndex [122] supports different types of agents with tool and memory integration. Commercial platforms include Agents for Amazon Bedrock [123] for creating agents that orchestrate models, call APIs, and use knowledge bases; Google's Vertex AI Agent Builder for building and orchestrating multi-agent experiences on proprietary infrastructure; and Azure AI Foundry Agent Service, which provides a dedicated service for creating agents with integrated tools (functions, connectors, multi-agent workflows). To date, most of these tools are moving towards using MCP as the basic protocol for providing context to agents. Indeed, the author notes that several SDK ecosystems are progressively adopting MCP as a common protocol for tool access and capability discovery. As these implementations mature, the operational differences between protocol-driven and SDK-

driven integrations are likely to narrow. In this paper therefore MCP is presented as an emerging convergence layer.

4. Potential Applications, Challenges and Risks of Agentic Systems in Industrial Maintenance

In this section, the author identifies some applications in which the use of AI agent-based systems could foster technological impact. Based on existing literature and technologies currently available in the field, it is possible to identify some applications where the presence of AI agents could contribute to the automation of certain processes. In this sense, a roadmap of future development could focus on these applications. For each of those summarize in Table 3, the risks to be considered in the implementation analysis and how they could be managed are analyzed. In this sense, actions that take place in the virtual ecosystem and actions that can take place through agents in the physical environment will also be identified.

Table 3. Potential uses of AI agents in industrial maintenance.

Use Case	Inputs and Actions	Key Risks and Concerns	Mitigations and Controls
Monitoring, diagnosis and RCA	• Operating context, manuals and standards • LLM inference over records • Edge feature extraction	• Out-of-distribution interpretation • Decision opacity and scalable oversight limits • Authority creep at the edge	• Confidence thresholds and alternative hypotheses • Formal allow and deny rules; remote kill switch • MCP resources, tools and prompts to structure RCA
Spare parts management and procurement	• PdM predictions, supplier lead times, budget constraints • Draft proposals with rationale and uncertainty	• Information asymmetry • Authority creep; misaligned objectives • Opaque delegation chains	• Duty-to-inform; approval gates • Least-privilege principle (RBAC) • Delegation policy and traceable logs
CMMS orchestration and planning	• Auto pre-fill work orders • Cross-reference forecasts, availability, spares • Propose maintenance plans	• Ticket intensification; process bypass • Opacity of motivations; loyalty drift • Opaque handoffs; oversight limits	• Approval gates; RBAC • Tracked handoffs; multi-objective planning • MCP tools for CMMS; workflow prompts
PdM model life-cycle management	• Monitor in-field performance; detect shift • Orchestrate retraining and test in sandbox • Propose model promotion with evidence	• Risky promotions; limited visibility • Insufficient scalable oversight	• Mandate limits; comprehensive logging • Objective checklists; auto-block on failure • Independent auditor agent; MCP pipelines

In the maintenance field, the AI agent could interface with the existing technology stack, which may consist of sensors with edge or cloud devices, Supervisory Control and Data Acquisition (SCADA) systems, data lakes, PdM models, CMMS, and supply or procurement systems. The agent could orchestrate the entire cycle, from the signal to diagnosis, to planning, to work orders, to post-intervention feedback analysis and reporting. The author of this work highlights how the development of MCP systems could particularly facilitate the adoption of such approaches. Provided that MCP servers have been developed, agents can actually be given the ability to receive context from multiple resources and tools.

In order to conduct this analysis, the author does not go into detail about the general risks and challenges of PdM, as these are already well known in the literature and can be briefly summarized as follows:

- from the technical point of view, sensor heterogeneity, vendor lock-in, and lack of standard interfaces hinder consistent and reliable condition monitoring [124,125]; models degrade as operating conditions change, and continuous learning remains unresolved; computational limitations and latency hinder real-time analytics and streaming at scale [126]; reliable system-level prognostics remain difficult, undermining user confidence [127];
- from the point of view of data, the scarcity of failure examples, the heterogeneity of formats, and limited attention to security and privacy are weighing heavily; moreover, labeling and annotation are costly and often unfeasible [124,126]. In implementation, integration with legacy, poor interoperability between vendors, and uncertain economic benefits block adoption; there is also a lack of analytical skills, especially in small and medium-sized enterprises [125];
- operationally, ROI could be uncertain, remote monitoring costs may be high, and without clear procedures, alerts can be ignored or lead to incorrect actions [125,128];
- at the organizational level, cultural and trust issues, the need for new roles and reskilling, gaps in governance of decision-making processes, and the need for iterative learning with HITL and good interfaces are challenging factors [128,129].

4.1. Proposal of Conceptual Framework

A proposal of generic architecture of an autonomous AI agent operating in the context of PdM is summarized in the Figure 2. The figure shows the logical flow of the PdM agent. At the top, the inputs include data from IoT sensors and user prompts. In the center, an LLM assists the core agent in reasoning and tool management. On the sides are two sets. Knowledge is given by standards, manuals, spare parts database, maintenance history, and reports, whereas tools may consist of signal processing, data analysis, code execution, information retrieval, report writing, API calls. At the bottom, outputs include diagnostic reports, maintenance scheduling, updating and management of diagnostic models, and operational prescriptions. In an MCP architecture, the knowledge and inputs blocks are exposed as *resources*, the tools panel as *tools*, while the LLM and agent use *prompts* and templates to standardize the decision chain. In this sense, the agent can have access to various functions. Perception functions can translate into the acquisition of signals from sensors and the operating context, but also documented knowledge including manuals, standards, reports, and maintenance history. Decision-making functions combine data analysis, diagnosis, and prognosis through an LLM that can support structured reasoning, the extraction of salient evidence, and the use of analytical tools. The agent could also translate decisions into operational proposals, plans, and draft work orders, interfacing with CMMS and in extended configurations. Also, it could affect operating parameters and be equipped with learning capabilities, updating rules and models based on post-intervention feedback with controlled release in a test environment prior to production. The tools and knowledge bases can also be provided through specific interoperability MCP protocols. These functions naturally map onto server primitives. Resources convey data and knowledge (perception), tools perform analysis and application integration (decision and action), while prompts and templates guide reproducible decision flows.

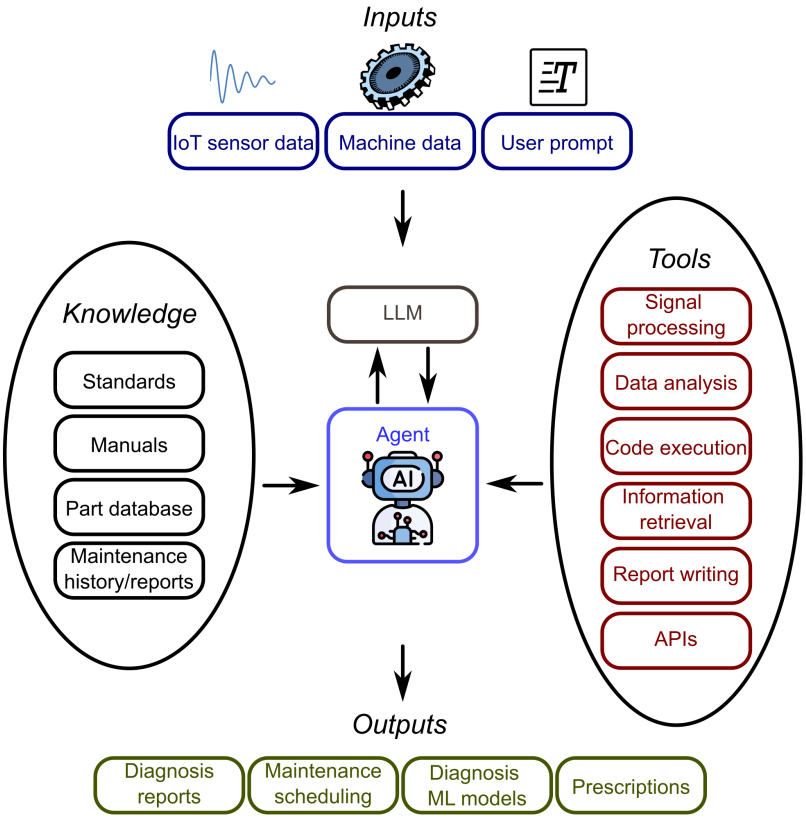

Figure 2. Proposal of generic AI agent architecture for predictive maintenance tasks.

It is also useful to distinguish between two operational profiles. In this work, soft agents are intended as the ones operating in the information domain, reading data and knowledge bases, producing diagnoses, prescriptions, plans, and reports, pre-filling work orders, and requesting approvals. Their risk profile is mainly informational (decision errors, distorted priorities, authority creep in information systems), which can be mitigated with authority limits, HITL policies, and explicit delegation rules. Authority limits refer to what the agent may decide or execute autonomously and what requires approval, whereas delegation rules refer to who delegates what to whom, including for instance audit trails. Hard agents are understood as those capable of interacting with the physical world. These are associated with additional operational security and compliance risks, which require more stringent constraints . This taxonomy sets the stage for the discussion on alignment and scalable oversight developed in the next section.

4.2. Governance: Agent Alignment and Scalable Oversight

In order to systematically identify concerns, possible implications, and risks of agentic applications, this work draws on the analysis proposed by Kolt in 2025 [29], which thoroughly examines the risks associated with current opportunities for the development of AI agent systems. In this context, this paper examine how the characteristics of the so-called alignment problem, understood as the challenge of designing agents that pursue their objectives in a reliable and secure manner, can manifest themselves in PdM applications. The literature emphasizes how principal-agent economic theory [29,30,130] and the principles of common law of agency [29] offer tools to characterize alignment more rigorously. Among those, it is possible to identify:

- information asymmetry, given by the fact that agents can access information not available to their principals placing humans in a vulnerable position;

- the issue of authority which concerns the extent of decision-making power granted to agents, how they interpret and implement instructions, and the risk of misconduct;
- the issue of loyalty understood as a duty to act in the user's interest and to seek their consent when appropriate;
- the issue of delegation given by cases in in which an agent entrusts activities to other agents (human or artificial) and the applicable rules.

These issues are addressed in the context of the common law doctrine on agency relationships [29] and in the economic theory of agency problem [30,130], also known as principial-agent problem. Recent literature [29] further discuss how some standard solutions to the principal-agent problem may be not sufficient to face alignment. Strategies for monitoring and supervising agents, in fact, conflict with scalable oversight [131], intended as the challenge of designing control methods that remain effective even when AI surpasses human supervisors in terms of capability. In other words, as the intelligence of systems grows, we must ensure that our ability to guide and control them grows proportionally, preventing them from becoming opaque or difficult to govern.

Transposed to PdM, automated decisions may affect plant safety, operational continuity as well as safety. The dimensions provided by the agency problem provide a conceptual map for assessing risks, controls, and responsibilities throughout the entire lifecycle of agentic solutions.

4.3. AI Agents for Machine Monitoring, Diagnosis and Root Cause Analsyis

In some contexts, faults and malfunctions are diagnosed based on predefined thresholds for monitored parameters. An AI agent could have access, including through telemetry systems, to monitored data, the operating context, technical standards and notes, and fault history to infer the causes of failure. In the field of rotating machinery, this approach can translate into identifying, for example, misalignments or damage to critical components such as bearings. The impact that automation would have in this regard could be to reduce the Mean Time To Repair (MTTR) and would have the advantage of including contexts and information expressed in natural language, which to date can only be interpreted and provided by human operators. In this sense, the development of MCP-based technologies could particularly facilitate development in this direction.

A similar approach with non-agentic behavior has recently been proposed with MaintAGT [83], a multimodal model for intelligent operation and maintenance that combines three components: a signal-to-text module (Sig2Txt) to convert raw monitoring signals into textual descriptions, a specialized textual model, and a multimodal model that integrates signals and text and adopts CoT [84] for step-by-step reasoning in diagnosis and questions and answering. The training dataset is constructed from scientific articles, manuals, international standards, and training materials for vibration analysts. In general tests, the system reports 70% accuracy, outperforming general-purpose LLMs and approaching the level of an ISO Level III analyst. To bridge the gap between signal data and textual analysis, they provide a HITL mechanism for continuous refinement. Overall, MaintAGT is presented as a step towards more automated and explainable maintenance, with stated benefits in condition monitoring, signal processing, and fault diagnosis.

If we consider the case of not having trained specific models but entrusting anomaly detection tasks entirely to the agent, the main risks asssociated with inference are related to the interpretation of data outside the historical distribution. This risk could be mitigated by setting up a system that acts on confidence thresholds and provides alternative hypotheses, actively integrating human input (human-in-the-loop) into a decision-making process that does not exclude human operators but facilitates their work and can increase their capabilities. The risk at this level is that of scalable oversight, which can occur when

agents reach speeds and ranges of action greater than those of human operators. The scalability of this approach is, in the opinion of the author, limited by the scalability of the supervision and should not be improved without proper analysis even if technologically possible. Nevertheless, the ability to guide the agent through, for example, appropriate MCP protocols that allow access to prompts, guidelines, template workflows, or even previously used models could reduce operational risks by appropriately constraining the ways in which the agent makes diagnoses and analyzes root causes. Such actions can in any case be performed in a virtual environment that interacts with an operator who then acts accordingly in the physical world.

Also, an agent operating at the edge could observe sensor data, perform quality checks, calculate features, and apply predictive pattern recognition models. In the presence of robust evidence of incipient failure, only significant data are sent to the cloud or CMMS. This would lead to a reduction in reaction latency and a drastic reduction in data traffic. Aside from the technical problems of implementing complex edge systems, the main risks are related to the opacity of decision-making and in the authority creep, given by the expansion of the agent's mandate beyond what was agreed upon. Indeed, in the physical world, such agents could theoretically act on the controls of the monitored apparatus. Risk mitigation would require a formal specification of what the agent can and cannot do, minimal but verifiable explanations (features, thresholds, reasons for reporting), a remote kill switch, and drift monitors that enforce downgrading to observation-only mode when uncertainty thresholds are exceeded. These precautions are consistent with the classic problems of information asymmetry and authority highlighted by agency theory applied to AI agents. Operationally, mitigations consist of defining a machine-readable capability policy that establishes what the agent can do with quantitative limits and confidence thresholds, and making every recommendation verifiable and subject to human approval when the impact is high. In practice, a remote kill switch and drift monitor are applied which, when operational thresholds are exceeded, force an automatic downgrade to observe-only.

The alignment problem could be particularly dangerous for actions taken in the physical world. Information asymmetry could prevent human operators from having enough information to intervene in a timely manner and could lead to catastrophic consequences if not carefully managed during the agent design phase. Therefore, such applications could only be implemented after a gradual process that fully integrates humans into the decision-making process.

4.4. Spare Parts Management and Procurement

A spare parts management agent could work on top of existing systems and combine three types of information: failure predictions from PdM, supplier lead times and reliability, and internal budget and service constraints. Based on this, it estimates the risks of stock depletion and proposes, in advance, the most reasonable actions: updating safety stocks, initiating a purchase request, expediting a delivery, or moving parts between sites. The agent caould prepare structured drafts with reasons for the proposal and the level of uncertainty, which the purchasing department can approve or review. The benefit translates into fewer downtimes due to lack of parts, fewer costly emergency purchases, and more balanced fixed assets.

Alongside the benefits, the use of an agent for spare parts and procurement exposes to recurring risks that must be governed by clear rules. First of all, there is information asymmetry since the agent may present proposals without explicitly stating uncertainties, assumptions, or alternatives (for example, underestimating the variability of delivery times). Therefore, a true duty to inform is needed. In terms of powers and authority, the mandate may extend beyond the initial intent (for example, by systematically applying

expensive urgent shipments). Human approval and minimum authority principle should be employed. The agent must also be aligned with multiple interests regarding not only cost reduction, but also operational continuity, security, and environmental impact. Since it can delegate to other tools or agents, a delegation policy, a traceable log are needed. Agent identity should be registered on each action. The scalability of the supervision may require auditing agents [132]. Operationally, the agent operates with a duty-to-inform in the sense that each proposal must include uncertainties, assumptions, and alternatives in a standardized manner, otherwise it cannot be forwarded. The delegation policy defines which tools or agent it can delegate to, with logs.

4.5. Orchestration and Planning of Interventions Using CMMS Systems

When an anomaly exceeds severity and impact thresholds, an AI agent can autonomously pre-fill a work order in the CMMS, including signals and trends, fault codes, assets involved, process criticality, suggested actions, and a priority consistent with the estimated risk. This can reduce backlogs and increase consistency between diagnosis and planning. Alongside these benefits, however, some concerns emerge: proliferation of unnecessary or duplicate tickets, bypassing of processes and roles with risks of authority creep, information asymmetry and decision-making opacity because of the difficulty in verifying motivations, possible deviations from loyalty to the user's interests since measurable Key Performance Indicators (KPIs) may be optimized at the expense of safety or compliance constraints, and delegation to sub-agents with poorly tracked decision-making chains. A first step in this direction could be seen as that taken by Bengtsson et al. [97] in the processing of textual data in CMMSs to identify recurring knowledge and failures.

Mitigations could be structured on multiple levels. Human approval gates bound Agents to receiving human confirmation before acting. Then, authority limits with role-based access control (RBAC) approach for sensitive CMMS operations could organize access to resources by defining roles with specific privileges and assigning them to users. Also in this case, HITL policy should be structured. Each step of the action may then require logs to ensure the traceability of events. During the planning phase, the agent can cross-reference failure forecasts, production windows, technician availability, and spare parts to propose maintenance plans. Scalable oversight problems could be particularly evident in these cases. Uncontrolled delegation could be mitigated by structuring authorized and tracked hand-offs, intended as explicit transfer of control and context from one actor to another within an agentic flow. Operationally, implementation involves approval gates and RBAC in the CMMS, with authority limits and a HITL policy for high-impact actions, as well as signed logs and end-to-end traceability of phases. During the planning phase, the agent cross-references failure predictions, production windows, and the availability of technicians and spare parts to generate executable plans, while delegations are only permitted through authorized hand-offs tracked between tools and agents.

4.6. Life-Cycle Management of Predictive Models for Machine Maintenance

In more structured systems with ad hoc predictive models, a dedicated agent could oversee the entire life cycle of the predictive models and observe their performance on real systems, detect variations in data distribution that could affect their reliability, and orchestrate retraining in a test environment, proposing the transition to production of the new version. In the field of PdM, this task is particularly important because failures are rare and often labeled late or with noise; operating conditions may change after each intervention and different sites show heterogeneous behavior. The model can become obsolete not only because new data arrives, but because the underlying world shifts.

Mitigation strategies should combine limits of authority, strong visibility, and scalable surveillance. On the first point, the agent should not be delegated to act in production with new versions. Its mandate should be to prepare the assessment and propose promotion when the set of evidence exceeds predefined thresholds. The visibility can be promoted through version logging. Surveillance should be scalable, then an autonomous agent-auditor should verify checklists and objective criteria (accuracy, false alarms, missed detections, average lead time, stability across sites, data drift indicators), automatically block promotions that do not comply with safety policies and report borderline cases to a human reviewer. Operationally, authority limits require that the agent does not operate in production with new versions, and visibility is ensured by version logging. Scalable surveillance could be implemented in an autonomous agent-auditor that verifies checklists and objective criteria.

5. Drivers and Barriers for the Adoption of Agentic Systems in Industrial Maintenance: A Global Economic Perspective

In order to identify barriers and drivers, reference is made in this section to the global economic impact demonstrated to date by this type of technologies in broad sectors. Given the relative scarcity of specific applied contributions, the analysis developed in this section is based on evidences and cross-sectional studies based on recent data and surveys from industry sector. To this end, the author draws on resources based on reports and sample surveys, used as an empirical basis for rigorously outlining the opportunities, risks, and challenges of adoption. In this context, the main sources of reference are the AI index report annually provided by the Stanford Institute for Human-Centered Artificial Intelligence [31], the research work on AI usage patterns by Handa et al. [133] and the research study of Brynjolfsson et al. [134] on the effects of AI assistance on productivity. The main drivers and barriers identified are shown in Table 4. This analysis is conducted on this framework, as operational evidence is still limited, acknowledging that substantial data on PdM-specific adoption will emerge only over time, enabling more in-depth analyses.

If the general context is analyzed and transposed to the specific field of PdM, a mixed but systematically favorable picture emerges for the adoption of agentic systems for industrial maintenance. On the one hand, the breadth of corporate adoption and the availability of capitals for investments are creating enabling conditions; on the other hand, there are still signs of caution regarding economic returns, the geographical distribution of investments, and the organizational capabilities needed to scale robustly.

Among the main drivers of development of these technologies is the flow of corporate investments made globally in AI by public and private companies, which in 2024 saw a trend increase of 25.5% [31], which could be interpreted as a sign of general confidence among companies in the AI field. In this regard, there has been a marked increase in private investment (+44.5%), and within this, generative AI alone raised $33.9 billion, marking an 18.7% increase compared to 2023 and almost ten times that of 2022 [31]. A similar growth trend can be seen in newly funded companies in the AI and generative AI fields. At the global system level, these signs could pave the way for resources and fertile ground for enterprise initiatives, including in domains related to PdM. In sectors that include monitoring and maintenance, such as manufacturing, the share of private investment has more than tripled compared to 2023, reaching almost 7% of the total [31]. Nevertheless, although maintenance and monitoring operations are classified in certain specific sectors such as manufacturing and oil & gas, the reports under analysis do not provide specific data on the sector and PdM technologies or the development possibilities proposed to the author in this work. Organizational adoption shows signs of acceleration overall, with the

use of generative AI in at least one function reaching 71% in 2024, compared to 33% in 2023. On the other hand, 78% of companies report using some form of AI [31].

Evidence of productivity gains is also consolidating, with analyses from 2024 documenting positive effects in multiple domains ranging from 10% to 45% [31]. The study proposed by Brynjolfsson et al. [134] was based on 5179 customer support employees and estimates a 14.2% increase in tickets resolved per hour after the introduction of a generative assistant. Similar trends have been observed in office tasks and specialist roles, albeit with heterogeneity linked to skills and contexts [31]. For those designing agents in the PdM field, these results could suggest concrete opportunities for efficiency in diagnosis, planning, and reporting activities, provided that process integration and impact measurement are taken care of. For example, a system that can access diagnostic tools through MCP could support operators in resolving machine diagnoses.

In terms of physical infrastructure, there has been growth in the robotics base, with the global operational stock of industrial robots reaching 4.282 million units in 2023 and collaborative robots increasing from 2.8% of new installations in 2017 to 10.5% in 2023 [31]. These figures could be a positive sign for scenarios in which software agents cooperate with physical systems for inspection and maintenance. However, it should be noted that new installations in 2023 have declined slightly and that adoption remains highly concentrated geographically. These factors call for caution when considering, for example, global supply chains and inventory management.

From the point of view of barriers, it should be emphasized that the evidence available today, for example based on conversations recently mapped by Claude.ai [133], shows that the pattern of use of some LLM-based systems is mainly oriented towards increasing human operational capabilities in a collaborative manner rather than in the complete delegation of tasks [133]. The pattern of use is mostly augmentative (57% of interactions) rather than of complete automation (43%), which facilitates introduction models that support and collaborate with operators rather than be completely delegated. As previously emphasized, this is a crucial aspect in maintenance contexts. These critical issues could in fact be considered in line with the problems of alignment and scalable oversight previously introduced. These considerations could highlight the justified caution in using and deploying agentic tools that have a high degree of autonomy. The recent study by Handa et al. [31,133] also highlights how the skills exhibited in conversations on Claude.ai currently leave little room for skills that require physical interaction, such as installation, repair, and equipment maintenance, whose presence is negligible compared to cognitive skills such as critical thinking or programming tasks.

Furthermore, in terms of barriers, the average financial impact reported by companies is still limited [31]. Many organizations that use AI report savings mainly in service operations and supply chains, but typically below 10%, whereas in terms of revenue, the most frequent increases are below 5%. This could be interpreted as a sign that for a large proportion of companies we are in the early stages of capturing value. Furthermore, adoption shows significant regional disparities and sectoral cyclicity that can influence the timing and models of return on investment in operations. Further caution comes from studies on work [133,134], which highlight how benefits depend on the ability to systematically integrate tools, data, and processes, and not just on the adoption of technology.

Overall, the outlook for agentic systems in maintenance appears positive and evidences show that two factors are present at the systemic level. In particular, this author identifies the direction and concentration of capital in the sector of AI and generative AI and the positive trends on technology adoption. Also, the first signs of productivity are encouraging, and the cyber-physical base (collaborative robots, IoT infrastructure) offers fertile ground. According to this author, reservations and cautions are mandatory, as there are few use

cases anchored to operational KPIs, and gradual integration paths are necessary. As previously highlighted, it is also essential to design a risk and responsibility governance system consistent with industrial constraints. In the author's opinion, the realistic trajectory of AI agent adoption for maintenance today hinges on this balance between enabling forces and concrete constraints.

Table 4. Adopting agentic systems in industrial maintenance: evidence and implications.

Factor	Type	Evidence	Implication
Investment momentum in AI and Generative AI	Driver	• Global corporate AI investment: +25.5% (2024) • Private investment: +44.5% • GenAI: +18.7% (10 times higher than 2022) [31]	• Capital available for PdM pilots • MCP-enabled integrations
Organizational adoption acceleration	Driver	• GenAI used by 71% of firms (2024) vs. 33% (2023) • 78% of companies use some AI [31]	• Readiness for agentic PdM with HITL patterns • Emphasis on measured outcomes
Productivity gains	Driver	• Meta-analyses show +10–45% • Generative assistant in customer support showed +14.2% tickets/hour [31,134]	• Target efficiency in diagnosis, planning, reporting
Robotics base growth	Driver	• 4.282 M industrial robots in operation (2023) • Cobots: 10.5% of new installs (2.8% in 2017) [31]	• Ground for software–physical cooperation (inspection/maintenance)
Modest average financial impact	Barrier	• Typical savings $< 10\%$ • Revenue gains often $< 5\%$ among adopters [31]	• Gradual deployments • Focus on high-ROI, KPI-tied use cases
Regional and sector asymmetries	Barrier	• Uneven adoption and investment • Slight decline in 2023 robot installs • Geographic concentration [31]	• Site-specific roll-outs • Alignment with supply-chain constraints
Usage pattern mainly augmentative	Barrier	• 57% augmentative vs. 43% full automation (mapped conversations) [133]	• Prefer HITL, approval gates, soft-agent roles before higher autonomy
Limited physical-interaction skills in LLM usage survey	Barrier	• Negligible presence of repair and maintenance skills vs. cognitive skills [133]	• Prioritize informational and decision-support agents

6. Future Perspectives and Roadmap

In light of the analyses presented, several lines of development emerge that can guide the evolution of research and practical application in the field of PdM based on LLMs. These directions are not merely incremental extensions of existing approaches, but outline a roadmap for research and development capable of integrating experimental validation, technological innovations, and organizational aspects. In particular, the proposed path is structured as follows. The proposed roadmap should be understood as an illustrative hypothesis, developed on the basis of currently available evidence and sources. It constitutes a conceptual exercise that is feasible under the conditions described, but does not claim to be predictive. Future developments, new evidence, or technological and organizational changes could obviously overturn the considerations made.

- Validation and foundation (1–2 years). In the short term, the integration of multimodal data, including text and sensor data, could be implemented, aiming at validation in laboratory environments that also seeks to identify evaluation standards:
  - multimodal integration and digital twins with robust sensor, text–image pipelines and simulations;
  - design evaluation standars for PdM agents: task suites, risk scenarios, autonomy levels, and security requirements;
  - validation in laboratory environments.
- Technological consolidation (3–5 years). In the medium term, systems with standardized and proven standardized retrieval protocols should be developed to ensure security in vertical applications, including the analysis of operations on edge systems and validation on real plants:
  - vertical LLMs with structured retrieval, secure and standard tool calling (MCP) to reduce hallucinations and increase procedural fidelity;
  - egde AI with efficient, privacy-preserving models and agents with latencies compatible with operational control;
  - experimental validation on real plants and public benchmarks with reproducible test protocols, realistic datasets, and shared metrics for CMMS diagnostics, prescriptions, and orchestration.
- Structured organizational adoption (5+ years). In the long term, a structured adoption at the organizational level and on a large scale could be targeted:
  - organizational integration into processes and data, operator training, and consistent roles and permission design;
  - governance and explainability.

7. Conclusions

This work leverages the tool of literature analysis and review to explore the potential of LLM-based AI agents to autonomously perform PdM tasks. Namely, this review contributes by consolidating fragmented evidence on LLM-based agents for PdM, outlining a structured roadmap and linking technological advances to industrial drivers and barriers. LLM-based autonomous agents represent an extension of discriminative and generative approaches to PdM, as they are configured as systems capable of perceiving, reasoning, planning, and interacting with tools and knowledge bases throughout the entire PdM cycle. In this context, the availability of the modern LLM application and technology stack with tool-use, RAG, and, increasingly, interoperability protocols such as the MCP offers a technological basis that can already be used to orchestrate data, procedures, and actions. Although there are not yet any solid cases of MCP in PdM, the literature documents adjacent patterns such as LLMs enhanced by tools and knowledge bases that make a transition to soft agents supporting diagnosis, decisions, prescriptions, and CMMS orchestration credible.

From an industrial perspective, macro analysis indicates a favorable environment and widespread adoption of generative AI with positive results in terms of productivity, but with average economic returns still limited and regional and sectoral variability. This results in a cautious trajectory that seems to suggest introducing agents in well-defined areas, linked to operational KPIs, with impact measurement and graduated rights of action. In the short term, an enabling role for information and decision-support agents is likely, while operational autonomy for interaction with controls requires further evidence, technical barriers, and more rigorous governance.

The main critical issues that emerged remain: the domain-specific knowledge gap for LLMs, which can be partially mitigated via RAG, fine-tuning, and vertical knowledge

bases; multimodal integration (sensors, text, images) and cross-domain and site generalization; computational and security constraints for edge deployment; interpretability and traceability of decisions; the absence of standard benchmarks and evaluations for LLMs and agents in PdM; possible vendor lock-in risks related to access to proprietary models underlying the systems; problems of scalable oversight and goal alignment in the presence of delegation and multi-agent chains. These challenges currently require an explicit governance architecture (visibility, traceability, accountability), with human approval gates and delegation policies.

Ultimately, the challenge and promise of LLM-based agents in PdM lie in their ability to combine perception, reasoning, and action in a unified manner. The potential industrial impact will therefore depend not only on the technology stack but also on the establishment of rigorous governance procedures, open standards, and measured, incremental deployment.

Future research should prioritize the use of open benchmarks and reproducible evaluation protocols. Governance frameworks for agentic systems should be investigated, in order to enable systematic validation and accelerate safe industrial adoption.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Ali, M.I.; Lai, N.S.; Abdulla, R. Predictive maintenance of rotational machinery using deep learning. *Int. J. Electr. Comput. Eng. (IJECE)* **2024**, *14*, 1112. [CrossRef]
2. Zhou, J.; Yang, J.; Qian, Q.; Qin, Y. A comprehensive survey of machine remaining useful life prediction approaches based on pattern recognition: Taxonomy and challenges. *Meas. Sci. Technol.* **2024**, *35*, 062001. [CrossRef]
3. Ma, S.; Flanigan, K.A.; Bergés, M. State-of-the-Art Review: The Use of Digital Twins to Support Artificial Intelligence-Guided Predictive Maintenance. *arXiv* **2024**, arXiv:2406.13117. [CrossRef]
4. Mikołajewska, E.; Mikołajewski, D.; Mikołajczyk, T.; Paczkowski, T. Generative AI in AI-Based Digital Twins for Fault Diagnosis for Predictive Maintenance in Industry 4.0/5.0. *Appl. Sci.* **2025**, *15*, 3166. [CrossRef]
5. Surucu, O.; Gadsden, S.A.; Yawney, J. Condition Monitoring using Machine Learning: A Review of Theory, Applications, and Recent Advances. *Expert Syst. Appl.* **2023**, *221*, 119738. [CrossRef]
6. Lei, Y.; Yang, B.; Jiang, X.; Jia, F.; Li, N.; Nandi, A.K. Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mech. Syst. Signal Process.* **2020**, *138*, 106587. [CrossRef]
7. Lei, Y.; Li, N.; Guo, L.; Li, N.; Yan, T.; Lin, J. Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mech. Syst. Signal Process.* **2018**, *104*, 799–834. [CrossRef]
8. Hendriks, J.; Dumond, P.; Knox, D. Towards better benchmarking using the CWRU bearing fault dataset. *Mech. Syst. Signal Process.* **2022**, *169*, 108732. [CrossRef]
9. Farooq, U.; Ademola, M.; Shaalan, A. Comparative Analysis of Machine Learning Models for Predictive Maintenance of Ball Bearing Systems. *Electronics* **2024**, *13*, 438. [CrossRef]
10. Giraudo, L.; Di Maggio, L.G.; Giorio, L.; Delprete, C. Dynamic Multibody Modeling of Spherical Roller Bearings with Localized Defects for Large-Scale Rotating Machinery. *Sensors* **2025**, *25*, 2419. [CrossRef]
11. Vehviläinen, M.; Tahkola, M.; Keränen, J.; El Bouharrouti, N.; Rahkola, P.; Halme, J.; Pippuri-Mäkeläinen, J.; Belahcen, A. 3D Multibody Simulation of Realistic Rolling Bearing Defects for Fault Classifier Development. In Proceedings of the 2024 International Conference on Electrical Machines (ICEM), Torino, Italy, 1–4 September 2024; pp. 1–7. [CrossRef]
12. Yang, B.; Lei, Y.; Jia, F.; Xing, S. An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings. *Mech. Syst. Signal Process.* **2019**, *122*, 692–706. [CrossRef]
13. Martínez-Rego, D.; Fontenla-Romero, O.; Alonso-Betanzos, A. Power wind mill fault detection via one-class v-SVM vibration signal analysis. *Proc. Int. Jt. Conf. Neural Netw.* **2011**, *10748*, 511–518. [CrossRef]

14. Widodo, A.; Yang, B.S. Support vector machine in machine condition monitoring and fault diagnosis. *Mech. Syst. Signal Process.* **2007**, *21*, 2560–2574. [CrossRef]
15. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. [CrossRef] [PubMed]
16. Di Maggio, L.G. Intelligent Fault Diagnosis of Industrial Bearings Using Transfer Learning and CNNs Pre-Trained for Audio Classification. *Sensors* **2022**, *23*, 211. [CrossRef]
17. Eang, C.; Lee, S. Predictive Maintenance and Fault Detection for Motor Drive Control Systems in Industrial Robots Using CNN-RNN-Based Observers. *Sensors* **2024**, *25*, 25. [CrossRef]
18. Thoppil, N.M.; Vasu, V.; Rao, C.S.P. Deep Learning Algorithms for Machinery Health Prognostics Using Time-Series Data: A Review. *J. Vib. Eng. Technol.* **2021**, *9*, 1123–1145. [CrossRef]
19. Wu, Y.; Sicard, B.; Gadsden, S.A. Physics-informed machine learning: A comprehensive review on applications in anomaly detection and condition monitoring. *Expert Syst. Appl.* **2024**, *255*, 124678. [CrossRef]
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30* . [CrossRef]
21. Jin, Y.; Hou, L.; Chen, Y. A Time Series Transformer based method for the rotating machinery fault diagnosis. *Neurocomputing* **2022**, *494*, 379–395. [CrossRef]
22. Cao, S.; Wen, L.; Li, X.; Gao, L. Application of Generative Adversarial Networks for Intelligent Fault Diagnosis. In Proceedings of the 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE), Munich, Germany, 20–24 August 2018. [CrossRef]
23. Yang, X.; Fang, C.; Liao, Y.; Yang, J.; Gryllias, K.; Chronopoulos, D. Deep Generative Models in Condition and Structural Health Monitoring: Opportunities, Limitations and Future Outlook. *arXiv* **2025**, arXiv:2507.15026. [CrossRef]
24. OpenAI. GPT-4 Technical Report. *arXiv* **2023**, arXiv:2303.08774. [CrossRef]
25. Wu, J.; Gan, W.; Chen, Z.; Wan, S.; Yu, P.S. Multimodal Large Language Models: A Survey. *arXiv* **2023**, arXiv:2311.13165. [CrossRef]
26. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. *arXiv* **2020**, arXiv:2005.14165. [CrossRef]
27. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.t.; Rocktäschel, T.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: Nice, France, 2020; Volume 33, pp. 9459–9474.
28. Reddicharla, N.; Ali, M.S. Innovating Oil and Gas Field Operations—Harnessing the Power of Generative Ai for Supporting Workforce Towards Achieving Autonomous Operations. In Proceedings of the ADIPEC, Abu Dhabi, United Arab Emirates, 4–7 November 2024; p. D011S020R005. [CrossRef]
29. Kolt, N. Governing AI Agents. *arXiv* **2025**, arXiv:2501.07913. [CrossRef]
30. Jensen, M.C.; Meckling, W.H. Theory of the firm: Managerial behavior, agency costs and ownership structure. In *Corporate Governance*; Gower: Aldershot, UK, 2019; pp. 77–132.
31. Maslej, N.; Fattorini, L.; Perrault, R.; Gil, Y.; Parli, V.; Kariuki, N.; Capstick, E.; Reuel, A.; Brynjolfsson, E.; Etchemendy, J.; et al. Artificial Intelligence Index Report 2025. *arXiv* **2025**, arXiv:2504.07139. [CrossRef]
32. Khan, S.; Yairi, T. A review on the application of deep learning in system health management. *Mech. Syst. Signal Process.* **2018**, *107*, 241–265. [CrossRef]
33. Zhao, R.; Yan, R.; Chen, Z.; Mao, K.; Wang, P.; Gao, R.X. Deep learning and its applications to machine health monitoring. *Mech. Syst. Signal Process.* **2019**, *115*, 213–237. [CrossRef]
34. Yang, D.; Karimi, H.R.; Gelman, L. A Fuzzy Fusion Rotating Machinery Fault Diagnosis Framework Based on the Enhancement Deep Convolutional Neural Networks. *Sensors* **2022**, *22*, 671. [CrossRef]
35. Kolar, D.; Lisjak, D.; Pająk, M.; Gudlin, M. Intelligent Fault Diagnosis of Rotary Machinery by Convolutional Neural Network with Automatic Hyper-Parameters Tuning Using Bayesian Optimization. *Sensors* **2021**, *21*, 2411. [CrossRef]
36. Ranjit M. Gawande. Machine Learning Approaches for Fault Detection and Diagnosis in Electrical Machines: A Comparative Study of Deep Learning and Classical Methods. *Panam. Math. J.* **2024**, *34*, 121–137. [CrossRef]
37. Sharma, S.S.; Vivek, V.; Malviya, A. AI-Enhanced Predictive Maintenance in Intelligent Systems for Industries. In Proceedings of the 2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET), Indore, India, 27–28 September 2024; pp. 1–6. [CrossRef]
38. Sperandio Nascimento, E.G.; Liang, J.S.; Figueiredo, I.S.; Guarieiro, L.L.N. T4pdm: A Deep Neural Network Based on the Transformer Architecture for Fault Diagnosis of Rotating Machinery. *SSRN Electron. J.* **2022** . [CrossRef]

39. Lu, Z.; Liang, L.; Zhu, J.; Zou, W.; Mao, L. Rotating Machinery Fault Diagnosis Under Multiple Working Conditions via a Time-Series Transformer Enhanced by Convolutional Neural Network. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–11. [CrossRef]
40. Brusa, E.; Delprete, C.; Di Maggio, L.G. Eigen-spectrograms: An interpretable feature space for bearing fault diagnosis based on artificial intelligence and image processing. *Mech. Adv. Mater. Struct.* **2022**, *30*, 4639–4651. [CrossRef]
41. Sawaqed, L.S.; Alrayes, A.M. Bearing fault diagnostic using machine learning algorithms. *Prog. Artif. Intell.* **2020**, *9*, 341–350. [CrossRef]
42. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
43. He, W.; Chen, J.; Zhou, Y.; Liu, X.; Chen, B.; Guo, B. An Intelligent Machinery Fault Diagnosis Method Based on GAN and Transfer Learning under Variable Working Conditions. *Sensors* **2022**, *22*, 9175. [CrossRef] [PubMed]
44. Guo, L.; Lei, Y.; Xing, S.; Yan, T.; Li, N. Deep Convolutional Transfer Learning Network: A New Method for Intelligent Fault Diagnosis of Machines with Unlabeled Data. *IEEE Trans. Ind. Electron.* **2019**, *66*, 7316–7325. [CrossRef]
45. Shen, S.; Lu, H.; Sadoughi, M.; Hu, C.; Nemani, V.; Thelen, A.; Webster, K.; Darr, M.; Sidon, J.; Kenny, S. A physics-informed deep learning approach for bearing fault detection. *Eng. Appl. Artif. Intell.* **2021**, *103*, 104295. [CrossRef]
46. Qin, Y.; Liu, H.; Wang, Y.; Mao, Y. Inverse physics–informed neural networks for digital twin–based bearing fault diagnosis under imbalanced samples. *Knowl.-Based Syst.* **2024**, *292*, 111641. [CrossRef]
47. Ni, Q.; Ji, J.; Halkon, B.; Feng, K.; Nandi, A.K. Physics-Informed Residual Network (PIResNet) for rolling element bearing fault diagnostics. *Mech. Syst. Signal Process.* **2023**, *200*, 110544. [CrossRef]
48. Lu, H.; Pavan Nemani, V.; Barzegar, V.; Allen, C.; Hu, C.; Laflamme, S.; Sarkar, S.; Zimmerman, A.T. A physics-informed feature weighting method for bearing fault diagnostics. *Mech. Syst. Signal Process.* **2023**, *191*, 110171. [CrossRef]
49. Jia, N.; Huang, W.; Ding, C.; Wang, J.; Zhu, Z. Physics-informed unsupervised domain adaptation framework for cross-machine bearing fault diagnosis. *Adv. Eng. Inform.* **2024**, *62*, 102774. [CrossRef]
50. Yue, K.; Li, J.; Chen, J.; Huang, R.; Li, W. Multiscale Wavelet Prototypical Network for Cross-Component Few-Shot Intelligent Fault Diagnosis. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 3502411. [CrossRef]
51. Zhang, S.; Ye, F.; Wang, B.; Habetler, T. Few-Shot Bearing Fault Diagnosis Based on Model-Agnostic Meta-Learning. *IEEE Trans. Ind. Appl.* **2021**, *57*, 4754–4764. [CrossRef]
52. Chang, L.; Lin, Y.H. Meta-Learning With Adaptive Learning Rates for Few-Shot Fault Diagnosis. *IEEE/ASME Trans. Mechatronics* **2022**, *27*, 5948–5958. [CrossRef]
53. Han, J.; Zhang, X.; Xie, Z.; Zhou, W.; Tan, Z. Federated Learning-Based Equipment Fault-Detection Algorithm. *Electronics* **2024**, *14*, 92. [CrossRef]
54. Milasheuski, U.; Baraldi, P.; Zio, E.; Savazzi, S. Federated Generative Models for Predictive Maintenance in Industrial Environments. In Proceedings of the 2024 8th International Conference on System Reliability and Safety (ICSRS), Sicily, Italy, 20–22 November 2024; pp. 156–161. [CrossRef]
55. Ma, X.; Wen, C.; Wen, T. An asynchronous and real-time update paradigm of federated learning for fault diagnosis. *IEEE Trans. Ind. Inform.* **2021**, *17*, 8531–8540. [CrossRef]
56. Zhang, W.; Li, X.; Ma, H.; Luo, Z.; Li, X. Federated learning for machinery fault diagnosis with dynamic validation and self-supervision. *Knowl.-Based Syst.* **2021**, *213*, 106679. [CrossRef]
57. Di Maggio, L.G.; Brusa, E.; Delprete, C. Novelty Detection in Rotating Machinery: Assessment of Unsupervised Machine Learning Models for Medium-Sized Industrial Bearings. In Proceedings of the 2025 International Conference on Control, Automation and Diagnosis (ICCAD), Barcelona, Spain, 1–3 July 2025; pp. 1–7. [CrossRef]
58. De Fabritiis, F.; Gryllias, K. Self-supervised Learning Approach for Anomaly Detection in Rotating Machinery. *Annu. Conf. Phm Soc.* **2024**, *16*, 1–10. [CrossRef]
59. Han, T.; Li, Y.F. Out-of-distribution detection-assisted trustworthy machinery fault diagnosis approach with uncertainty-aware deep ensembles. *Reliab. Eng. Syst. Saf.* **2022**, *226*, 108648. [CrossRef]
60. Zhou, T.; Han, T.; Droguett, E.L. Towards trustworthy machine fault diagnosis: A probabilistic Bayesian deep learning framework. *Reliab. Eng. Syst. Saf.* **2022**, *224*, 108525. [CrossRef]
61. Brito, L.C.; Susto, G.A.; Brito, J.N.; Duarte, M.A. An explainable artificial intelligence approach for unsupervised fault detection and diagnosis in rotating machinery. *Mech. Syst. Signal Process.* **2022**, *163*, 108105. [CrossRef]
62. Brusa, E.; Cibrario, L.; Delprete, C.; Di Maggio, L.G. Explainable AI for Machine Fault Diagnosis: Understanding Features' Contribution in Machine Learning Models for Industrial Condition Monitoring. *Appl. Sci.* **2023**, *13*, 2038. [CrossRef]
63. Harvinder, S. Reimagining manufacturing with generative AI: A comprehensive review of current applications and future directions. *i-manager's J. Future Eng. Technol.* **2025**, *20*, 51. [CrossRef]
64. Mohapatra, A. Generative AI for Predictive Maintenance: Predicting Equipment Failures and Optimizing Maintenance Schedules Using AI. *Int. J. Sci. Res. Manag. (IJSRM)* **2024**, *12*, 1648–1672. [CrossRef]
65. Di Maggio, L.G.; Brusa, E.; Delprete, C. Zero-Shot Generative AI for Rotating Machinery Fault Diagnosis: Synthesizing Highly Realistic Training Data via Cycle-Consistent Adversarial Networks. *Appl. Sci.* **2023**, *13*, 12458. [CrossRef]

66. Hakami, A. Strategies for overcoming data scarcity, imbalance, and feature selection challenges in machine learning models for predictive maintenance. *Sci. Rep.* **2024**, *14*, 9645. [CrossRef]
67. Huo, L.; Qi, H.; Fei, S.; Guan, C.; Li, J. A Generative Adversarial Network Based a Rolling Bearing Data Generation Method Towards Fault Diagnosis. *Comput. Intell. Neurosci.* **2022**, *2022*, 1–21. [CrossRef]
68. Guo, Q.; Li, Y.; Liu, Y.; Gao, S.; Song, Y. Data Augmentation for Intelligent Mechanical Fault Diagnosis Based on Local Shared Multiple-Generator GAN. *IEEE Sens. J.* **2022**, *22*, 9598–9609. [CrossRef]
69. Ding, Y.; Ma, L.; Ma, J.; Wang, C.; Lu, C. A Generative Adversarial Network-Based Intelligent Fault Diagnosis Method for Rotating Machinery Under Small Sample Size Conditions. *IEEE Access* **2019**, *7*, 149736–149749. [CrossRef]
70. Zhao, B.; Yuan, Q. Improved generative adversarial network for vibration-based fault diagnosis with imbalanced data. *Measurement* **2021**, *169*, 108522. [CrossRef]
71. Brusa, E.; Delprete, C.; Giorio, L.; Di Maggio, L.G.; Zanella, V. Design of an Innovative Test Rig for Industrial Bearing Monitoring with Self-Balancing Layout. *Machines* **2022**, *10*, 54. [CrossRef]
72. Yigin, B.; Celik, M. A Prescriptive Model for Failure Analysis in Ship Machinery Monitoring Using Generative Adversarial Networks. *J. Mar. Sci. Eng.* **2024**, *12*, 493. [CrossRef]
73. Choubey, S.; Benton, R.; Johnsten, T. Prescriptive Equipment Maintenance: A Framework. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; pp. 4366–4374. [CrossRef]
74. Ansari, F.; Glawar, R.; Nemeth, T. PriMa: A prescriptive maintenance model for cyber-physical production systems. *Int. J. Comput. Integr. Manuf.* **2019**, *32*, 482–503. [CrossRef]
75. Li, L.; Fan, Y.; Tse, M.; Lin, K.Y. A review of applications in federated learning. *Comput. Ind. Eng.* **2020**, *149*, 106854. [CrossRef]
76. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [CrossRef]
77. Thompson, A.; Sommers, A.; Russell-Gilbert, A.; Cummins, L.; Mittal, S.; Rahimi, S.; Seale, M.; Jaboure, J.; Arnold, T.; Church, J. Multivariate Data Augmentation for Predictive Maintenance using Diffusion. In Proceedings of the 2024 IEEE International Conference on Big Data (BigData), Washington, DC, USA, 15–18 December 2024; pp. 4240–4247. [CrossRef]
78. Solís-Martín, D.; Galán-Páez, J.; Borrego-Díaz, J. difLIME: Enhancing Explainability with a Diffusion-Based LIME Algorithm for Predictive Maintenance. *Int. J. Progn. Health Manag.* **2025**, *16* . [CrossRef]
79. Zhao, Y.; Yang, J.; Wang, W.; Yang, H.; Niyato, D. TranDRL: A Transformer-Driven Deep Reinforcement Learning Enabled Prescriptive Maintenance Framework. *IEEE Internet Things J.* **2024**, *11*, 35432–35444. [CrossRef]
80. Wang, Y.; Yu, Y.; Sun, K.; Lei, P.; Zhang, Y.; Zio, E.; Xia, A.; Li, Y. RmGPT: A Foundation Model With Generative Pre-trained Transformer for Fault Diagnosis and Prognosis in Rotating Machinery. *IEEE Internet Things J.* **2025**, *12*, 41562–41573. [CrossRef]
81. Lynch, K.; Lorenzi, F.; Sheehan, J.; Kabakci-Zorlu, D.; Eck, B. FMEA Builder: Expert Guided Text Generation for Equipment Maintenance. *arXiv* **2024**, arXiv:2411.05054. [CrossRef]
82. Moran, G.E.; Aragam, B. Towards Interpretable Deep Generative Models via Causal Representation Learning. *arXiv* **2025**, arXiv:2504.11609. [CrossRef]
83. He, H.; Huang, J.; Li, Q.; Wang, X.; Zhang, F.; Yang, K.; Meng, L.; Chu, F. MaintAGT:Sim2Real-Guided Multimodal Large Model for Intelligent Maintenance with Chain-of-Thought Reasoning. *arXiv* **2024**, arXiv:2412.00481. [CrossRef]
84. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc.: Nice, France, 2022; Volume 35, pp. 24824–24837.
85. Klekowicki, M.; Szymański, G.M.; Waligórski, M.; Misztal, W. Application of large language models in diagnostics and maintenance of aircraft propulsion systems. *Adv. Sci. Technol. Res. J.* **2024**, *19*, 304–320. [CrossRef]
86. Wang, H.; Li, Y.F. Large-Scale Language Models for PHM in Railway Systems - Potential Applications, Limitations, and Solutions. In *Proceedings of the 6th International Conference on Electrical Engineering and Information Technologies for Rail Transportation (EITRT) 2023*; Qin, Y., Jia, L., Yang, J., Diao, L., Yao, D., An, M., Eds.; Springer Nature: Singapore, 2024; Volume 1137, pp. 591–599. [CrossRef]
87. Tao, L.; Liu, H.; Ning, G.; Cao, W.; Huang, B.; Lu, C. LLM-based Framework for Bearing Fault Diagnosis. *arXiv* **2024**, arXiv:2411.02718. [CrossRef]
88. Qaid, H.A.A.M.; Zhang, B.; Li, D.; Ng, S.K.; Li, W. FD-LLM: Large Language Model for Fault Diagnosis of Machines. *arXiv* **2024**, arXiv:2412.01218. [CrossRef]
89. Deng, H.; Namoano, B.; Zheng, B.; Khan, S.; Ahmet Erkoyuncu, J. From Prediction to Prescription: Large Language Model Agent for Context-Aware Maintenance Decision Support. *PHM Soc. Eur. Conf.* **2024**, *8*, 10. [CrossRef]
90. Walker, C.; Rothon, C.; Aslansefat, K.; Papadopoulos, Y.; Dethlefs, N. Using Large Language Models to Recommend Repair Actions for Offshore Wind Maintenance. *J. Phys. Conf. Ser.* **2024**, *2875*, 012025. [CrossRef]
91. Lukens, S.; Ali, A. Evaluating the Performance of ChatGPT in the Automation of Maintenance Recommendations for Prognostics and Health Management. *Annu. Conf. PHM Soc.* **2023**, *15*, 1–18. [CrossRef]

92. Sun, Y.; Zhang, Q.; Bao, J.; Lu, Y.; Liu, S. Empowering digital twins with large language models for global temporal feature learning. *J. Manuf. Syst.* **2024**, *74*, 83–99. [CrossRef]
93. Zhang, X.; Sun, W.; Chen, K.; Jiang, R. A multimodal expert system for the intelligent monitoring and maintenance of transformers enhanced by multimodal language large model fine-tuning and digital twins. *IET Collab. Intell. Manuf.* **2024**, *6*, e70007. [CrossRef]
94. Ferdousi, R.; Hossain, M.A.; Yang, C.; Saddik, A.E. DefectTwin: When LLM Meets Digital Twin for Railway Defect Inspection. *arXiv* **2024**, arXiv:2409.06725. [CrossRef]
95. Alsaif, K.; Albeshri, A.; Khemakhem, M.; Eassa, F. Multimodal Large Language Model-Based Fault Detection and Diagnosis in Context of Industry 4.0. *Electronics* **2024**, *13*, 4912. [CrossRef]
96. Kumar, A.; Alam, M.; Farahat, A.; Somineni, M.; Gupta, C. Diagnostics-LLaVA: A Visual Language Model for Domain-Specific Diagnostics of Equipment. *Annu. Conf. PHM Soc.* **2024**, *16* . [CrossRef]
97. Bengtsson, M.; D'Cruze, R.S.; Ahmed, M.U.; Sakao, T.; Funk, P.; Sohlberg, R. Combining Ontology and Large Language Models to Identify Recurring Machine Failures in Free-Text Fields. In *Advances in Transdisciplinary Engineering*; Andersson, J., Joshi, S., Malmsköld, L., Hanning, F., Eds.; IOS Press: Amsterdam, The Netherlands, 2024. [CrossRef]
98. Wang, H.; Li, Y.F. Large Language Model Empowered by Domain-Specific Knowledge Base for Industrial Equipment Operation and Maintenance. In Proceedings of the 2023 5th International Conference on System Reliability and Safety Engineering (SRSE), Beijing, China, 20–23 October 2023; pp. 474–479. [CrossRef]
99. Kok, I.; Demirci, O.; Ozdemir, S. When IoT Meet LLMs: Applications and Challenges. *arXiv* **2024**, arXiv:2411.17722. [CrossRef]
100. Mahr, F.; Angeli, G.; Sindel, T.; Schmidt, K.; Franke, J. A Reference Architecture for Deploying Large Language Model Applications in Industrial Environments. In Proceedings of the 2024 IEEE 30th International Symposium for Design and Technology in Electronic Packaging (SIITME), Sibiu, Romania, 16–18 October 2024; pp. 19–23. [CrossRef]
101. Zheng, Y.; Chen, Y.; Qian, B.; Shi, X.; Shu, Y.; Chen, J. A Review on Edge Large Language Models: Design, Execution, and Applications. *arXiv* **2024**, arXiv:2410.11845. [CrossRef]
102. Friha, O.; Amine Ferrag, M.; Kantarci, B.; Cakmak, B.; Ozgun, A.; Ghoualmi-Zine, N. LLM-Based Edge Intelligence: A Comprehensive Survey on Architectures, Applications, Security and Trustworthiness. *IEEE Open J. Commun. Soc.* **2024**, *5*, 5799–5856. [CrossRef]
103. Dashdamirli, N. Integration of Large Language Models with microcontrollers to optimize industrial automation processes. *InterConf* **2024**, *46*, 500–507. [CrossRef]
104. Vyas, J.; Mercangöz, M. Autonomous Industrial Control using an Agentic Framework with Large Language Models. *arXiv* **2024**, arXiv:2411.05904. [CrossRef]
105. Wang, Z.; Qin, H. Intelligent industrial production process automatic regulation system based on LLM agents. In Proceedings of the 2024 5th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA), Shenzhen, China, 14–16 June 2024; pp. 133–137. [CrossRef]
106. Mavroudis, V. LangChain. 2024. Available online: https://www.preprints.org/manuscript/202411.0566 (accessed on 20 August 2024). [CrossRef]
107. Xia, Y.; Jazdi, N.; Zhang, J.; Shah, C.; Weyrich, M. Control Industrial Automation System with Large Language Model Agents. *arXiv* **2024**, arXiv:2409.18009. [CrossRef]
108. Xi, Z.; Chen, W.; Guo, X.; He, W.; Ding, Y.; Hong, B.; Zhang, M.; Wang, J.; Jin, S.; Zhou, E.; et al. The rise and potential of large language model based agents: A survey. *Sci. China Inf. Sci.* **2025**, *68*, 121101. [CrossRef]
109. Masterman, T.; Besen, S.; Sawtell, M.; Chao, A. The Landscape of Emerging AI Agent Architectures for Reasoning, Planning, and Tool Calling: A Survey. *arXiv* **2024**, arXiv:2404.11584. [CrossRef]
110. OpenAI. A Practical Guide to Building Agents. 2025. Available online: https://cdn.openai.com/business-guides-and-resources/a-practical-guide-to-building-agents.pdf (accessed on 15 September 2025).
111. Putta, P.; Mills, E.; Garg, N.; Motwani, S.; Finn, C.; Garg, D.; Rafailov, R. Agent Q: Advanced Reasoning and Learning for Autonomous AI Agents. *arXiv* **2024**, arXiv:2408.07199. [CrossRef]
112. Wilkins, D.E. *Practical Planning: Extending the Classical AI Planning Paradigm*; Elsevier: Amsterdam, The Netherlands, 2014.
113. Maes, P. *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*; MIT Press: Cambridge, MA, USA, 1990.
114. Bommasani, R.; Hudson, D.A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M.S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. On the Opportunities and Risks of Foundation Models. *arXiv* **2022**, arXiv:2108.07258.
115. Gozalo-Brizuela, R.; Garrido-Merchan, E.C. ChatGPT is not all you need. A State of the Art Review of large Generative AI models. *arXiv* **2023**, arXiv:2301.04655. [CrossRef]
116. Russell, S. *Human Compatible: AI and the Problem of Control*; Penguin: London, UK, 2019.
117. Ray, P.P. A Survey on Model Context Protocol: Architecture, State-of-the-art, Challenges and Future Directions. *TechRxiv* **2025** . [CrossRef]

118. Hou, X.; Zhao, Y.; Wang, S.; Wang, H. Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions. *arXiv* **2025**, arXiv:2503.23278. [CrossRef]
119. Xi, Z.; Ding, Y.; Chen, W.; Hong, B.; Guo, H.; Wang, J.; Yang, D.; Liao, C.; Guo, X.; He, W.; et al. AgentGym: Evolving Large Language Model-based Agents across Diverse Environments. *arXiv* **2024**, arXiv:2406.04151. [CrossRef]
120. Walker, C.; Gharaibeh, T.; Alsmadi, R.; Hall, C.; Baggili, I. Forensic Analysis of Artifacts from Microsoft's Multi-Agent LLM Platform AutoGen. In Proceedings of the 19th International Conference on Availability, Reliability and Security, Vienna, Austria, 30 July–2 August 2024; pp. 1–9. [CrossRef]
121. Chen, W.; Su, Y.; Zuo, J.; Yang, C.; Yuan, C.; Chan, C.M.; Yu, H.; Lu, Y.; Hung, Y.H.; Qian, C.; et al. AgentVerse: Facilitating Multi-Agent Collaboration and Exploring Emergent Behaviors. *arXiv* **2023**, arXiv:2308.10848. [CrossRef]
122. Huang, K.; Huang, J. AI Agent Tools and Frameworks. In *Agentic AI: Theories and Practices*; Huang, K., Ed.; Springer Nature: Cham, Switzerland, 2025; pp. 23–50. [CrossRef]
123. Bhattacharjee, A. Overview of Amazon Bedrock Agents. In *A Practical Guide to Generative AI Using Amazon Bedrock: Building, Deploying, and Securing Generative AI Applications*; Apress: Berkeley, CA, USA, 2025; pp. 229–248. [CrossRef]
124. Giada, C.V.; Rossella, P. Barriers to Predictive Maintenance implementation in the Italian machinery industry. *IFAC-PapersOnLine* **2021**, *54*, 1266–1271. [CrossRef]
125. Chen, Q.; Cao, J.; Zhu, S. Data-Driven Monitoring and Predictive Maintenance for Engineering Structures: Technologies, Implementation Challenges, and Future Directions. *IEEE Internet Things J.* **2023**, *10*, 14527–14551. [CrossRef]
126. Tiddens, W.; Braaksma, J.; Tinga, T. Exploring predictive maintenance applications in industry. *J. Qual. Maint. Eng.* **2022**, *28*, 68–85. [CrossRef]
127. Molęda, M.; Małysiak-Mrozek, B.; Ding, W.; Sunderam, V.; Mrozek, D. From Corrective to Predictive Maintenance—A Review of Maintenance Approaches for the Power Industry. *Sensors* **2023**, *23*, 5970. [CrossRef] [PubMed]
128. Azari, M.S.; Flammini, F.; Santini, S.; Caporuscio, M. A Systematic Literature Review on Transfer Learning for Predictive Maintenance in Industry 4.0. *IEEE Access* **2023**, *11*, 12887–12910. [CrossRef]
129. Strielkowski, W.; Vlasov, A.; Selivanov, K.; Muraviev, K.; Shakhnov, V. Prospects and Challenges of the Machine Learning and Data-Driven Methods for the Predictive Analysis of Power Systems: A Review. *Energies* **2023**, *16*, 4025. [CrossRef]
130. Ross, S.A. The economic theory of agency: The principal's problem. *Am. Econ. Rev.* **1973**, *63*, 134–139.
131. Engels, J.; Baek, D.D.; Kantamneni, S.; Tegmark, M. Scaling Laws For Scalable Oversight. *arXiv* **2025**, arXiv:2504.18530. [CrossRef]
132. Marks, S.; Treutlein, J.; Bricken, T.; Lindsey, J.; Marcus, J.; Mishra-Sharma, S.; Ziegler, D.; Ameisen, E.; Batson, J.; Belonax, T.; et al. Auditing language models for hidden objectives. *arXiv* **2025**, arXiv:2503.10965. [CrossRef]
133. Handa, K.; Tamkin, A.; McCain, M.; Huang, S.; Durmus, E.; Heck, S.; Mueller, J.; Hong, J.; Ritchie, S.; Belonax, T.; et al. Which Economic Tasks are Performed with AI? Evidence from Millions of Claude Conversations. *arXiv* **2025**, arXiv:2503.04761. [CrossRef]
134. Brynjolfsson, E.; Li, D.; Raymond, L. Generative AI at Work. *Q. J. Econ.* **2025**, *140*, 889–942. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.