

A SHAP-Guided Retrieval-Augmented Generation Framework for Predictive Maintenance Decision Support

AHMED AIT HADJ IAICH

ENSAM-Meknes: Universite Moulay Ismail Ecole Nationale Superieure d'Arts et Metiers

Smail ZAKI

smail.zaki@gmail.com

ENSAM-Meknes: Universite Moulay Ismail Ecole Nationale Superieure d'Arts et Metiers

<https://orcid.org/0000-0003-1586-6352>

Mohamed ABOUSSALEH

ENSAM-Meknes: Universite Moulay Ismail Ecole Nationale Superieure d'Arts et Metiers

Research Article

Keywords: Predictive maintenance, Explainable AI, Retrieval-augmented generation, SHAP-guided query construction, Machine learning, Industry 4.0

Posted Date: October 28th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11333020/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

A SHAP-Guided Retrieval-Augmented Generation Framework for Predictive Maintenance Decision Support

Ahmed AIT HADJ IAICH^a, Smail ZAKI^{a,1}, Mohamed ABOUSSALEH^a

^a *Engineering of Complex Systems and Structures Team ; Laboratory of Innovation and Systems Engineering ; Mechanical and Structural Engineering Department ; ENSAM, Moulay Ismail University, Meknes, MOROCCO.*

Abstract

Predictive maintenance models can reach high detection performance, but their outputs often remain difficult to operationalize in industrial environments, where maintenance decisions require interpretability and actionable guidance. Existing approaches rarely establish a direct link between failure predictions, model explanations, and maintenance-oriented recommendations. This study proposes a proof-of-concept framework that extends retrieval-augmented generation (RAG) to tabular predictive maintenance data through SHAP-guided query construction. A supervised machine learning model is trained to predict machine failures from operational data. SHAP explanations are then extracted to identify the dominant factors driving each prediction and converted into structured queries. These queries guide retrieval from an experimental maintenance knowledge base using embedding-based similarity search. The retrieved information is used by a generation module to produce case-specific maintenance recommendations aligned with the predictive output and its explanation. The framework is evaluated on a benchmark predictive maintenance dataset, where the predictive model shows strong failure detection capability, with particular improvement in recall for rare failure events. The retrieval module further indicates that explanation-derived queries can retrieve semantically relevant maintenance information. The results support the feasibility of integrating predictive modeling, SHAP-based explanation, retrieval, and LLM-based generation for maintenance-oriented interpretation. The proposed framework remains a proof of concept and does not yet constitute a fully validated industrial decision-support system.

Keywords: Predictive maintenance, Explainable AI, Retrieval-augmented generation, SHAP-guided query construction, Machine learning, Industry 4.0

1. Introduction:

Predictive maintenance (PdM) has gained substantial attention in industrial applications, particularly for failure detection and maintenance planning [1,2]. By applying machine learning to sensor and operational data, PdM systems can identify early degradation patterns and anticipate equipment failures before they disrupt production. Industrial deployment remains difficult. Many PdM models are developed under controlled conditions that only partially reflect production complexity. Class imbalance, sensor variability, operating regime changes, and machine heterogeneity can weaken robustness and limit generalization [1,3]. Transparency is another barrier. In high-stakes industrial environments, a failure prediction must be technically justifiable; black-box outputs are rarely sufficient for maintenance decisions [4].

Explainable artificial intelligence (XAI) methods, including LIME and SHAP, have been widely adopted to improve interpretability [5–7]. In PdM, these methods help engineers identify the variables that drive failure predictions and assess whether model behavior is consistent with known degradation mechanisms [8,9]. Yet most XAI approaches remain centered on feature attribution and visualization. They explain the prediction, but they do not systematically translate it into maintenance-relevant interpretation. Evaluation of explanation quality also remains challenging. Recent studies highlight issues related to robustness, consistency, and user interpretation [11]. Human-centered work further shows that explanations do not always improve decision quality and may increase over-reliance on AI outputs when trust is poorly calibrated [12,13].

This gap motivates the integration of predictive models, explainable AI, and retrieval-augmented generation (RAG) for maintenance decision support. Existing approaches rarely provide a systematic mechanism for connecting model explanations to relevant maintenance documentation. To address this limitation, this paper proposes a proof-of-concept framework that adapts RAG to tabular predictive maintenance data through SHAP-guided query construction. SHAP explanations are transformed into structured queries that guide retrieval from an experimental maintenance corpus using embeddings and vector-based similarity search. The retrieved information is then used by an LLM-based

* Corresponding author. Tel.: +212-6-67-95-38-67; fax: +212.5.35.46.71.63.
E-mail address: s.zaki@umi.ac.ma / smail.zaki@gmail.com

generation module to produce contextualized maintenance recommendations aligned with model outputs and domain knowledge.

The main contributions of this work are summarized as follows:

1. We formalize a SHAP-guided query construction method that converts feature attributions into structured retrieval queries.
2. We introduce a retrieval-based mechanism linking predictive model outputs and SHAP explanations with relevant maintenance knowledge.
3. We develop an end-to-end RAG-based decision-support pipeline connecting predictive outputs, model explanations, retrieval, and LLM-based generation.
4. We evaluate the framework through predictive performance metrics and retrieval relevance assessment.

2. Related Work:

2.1. Predictive Maintenance in Industrial Environments

Predictive maintenance has become a central component of Industry 4.0, supporting data-driven assessment of equipment health to anticipate failures and improve operational efficiency. Maintenance strategies have evolved from reactive interventions, where action follows failure, to preventive maintenance based on scheduled inspections, and then to predictive maintenance enabled by sensor data and machine learning. Recent work extends this trajectory toward explainable predictive maintenance systems connected to retrieval-augmented generation (RAG), where predictive models, interpretable explanations, and retrieved domain knowledge are combined to support maintenance decisions. This evolution is illustrated in Figure 1.

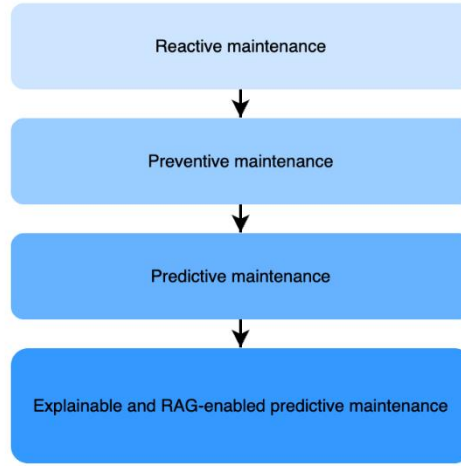


Figure 1: Evolution of maintenance strategies toward explainable predictive maintenance systems integrating retrieval-augmented generation

In predictive maintenance, machine learning models infer machine states from sensor measurements, operating conditions, and maintenance history. Carvalho et al. [1] show that supervised learning methods, including Random Forest, Support Vector Machines, and Artificial Neural Networks, remain widely used for fault detection and classification. Most studies emphasize predictive performance through metrics such as accuracy, precision, recall, and F1-score, while fewer address deployment constraints such as robustness, maintainability, and expert acceptance.

Deep learning has expanded the modeling capacity of predictive maintenance systems, particularly for multivariate time-series data. Lei et al. [2] review machinery health prognostics and show how convolutional and recurrent neural networks capture complex degradation patterns. These architectures are especially relevant for remaining useful life estimation, where temporal dependencies and progressive degradation dynamics must be represented. CNN-based feature extraction combined with LSTM-based temporal modeling has therefore become a common design in modern PdM applications.

Despite strong results on benchmark datasets, industrial adoption remains limited. Many studies rely on curated and preprocessed datasets that do not reflect the variability, missing data, regime changes, and cross-machine heterogeneity of production environments. Models trained on one machine, operating condition, or site often fail to generalize without adaptation. Predictive accuracy alone is therefore insufficient. Operational value also depends on whether model outputs can be interpreted, connected to maintenance knowledge, and translated into actions that engineers can trust and apply.

2.2. Challenges in Deploying Predictive Maintenance Models

The limited industrial adoption of high-performing predictive maintenance models is linked to several structural and organizational constraints. Failure events are scarce in most production environments, which creates strongly imbalanced datasets and weakens model reliability. Oversampling and data augmentation can mitigate this imbalance, but they may also introduce artificial patterns that reduce generalization under real operating conditions.

Industrial data pipelines add another layer of difficulty. Sensor drift, missing values, firmware updates, and hardware replacements can shift data distributions over time, creating training-serving skew and model staleness. Breck et al. [3] frame this problem as an engineering issue rather than a purely modeling issue. Their ML Test Score rubric emphasizes data validation, infrastructure reliability, monitoring, and rollback procedures, showing that production readiness depends on lifecycle control, not isolated optimization.

Maintenance integration is also socio-technical. Decisions involve safety, cost, downtime, and production continuity. A failure probability is rarely sufficient on its own. Engineers need to understand which signals drive the prediction and how these signals relate to mechanical degradation. Without this link, predictions remain difficult to trust and difficult to act on.

Trust is central in this context. Rudin [4] argues against the use of black-box models in high-stakes settings when interpretable alternatives are available. Predictive maintenance is such a setting: false decisions can lead to equipment damage, unnecessary intervention, or safety risks. Transparency, auditability, and technical traceability therefore become conditions for adoption.

These challenges show that predictive maintenance systems must combine reliability, interpretability, and operational integration. In this work, retrieval-augmented generation (RAG) is positioned as a translation layer between model outputs and maintenance decision support. By integrating SHAP-guided query construction, embedding-based retrieval, and LLM-based generation, the proposed approach connects tabular model explanations to relevant maintenance knowledge and turns predictions into actionable recommendations.

2.3. Explainable Artificial Intelligence for Industrial Decision Support

Explainable artificial intelligence (XAI) provides methods for interpreting complex machine learning models. Guidotti et al. [5] propose a taxonomy that distinguishes model-specific from model-agnostic approaches, and local from global explanations. Among model-agnostic methods, LIME [6] and SHAP [7] are widely used in industrial applications. LIME approximates local decision boundaries through surrogate linear models, while SHAP relies on Shapley values from cooperative game theory to estimate feature contributions.

In predictive maintenance, feature attribution methods are commonly used to identify the sensor variables driving a failure prediction. Reviews on explainable predictive maintenance [8,9] report the frequent use of SHAP and LIME in industrial settings. These methods help engineers verify whether predicted failures are consistent with known degradation mechanisms. Their output often remains limited to attribution scores or visual summaries, and may not capture temporal interactions in multivariate time-series data.

The systematic review on explainable AI for industrial fault diagnosis [10] shows that most studies focus on feature-importance visualization rather than structured interpretation or causal analysis. Evaluation also remains fragmented. Pawlicki et al. [11] compare several explainability metrics and show that fidelity, robustness, and stability can overlap without providing a unified basis for assessment. This lack of standardization limits comparison across studies.

Human-centered evaluation adds another constraint. Research on human-in-the-loop XAI for predictive maintenance [12] indicates that interactive explanations can improve perceived trust and usability. Explanations do

not necessarily improve decision accuracy. Bansal et al. [13] show that explanations may increase reliance on AI outputs even when predictions are wrong, highlighting the need for calibrated trust and context-aware explanation design.

Taken together, current XAI methods improve interpretability but often remain weakly connected to maintenance knowledge. Feature attributions alone do not directly translate into maintenance actions. This limitation motivates the use of retrieval-augmented generation (RAG) as a translation layer. In the proposed framework, SHAP-guided query construction converts tabular model explanations into retrieval queries, links them to relevant maintenance documents through embedding-based retrieval, and supports LLM-based generation of actionable recommendations.

2.4. Generative AI for Industrial Maintenance

Generative AI has attracted increasing interest in industrial maintenance because it can synthesize heterogeneous information and support decision-oriented outputs. Li et al. [14] identify four main applications in predictive maintenance: synthetic fault data generation, collaborative diagnosis with large language models, maintenance instruction generation, and knowledge graph integration. Generative models such as GANs and VAEs can help address class imbalance by producing synthetic fault samples, which is useful in rare-failure settings. Large language models offer a different capability: they enable natural language interaction with maintenance systems and can translate technical signals into explanations or procedural guidance.

Reliability remains a major constraint. Large language models may produce outdated, unsupported, or technically incorrect recommendations. In industrial maintenance, such errors can lead to unnecessary interventions, missed degradation mechanisms, or safety risks. For this reason, generation should not rely only on the parametric knowledge of the model. Retrieval-augmented generation (RAG) offers a more controlled architecture by coupling LLM-based generation with retrieval from domain-specific maintenance documents.

In the proposed framework, RAG is adapted to tabular predictive maintenance through SHAP-guided query construction. Model explanations are converted into structured retrieval queries, which are used to search a maintenance corpus through embedding-based retrieval. The retrieved documents then condition the generation module, allowing the system to produce recommendations that remain aligned with relevant maintenance knowledge. This integration of predictive modeling, SHAP-based explanation, retrieval, and LLM-based generation is illustrated in Figure 2.

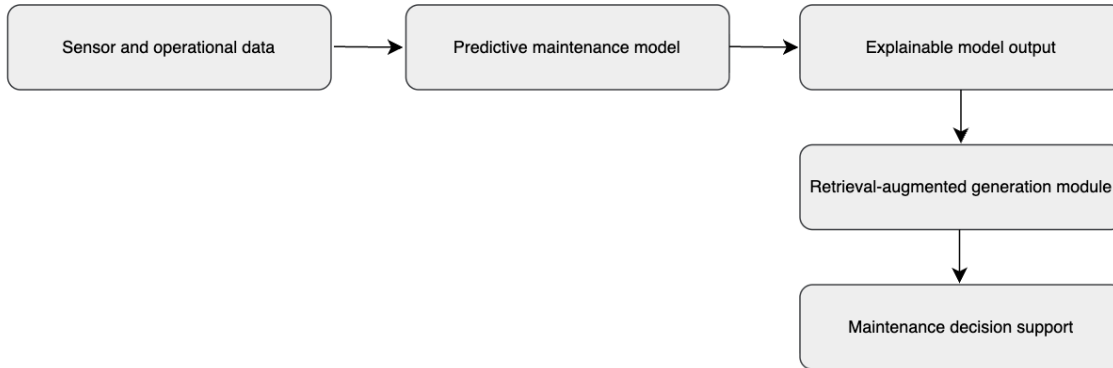


Figure 2: Conceptual architecture of the proposed explainable predictive maintenance framework with retrieval-augmented generation

2.5. Retrieval Augmented Generation for Knowledge Grounded AI

Retrieval-augmented generation (RAG) addresses a key limitation of purely parametric language models by combining document retrieval with LLM-based generation. Lewis et al. [15] define RAG as a hybrid architecture in which relevant documents are retrieved from an external corpus and used as contextual input for generation. This mechanism improves factual consistency and reduces unsupported outputs by constraining the response with retrieved evidence. The general architecture of a RAG system is shown in Figure 3.

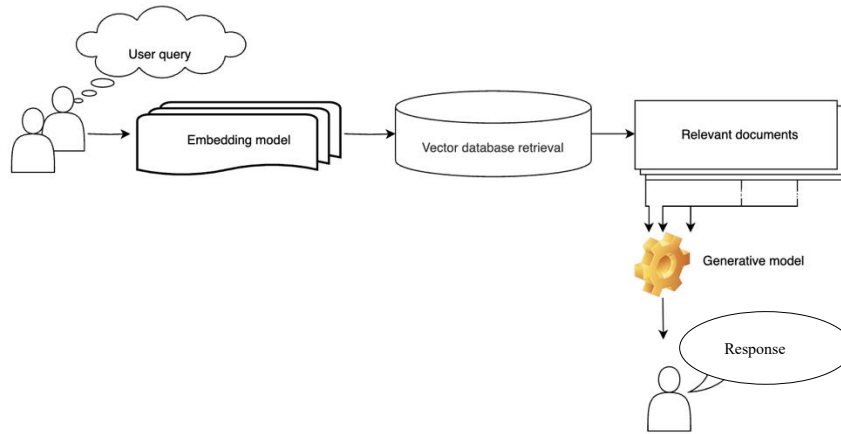


Figure 3: Architecture of a Retrieval-Augmented Generation (RAG) system for knowledge-grounded response generation

By incorporating external knowledge bases, RAG provides a practical mechanism for knowledge grounding, traceability, and interpretability. The broader RAG architecture formalizes this process through embedding models, vector databases, retrievers, and generators [16]. Retrieval refers to the selection of relevant documents through similarity search in the embedding space, while generation corresponds to the production of the final natural language response conditioned on the retrieved content. RAGAS [17] proposes reference-free evaluation metrics such as faithfulness, answer relevance, and context relevance to assess whether generated responses are supported by retrieved documents and whether retrieval aligns with the input query. These metrics are particularly relevant in industrial settings, where large-scale expert annotation is costly.

Despite its success in knowledge-intensive NLP tasks, RAG remains underexplored in predictive maintenance. Most PdM systems operate on numerical sensor data, whereas RAG relies primarily on textual corpora. The central challenge is therefore not generation alone, but knowledge grounding: linking predictive outputs and model explanations to relevant maintenance knowledge. Connecting SHAP-derived explanations with retrieval from maintenance documentation offers a way to translate numerical failure predictions into meaningful industrial interpretation.

2.6. Evaluation and Reliability of Industrial AI Systems

Reliability is a core condition for industrial AI adoption. The ML Test Score [3] frames this requirement as an engineering problem, covering data validation, model infrastructure, monitoring, and rollback mechanisms. Without such controls, predictive models can accumulate technical debt, drift from their training conditions, and lose operational validity over time.

Evaluation of explainability remains less mature. Pawlicki et al. [11] show that existing metrics capture different properties of explanation quality, such as fidelity, robustness, and stability, but do not yet provide a standardized assessment framework. RAGAS [17] addresses evaluation for retrieval-augmented generation systems by measuring whether generated outputs are supported by retrieved context. These evaluation streams remain largely separate.

Industrial AI requires broader validation. A deployable system must be assessed not only through predictive performance, but also through explanation fidelity, retrieval relevance, knowledge grounding, and production readiness. Few studies combine these dimensions within a single evaluation framework.

The literature reports substantial progress in predictive maintenance modeling [1,2], explainable AI methods [5–9], generative AI applications [14], and retrieval-augmented generation architectures [15–17]. These areas are still mostly treated independently. Predictive maintenance research often prioritizes model accuracy without systematically linking predictions to maintenance knowledge. XAI studies focus on feature attribution, but feature scores alone rarely

provide industrial interpretation. Retrieval-augmented generation improves document-supported output in NLP tasks, yet its use as a mechanism for connecting sensor-based predictions to maintenance documentation remains limited.

This gap motivates the present work. The proposed framework combines predictive maintenance, SHAP-based explanation, retrieval from a vector database, and LLM-based generation to support knowledge grounding in industrial decision support. The main contribution lies in linking model outputs to domain knowledge, while retrieval serves as the access mechanism and generation structures the final maintenance recommendation.

3. Methodology

This section describes the proposed framework integrating predictive maintenance (PdM), SHAP-based explainable artificial intelligence (XAI), and retrieval-augmented generation. The methodology follows a sequential process from sensor data processing to maintenance decision support, with the objective of linking predictive outputs with relevant domain knowledge.

3.1. System Overview

The framework combines three tightly coupled modules: a predictive model operating on numerical sensor data, a SHAP-based explainability layer, and a retrieval component embedded within a RAG pipeline.

Operational data are first processed by a supervised learning model that outputs both a binary decision (failure vs. non-failure) and an associated probability. SHAP is then applied at the instance level to decompose the prediction into feature-wise contributions. These contributions are not used for visualization alone. They drive a query construction step.

Specifically, the top-k features ranked by absolute SHAP values are selected. Each feature is paired with its contribution sign, capturing whether it increases or reduces failure risk. The resulting representation encodes operating conditions in a structured form aligned with failure mechanisms. These elements are aggregated into a query tailored for retrieval.

The retrieval stage extends standard RAG by incorporating SHAP-guided query construction adapted to tabular PdM data. Queries are embedded and matched against a vector database using cosine similarity. Documents are encoded with sentence embeddings (all-MiniLM-L6-v2). The top-k most similar entries are retrieved.

Generation follows. A GPT-4o-based model consumes the predictive output, SHAP-derived signals, and retrieved documents to produce maintenance recommendations. The generation is conditioned on retrieved content, ensuring alignment with domain knowledge while limiting unsupported outputs. A lightweight rule-based layer post-processes the response to enforce structure and consistency for industrial use.

This integration transforms raw predictions into interpretable and actionable outputs. It operationalizes the link between model explanations and retrieval, extending RAG to structured industrial data.

3.2. Data Sources

The framework relies on two complementary inputs: a numerical dataset for predictive modeling and a textual corpus for retrieval.

3.2.1. Sensor Dataset

The predictive model is trained and evaluated on the AI4I 2020 predictive maintenance dataset, a publicly available benchmark introduced by Matzka (2020) [18]. The dataset comprises 10,000 observations simulating industrial machine states. Each instance is described by operational variables, including air temperature, process temperature, rotational speed, torque, and tool wear, with a binary failure label as target.

The dataset reflects typical PdM constraints, including class imbalance and heterogeneous operating regimes. These characteristics enable controlled evaluation while preserving representative complexity. As the data are synthetic and not derived from real production systems, the experiments assess methodological validity rather than deployment readiness. Key variables and sample observations are reported in Tables 1 and 2.

Table 1: Sample observations from the AI4I predictive maintenance dataset.

Type	Air temperature [K]	Process temperature [K]	Rotational speed [rpm]	Torque [Nm]	Tool wear [min]	Machine failure
M	298.1	308.6	1551	42.8	0	0
L	298.2	308.7	1408	46.3	3	0
L	298.1	308.5	1498	49.4	5	0
L	298.9	309.1	2861	4.6	143	1
L	298.9	309.0	1410	65.7	191	1

Table 2: Key Features description.

Feature	Description
Type	Product type category
Air temperature (K)	Ambient air temperature
Process temperature	Temperature during operation
Rotational speed	Machine rotational speed
Torque	Mechanical torque applied
Tool wear	Accumulated tool wear
Machine failure	Binary machine failure label (1 = failure occurred, 0 = normal operation)

3.2.2. Maintenance Knowledge Corpus

The retrieval component relies on a compact, semantically structured maintenance corpus. Documents are not generated from direct mappings of feature values. Instead, they describe failure mechanisms in physical terms, aligned with established maintenance standards. This design enforces semantic matching between SHAP-derived conditions and domain knowledge rather than numerical overlap.

The corpus covers five failure modes from the AI4I dataset: tool wear, heat dissipation, power failure, overstrain, and random failure. Two documents are defined per mode, resulting in ten entries. Each document includes three fields: a failure condition, a recommended action, and a detailed description grounded in degradation physics. Content is derived from standards such as ISO 8688, ISO 13373-3, ISO 17359, and DIN 31051.

To decouple evaluation from the SHAP pipeline, query–document pairs were assessed by domain experts in machining maintenance. Relevance was rated on a three-level scale (relevant, partially relevant, not relevant), providing an external reference for retrieval performance. This protocol avoids self-referential evaluation, where query construction and validation rely on the same signals.

The corpus remains limited in size, consistent with a proof-of-concept objective. It enables controlled validation of SHAP-guided retrieval within a RAG setting. Scaling to industrial corpora and integrating real maintenance documentation are natural next steps. Example entries are provided in Table 3.

Table 3: Example entries from the maintenance knowledge corpus.

ID	Failure condition	Recommended action	Maintenance description	Source
----	-------------------	--------------------	-------------------------	--------

50	Power transmission anomaly: low torque combined with abnormally high rotational speed, suggesting drivetrain instability or load decoupling.	Verify shaft alignment and inspect flexible couplings. Check belt tension and power transmission components before resuming full-load operation.	A significant mismatch between torque and rotational speed is a recognized indicator of drivetrain instability, potentially caused by coupling wear, belt slippage, or load decoupling. In such conditions, energy transfer efficiency decreases and mechanical fatigue on transmission components accelerates. Early diagnostic inspection of the power transmission chain is recommended to prevent progressive degradation and unplanned downtime.	ISO 13373-3 — Vibration condition monitoring and diagnostics
77	Progressive cutting tool degradation approaching end-of-life threshold, consistent with advanced wear-phase behavior.	Inspect cutting edge geometry. Replace tool if wear land exceeds defined threshold. Perform a validation cut before resuming normal production.	Cutting tool degradation follows a well-documented three-phase wear curve: initial break-in, steady-state wear, and rapid deterioration. As accumulated wear progresses into the final phase, surface finish quality degrades, cutting forces increase, and thermal stress on the workpiece rises. Timely replacement when wear exceeds manufacturer-defined limits is essential to maintaining process stability and machining quality.	ISO 8688 — Tool life testing in milling
160	Combined mechanical overstrain: elevated torque load concurrent with advanced tool wear, indicating compounded stress on the machining system.	Reduce operating load to 60–70% of nominal capacity. Inspect spindle bearings and replace worn tooling. Perform a test cut before returning to full production.	The simultaneous presence of high cutting torque and advanced tool wear creates compounded mechanical stress on the machining system. Elevated torque increases radial and axial forces on spindle bearings, while degraded tooling amplifies vibration and reduces process damping. This combination accelerates fatigue crack propagation in both the tool and the spindle assembly. Following maintenance fundamentals outlined in DIN 31051, immediate load reduction and full inspection of critical components are recommended.	DIN 31051 — Fundamentals of maintenance

3.3. Data Preprocessing

Data preprocessing prepares the sensor dataset for predictive modeling while ensuring consistency with the explainability and retrieval stages of the retrieval-augmented generation (RAG) framework. Categorical variables, such as product type, are encoded using one-hot encoding. Numerical features are kept in their original scale, as the selected models are insensitive to feature scaling.

Model evaluation relies on a repeated stratified k-fold cross-validation scheme. The outer loop uses 5 folds repeated 3 times, reducing sensitivity to a single split and stabilizing generalization estimates. Stratification preserves the proportion of failure events across folds, which is essential under class imbalance. An inner stratified k-fold ($k = 4$) is used for hyperparameter tuning. Randomized search samples candidate configurations from predefined distributions, allowing efficient exploration without exhaustive computation.

Class imbalance is handled through class weighting at the model level, avoiding explicit resampling. Performance is assessed using metrics suited to imbalanced classification, including recall, F1-score, and balanced accuracy. Recall for the failure class is prioritized, as missed failures carry higher operational cost than false positives.

Preprocessing also supports the RAG pipeline. For each prediction, SHAP values are computed to quantify feature contributions. The most influential features are selected based on absolute SHAP values and transformed into structured representations. This SHAP-guided query construction step directly feeds the retrieval module, establishing a consistent link between model explanations and retrieval in a tabular PdM setting.

3.4. Predictive Maintenance Model

The predictive maintenance module estimates the probability of machine failure from operational data. Given the tabular structure and moderate dataset size, ensemble tree-based methods are selected. Two complementary approaches are considered: Random Forest (bagging) and Gradient Boosting (boosting).

Random Forest aggregates independently trained decision trees on bootstrapped samples, improving robustness and reducing variance. Gradient Boosting builds trees sequentially, each focusing on correcting previous errors, allowing the model to capture more complex patterns.

Both methods are compatible with SHAP-based explainability through efficient implementations such as Tree SHAP, which provide consistent feature attribution with manageable computational cost. Models are trained within the evaluation framework defined in Section 3.3, including randomized hyperparameter optimization and class weighting. Each model outputs a failure probability, converted into a binary decision using a threshold selected from validation results. A comparative analysis is presented in Section 4 to identify the most suitable model for integration within the RAG pipeline.

3.5. Explainable AI Module

SHAP is applied to quantify feature contributions to the predicted failure probability at the instance level. The resulting attribution highlights the dominant drivers of each prediction. These drivers are not used solely for interpretation. They define the basis of query construction.

The most influential features are transformed into structured descriptors that capture both operating conditions and their impact on failure risk (e.g., high torque, increased tool wear, elevated temperature). These descriptors form the input to the retrieval stage, enabling alignment between model-level explanations and relevant maintenance knowledge.

3.6. SHAP-Guided Retrieval-Augmented Generation Module

The retrieval-augmented generation (RAG) module establishes a direct link between instance-level SHAP explanations and domain-specific maintenance knowledge. It operates through three stages: document indexing, SHAP-guided query construction with retrieval, and generation. The transformation from SHAP outputs to structured queries is formalized in Algorithm 1.

Input:

- x: input sample
- $f(x)$: trained predictive model
- SHAP(x): SHAP values for input sample
- k: number of top features to select

Output:

- q: structured query for retrieval

Steps:

1. Compute prediction:
 $y_{pred} = f(x)$
2. Compute SHAP values:
 $shap_values = SHAP(x)$
3. Select top-k contributing features:
 $F_k = \text{top } k \text{ features ranked by } |shap_values|$
4. Initialize query representation:
 $q = []$
5. For each feature f_i in F_k :
 - a. Retrieve feature value v_i
 - b. Determine contribution direction:
if $shap_values[f_i] > 0$:
direction = "increasing failure risk"

```

else:
    direction = "reducing failure risk"
c. Map feature to semantic descriptor:
    d_i = semantic_mapping(f_i, v_i, direction)
d. Append descriptor to query:
    q.append(d_i)
6. Infer high-level failure hypothesis:
    H = infer_failure_type(F_k)
7. Construct final query:
    q_final = concatenate (H, q)
Return q_final

```

In this study, k is set to 3, meaning that the three most influential variables are selected according to their absolute SHAP values. The `semantic_mapping` function converts each selected feature into a maintenance-oriented descriptor by combining the feature name, its observed value, its SHAP contribution direction, and its physical interpretation. For example, a positive SHAP contribution associated with torque is converted into a descriptor such as “high torque increasing failure risk”, while a positive contribution associated with tool wear is represented as “advanced tool wear increasing failure risk”. This mapping prevents the retrieval query from remaining a raw numerical description and reformulates the model explanation in terms closer to maintenance reasoning.

The `infer_failure_type` function assigns a high-level failure hypothesis using predefined rules derived from the failure mechanisms represented in the maintenance corpus. For instance, dominant tool wear contributions are associated with tool wear failure (TWF), the joint dominance of torque and tool wear is associated with overstrain failure (OSF), and dominant temperature-related variables are associated with heat dissipation failure (HDF). These rules are not intended to replace the predictive model. They provide a structured intermediate representation that connects SHAP explanations to retrieval.

During document indexing, each entry in the maintenance corpus is encoded into a dense vector using the all-MiniLM-L6-v2 sentence embedding model. The embeddings are stored in a vector database to support efficient similarity search. This representation preserves the semantic content of maintenance descriptions, allowing retrieval to rely on conceptual proximity rather than lexical overlap.

Query-driven retrieval follows the SHAP-guided query construction step. The structured query is embedded in the same vector space using the same sentence encoder, ensuring consistency between query and document representations. Cosine similarity is computed between the query embedding and the corpus embeddings, and the top- k most similar documents are returned. When SHAP highlights torque and tool wear as dominant contributors, retrieval prioritizes documents describing mechanical overstrain or tool degradation, consistent with corpus entries derived from DIN 31051 and ISO 8688. This integration between SHAP-derived conditions and embedding-based retrieval defines the core retrieval mechanism of the proposed RAG adaptation to tabular predictive maintenance data.

The generation stage integrates the predicted failure probability, SHAP explanation, and retrieved documents. A GPT-4o-based model generates maintenance recommendations using a structured prompt conditioned on these inputs. The generation is conditioned on the retrieved content to reduce unsupported outputs and maintain alignment with the retrieved maintenance information. A lightweight rule-based post-processing step enforces a standardized format, including failure assessment, likely mechanism, and prioritized maintenance actions.

The module does not operate as an autonomous decision system. Generated outputs are designed to support maintenance operators and require validation by qualified personnel before deployment.

3.7. Maintenance Decision Support

The final stage builds on the retrieval-augmented generation (RAG) pipeline by integrating predictive outputs, SHAP-based explanations, and retrieved maintenance knowledge into a structured decision support output. The complete workflow is illustrated in Figure 4.

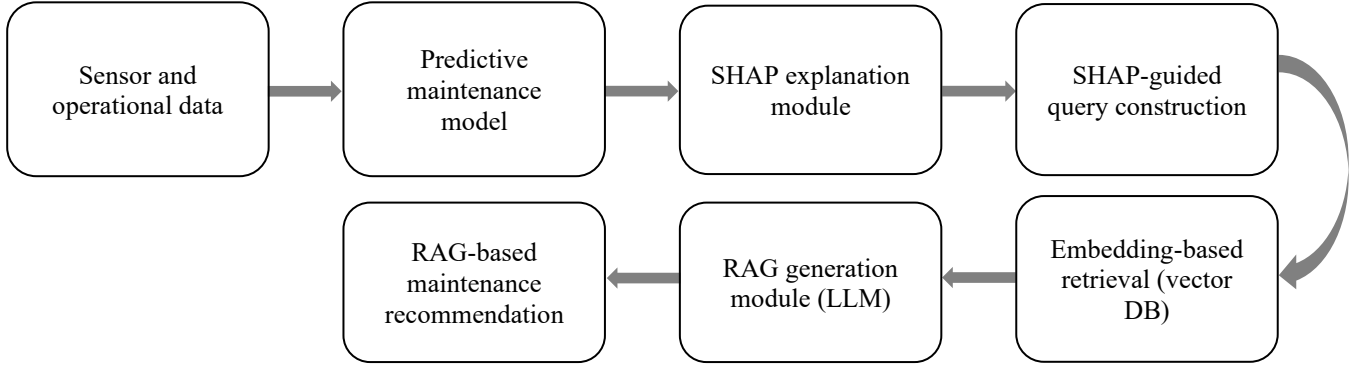


Figure 4: Conceptual architecture of the proposed SHAP-guided retrieval-augmented generation (RAG) framework for predictive maintenance

Sensor and operational data are first processed by the predictive maintenance model, which estimates the probability of machine failure and produces a binary classification. For each predicted failure, SHAP computes instance-level feature contributions, identifying the dominant operational drivers. These contributions are converted through a SHAP-guided query construction step into a structured natural language query, as defined in Section 3.5, encoding failure-relevant conditions in terms of physical degradation mechanisms rather than raw numerical values.

The resulting query is passed to the retrieval module, which searches a vector database using cosine similarity over dense sentence embeddings. The top-k documents are selected based on semantic proximity and provide the domain context for generation. This step illustrates the adaptation of RAG to tabular predictive maintenance data, where structured explanations directly guide retrieval.

The generation stage combines the predicted failure probability, SHAP explanations, and retrieved documents within a GPT-4o-based model to produce maintenance recommendations through a structured prompt. The output is standardized via a rule-based post-processing step into three components: failure risk assessment, probable failure mechanism inferred from SHAP contributions, and prioritized maintenance actions aligned with retrieved knowledge.

The architecture ensures traceability across the pipeline. Predictions are supported by instance-level explanations and connected to relevant maintenance knowledge through retrieval, then transformed into actionable outputs through generation. This integration moves beyond feature attribution toward decision-oriented outputs, addressing a known limitation in explainable predictive maintenance.

4. Results and discussion

4.1. Class Distribution

The predictive maintenance dataset exhibits a strong class imbalance, with 9,661 instances of normal operation and 339 failure events, corresponding to only 3.4% of the data. This distribution reflects realistic industrial conditions, where failures occur infrequently compared to normal operation.

Such imbalance introduces a bias toward the majority class and can significantly degrade the model’s ability to detect failure events. Therefore, the objective is not limited to maximizing overall accuracy, but rather to improve failure detection performance, particularly by reducing false negatives.

In this context, recall for the failure class is considered a key evaluation metric, as undetected failures may lead to unexpected breakdowns and production downtime

4.2. Model Training and Evaluation

Random Forest and Gradient Boosting models, introduced in Section 3.4, are trained using the available process variables and engineered features. Feature engineering captures interactions between operational parameters, including the difference between process and air temperature, which reflects thermal load during operation. Model

evaluation relies on stratified cross-validation to preserve class distribution across folds. Precision, recall, and F1-score are used to assess performance under imbalance. Recall remains the primary metric, as it measures the ability to detect critical failure events. Results are compared against the Bagged Trees classifier reported in the original AI4I study [18], used as a baseline.

4.3. Model Performance and Comparison with Baseline

Performance is evaluated using stratified cross-validation to preserve class distribution across folds. Precision, recall, and F1-score are used as evaluation metrics. Table 4 summarizes the comparison between Random Forest, Gradient Boosting, and the baseline Bagged Trees model reported in the original AI4I study [18].

Table 4: Performance comparison of the evaluated models and the baseline method reported in [18].

Model	Precision	Recall	F1-score
Bagged Trees (baseline)	0.867	0.708	0.779
Random Forest	0.805	0.898	0.85
Gradient Boosting	0.89	0.79	0.84

The results show distinct trade-offs. Random Forest achieves the highest recall (0.898) and F1-score (0.85), significantly improving failure detection over the baseline (recall of 0.708). Gradient Boosting reaches the highest precision (0.89), reducing false positives but missing more failure events.

In predictive maintenance, false negatives correspond to undetected failures. Their cost exceeds that of false positives. The confusion matrix in Figure 5 confirms that Random Forest reduces false negatives compared to the baseline.

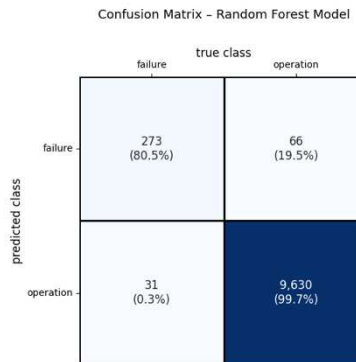


Figure 5: Confusion matrix of the Random Forest predictive maintenance model

The increase in false positives remains limited and acceptable in this context, where early detection is preferred over missed failures. Random Forest is therefore selected as the predictive model for the retrieval-augmented generation framework, as it provides the strongest failure detection performance. Its probability outputs and SHAP explanations directly support the SHAP-guided query construction step, ensuring consistent integration between prediction, retrieval, and generation.

4.4. SHAP Based Explainability for Failure Interpretation

In the proposed retrieval-augmented generation (RAG) framework, SHAP is used beyond post-hoc interpretation. It provides the basis for SHAP-guided query construction. For each prediction, the most influential features are selected from their SHAP values and converted into explanatory signals. These signals encode both relative importance and contribution direction to the predicted failure risk, such as high rotational speed, high torque, increased tool wear, or abnormal temperature difference.

This structured representation connects the predictive model to the retrieval module. It guides document search toward maintenance knowledge relevant to the observed operating conditions. In this setting, SHAP acts as an interface between tabular model explanations and embedding-based retrieval, enabling the adaptation of RAG to

predictive maintenance data. The framework links data-driven predictions to retrieved maintenance knowledge and then to generation of actionable recommendations.

Figure 6 presents the SHAP summary plot of the Random Forest model, showing the global contribution and directional effect of the most influential variables on machine failure prediction.

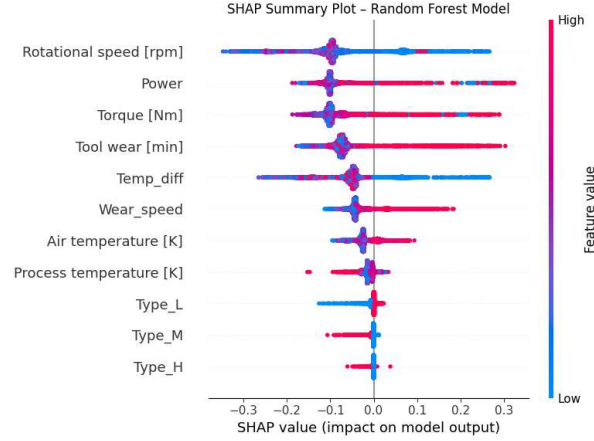


Figure 6: SHAP summary plot of the proposed Random Forest model showing the contribution of the most influential variables to machine failure prediction

4.5. SHAP-Guided Retrieval-Augmented Generation for Maintenance Interpretation

After the SHAP-based explainability layer, the retrieval-augmented generation (RAG) module converts instance-level explanations into structured maintenance recommendations. For each predicted failure, the dominant SHAP contributors are mapped to semantic descriptors using the `semantic_mapping` function defined in Section 3.6. The `infer_failure_type` rule set then associates these descriptors with a candidate failure mechanism. This structured representation forms the query construction step. The query is submitted to the retrieval module, which selects the most semantically relevant maintenance documents using cosine similarity over sentence embeddings.

Table 5 reports representative outputs from the proposed framework. The examples show how SHAP-derived signals are converted into retrieval-oriented queries and then used during generation to produce maintenance recommendations. For normal operation, retrieval is skipped and the output remains limited to monitoring and preventive maintenance. For failure cases, such as tool wear failure or overstrain failure, the retrieved documents provide the technical context used by the generation module to formulate specific inspection, replacement, load reduction, and validation actions.

Table 5: Representative examples of SHAP-guided RAG outputs for predictive maintenance

ID	Prediction	Key SHAP Factors	Failure Hypothesis	Generated Output (Summary)
0	Normal operation	No dominant failure-related SHAP factors	No failure detected	The system indicates stable machine operation. No failure indicators are detected. Recommended actions include continuous monitoring and routine preventive maintenance.
77	Machine failure	High tool wear, high wear-speed	Tool wear failure (TWF): progressive cutting tool degradation approaching end-of-life threshold	The model identifies progressive cutting tool degradation consistent with advanced wear-phase behavior. The cutting tool is approaching its mechanical end-of-life threshold, with accelerated wear rate confirmed by high wear-speed contribution. Recommended actions: inspect cutting edge geometry, measure wear land, replace the tool if the defined threshold is exceeded, and perform a validation cut before resuming normal production.
160	Machine failure	High torque, high tool wear, low speed	Overstrain failure (OSF): compounded mechanical stress on spindle and	The model detects a mechanical overstrain condition caused by the co-occurrence of excessive torque load and advanced tool wear, further confirmed by reduced

tooling due to simultaneous high torque and advanced tool wear

rotational speed indicating increased cutting resistance. This configuration imposes compounded stress on the spindle assembly and accelerates fatigue crack propagation. Recommended actions: reduce operating load to 60–70% of nominal capacity, inspect spindle bearings, replace worn tooling, and perform a controlled test cut before returning to full production.

The retrieval module was evaluated on all predicted failure cases. Given the strong class imbalance, retrieval is activated only for instances classified as machine failure; normal operating conditions bypass retrieval. The Normal Skip Rate of 1.0 confirms that the system correctly suppresses retrieval for non-failure predictions, avoiding the introduction of failure-related maintenance content into normal-operation outputs. Retrieval performance was assessed using two complementary protocols, summarized in Table 6.

Table 6: Quantitative evaluation of the SHAP-guided RAG module

Metric	Protocol	Value
Normal Skip Rate	Internal consistency	1
Average Similarity@3	Internal consistency	0.71
Hit Rate@3	Internal consistency	0.87
Precision@3	Internal consistency	0.69
Distractor Rejection Rate	Independent expert annotation	1
Expert Relevance Rate (score = 2)	Independent expert annotation	0.81
Expert Relevance Rate (score ≥ 1)	Independent expert annotation	0.905
Mean inter-annotator agreement (κ)	Independent expert annotation	0.709

The internal consistency evaluation relies on embedding-based metrics computed over the predicted failure cases. Average Similarity@3 reaches 0.71, indicating semantic proximity between SHAP-derived queries and retrieved corpus entries. Hit Rate@3 is 0.87, meaning that at least one relevant document is retrieved in 87% of failure cases. Precision@3 is 0.69, showing that roughly two of the three retrieved documents per query are consistent with the inferred failure mechanism. These results support the coherence of the retrieval step within the shared embedding space.

An independent expert annotation protocol was also used to avoid a self-referential evaluation of retrieval quality. A stratified sample of 10 query–document pairs was selected across the three main failure modes represented in the dataset: power failure (PWF), tool wear failure (TWF), and overstrain failure (OSF). Three distractor pairs were included to test annotation discrimination. Six domain experts in industrial maintenance rated each pair on a three-point scale: 0 for not relevant, 1 for partially relevant, and 2 for relevant.

All annotators assigned a score of 0 to the three distractor pairs, corresponding to a Distractor Rejection Rate of 1.0. For the seven non-distractor pairs, the Expert Relevance Rate reached 81.0% for fully relevant assessments and 90.5% when partial relevance was included. The mean pairwise Cohen’s kappa was 0.709, indicating substantial agreement according to the Landis and Koch scale. Individual kappa values ranged from 0.600 to 1.000. The lowest agreement appeared mainly on pairs 4 and 5, both related to tool wear failure, where experts differed on whether the retrieved document sufficiently covered the fracture risk scenario. This disagreement points to a concrete direction for refining the maintenance corpus.

Overall, the embedding-based metrics and expert annotations indicate that the SHAP-guided retrieval mechanism produces semantically coherent and practically relevant outputs under controlled experimental conditions. The results

also clarify the role of the proposed RAG adaptation: SHAP explanations from tabular PdM data are not only interpreted, but converted into queries that drive retrieval and condition LLM-based generation. The current annotation sample remains limited, and broader validation on real industrial maintenance documentation is still required.

5. Discussion

The results raise three connected points: the behavior of predictive models under severe class imbalance, the role of SHAP-guided query construction beyond conventional explainability, and the practical value of expert validation for retrieval-augmented generation (RAG) in predictive maintenance.

Random Forest improves recall compared with the baseline Bagged Trees classifier, which is consistent with the operational asymmetry of predictive maintenance. A missed failure is usually more costly than a false alarm. This supports the choice to prioritize recall, even with a moderate increase in false positives. Class weighting was retained instead of resampling to avoid introducing artificial patterns that could weaken generalization. Gradient Boosting shows a slight advantage in precision, reflecting a more conservative decision behavior, but this is less suitable when early detection is the main objective.

The main contribution of the explainability layer is not the use of SHAP alone. SHAP is already widely used for interpreting predictive maintenance models. Its role here is different: SHAP outputs are converted into structured retrieval queries through semantic mapping and failure-type inference. This SHAP-guided query construction step transforms numerical feature contributions into physical degradation descriptors that can interact with a maintenance corpus. The approach adapts RAG to tabular predictive maintenance data by using model explanations to drive retrieval, rather than relying on raw sensor similarity or generic document search. This integration between explanations and retrieval is the core mechanism evaluated in the retrieval experiments.

The expert annotation results support the practical relevance of this mechanism. The fully relevant retrieval rate reaches 81.0%, with a mean inter-annotator agreement of $\kappa = 0.709$. These values indicate that the retrieved documents were generally judged meaningful by maintenance experts. The unanimous rejection of distractor pairs by all six annotators further suggests that the evaluation was conducted with adequate discrimination. The disagreement observed for tool wear failure is also informative. It shows that progressive wear and sudden fracture risk may require separate corpus entries, as they represent different degradation sub-mechanisms. Refining the corpus along these lines would likely improve retrieval specificity.

More broadly, the study suggests that the main limitation in explainable predictive maintenance is not only predictive accuracy, but the translation of model outputs into usable maintenance knowledge. The proposed RAG framework addresses this gap by linking SHAP-derived explanations to retrieval and then to LLM-based generation. The proof-of-concept setting limits the scope of the claims, but the agreement between embedding-based retrieval metrics and independent expert judgment provides initial evidence that SHAP-guided retrieval is technically feasible and practically relevant. The contribution is therefore best understood as an architectural blueprint for future work using real industrial datasets, larger expert-annotated corpora, and adaptive retrieval strategies.

6. Conclusion and Perspectives

This work addresses a central limitation of explainable predictive maintenance: the lack of a systematic mechanism connecting model-level explanations to actionable maintenance knowledge. The proposed retrieval-augmented generation (RAG) framework combines three components: a Random Forest classifier reaching a recall of 0.898 on rare failure events, a SHAP-based explainability module structured through semantic mapping and failure-type inference, and a RAG module supported by a maintenance corpus derived from internationally recognized standards.

A key contribution lies in the formalization of the SHAP-to-query transformation. Numerical feature contributions are converted into physical degradation descriptors that can be used for retrieval from a domain knowledge base constructed independently of the sensor dataset. This design reduces the circularity often found in retrieval evaluation, where query construction and relevance assessment are produced by the same pipeline.

The retrieval module was evaluated through two complementary protocols. Embedding-based metrics show semantic alignment between SHAP-derived queries and corpus documents, with a Hit Rate@3 of 0.87 and an Average Similarity@3 of 0.71. Independent annotation by six domain experts produced a fully relevant retrieval rate of 81.0% and a mean inter-annotator agreement of $\kappa = 0.709$. These results indicate that the retrieved evidence and the resulting generated recommendations are largely consistent with expert maintenance judgment under the conditions tested.

The findings suggest that the main barrier to deployable explainable predictive maintenance is not only predictive accuracy, but the translation of model outputs into operational guidance. The proposed framework provides an architectural blueprint showing that this translation can be achieved by integrating SHAP-guided query construction, embedding-based retrieval, and LLM-based generation. The scope remains limited by the proof-of-concept setting and the use of a synthetic benchmark dataset.

Future work should follow three directions. First, the framework should be validated on real industrial datasets with authentic sensor variability, cross-machine heterogeneity, and production constraints. Second, the maintenance corpus should be expanded using real documentation and larger expert annotation protocols, especially to cover finer failure sub-mechanisms such as distinct tool wear scenarios. Third, re-ranking strategies and agent-based reasoning architectures could be investigated to improve contextual adaptation and support operators in more complex industrial environments.

Declarations

- a. Funding: the authors declare that no funds, grants, or other support were received during the preparation of this manuscript.
- b. Financial interests: the authors have no relevant financial or non-financial interests to disclose.
- c. Availability of data and material: The dataset used in this study is publicly available (AI4I predictive maintenance dataset).
- d. Code availability: Not applicable
- e. Ethics approval: Not applicable
- f. Consent to participate: Not applicable
- g. Consent for publication: Not applicable

References

- [1] Carvalho TP, Soares FAAMN, Vita R, Francisco R da P, Basto JP, Alcalá SGS. A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering* 2019;137:106024. <https://doi.org/10.1016/j.cie.2019.106024>
- [2] Lei Y, Li N, Guo L, Li N, Yan T, Lin J. Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mechanical Systems and Signal Processing* 2018;104:799–834. <https://doi.org/10.1016/j.ymssp.2017.11.016>
- [3] Breck E, Cai S, Nielsen E, Salib M, Sculley D. The ML test score: A rubric for ML production readiness and technical debt reduction. In: *Proceedings of the IEEE International Conference on Big Data*; 2017
- [4] Rudin C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 2019;1:206–215. <https://doi.org/10.1038/s42256-019-0048-x>
- [5] Guidotti R, Monreale A, Ruggieri S, Turini F, Giannotti F, Pedreschi D. A survey of methods for explaining black box models. *ACM Computing Surveys* 2018;51(5):93. <https://doi.org/10.1145/3236009>
- [6] Ribeiro MT, Singh S, Guestrin C. "Why should I trust you?": Explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*; 2016. p. 1135–1144. <https://doi.org/10.1145/2939672.2939778>
- [7] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*; 2017. p. 4768–4777
- [8] Cummins L, Sommers A, Ramezani SB, Mittal S, Jabour J, Seale M, Rahimi S. Explainable predictive maintenance: A survey of current methods, challenges and opportunities. *IEEE Access* 2024. <https://doi.org/10.1109/ACCESS.2024.3391130>
- [9] Zemmouchi-Ghomari L. Explainable AI for predictive maintenance: A review and standardized evaluation framework. *Management Science Letters* 2026. <https://doi.org/10.5267/j.msl.2025.11.001>
- [10] Cação J, Santos J, Antunes M. Explainable AI for industrial fault diagnosis: A systematic review. *Journal of Industrial Information Integration* 2025;47:100905. <https://doi.org/10.1016/j.jii.2025.100905>
- [11] Pawlicki M, Pawlicka A, Uccello F, Szelest S, D'Antonio S, Kozik R, Choraś M. Evaluating the necessity of the

- multiple metrics for assessing explainable AI: A critical examination. *Neurocomputing* 2024;602:128282. <https://doi.org/10.1016/j.neucom.2024.128282>
- [12] Amaliah NR, Tjahjono B, Palade V. Human-in-the-loop XAI for predictive maintenance: A systematic review of interactive systems and their effectiveness in maintenance decision-making. *Electronics* 2025;14(17):3384. <https://doi.org/10.3390/electronics14173384>
- [13] Bansal G, Wu T, Zhou J, Fok R, Nushi B, Kamar E, Ribeiro MT, Weld D. Does the whole exceed its parts? The effect of AI explanations on complementary team performance. In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*; 2021. Article 81, p. 1–16. <https://doi.org/10.1145/3411764.3445717>
- [14] Li X, Wang P, Li X, Lee J, Zhou Q, Jun MBG, Gao RX. Generative AI for predictive maintenance for manufacturing: A survey. *Procedia CIRP* 2026;139:35–40. <https://doi.org/10.1016/j.procir.2025.09.012>
- [15] Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, Küttler H, Lewis M, Yih WT, Rocktäschel T, Riedel S, Kiela D. Retrieval-augmented generation for knowledge-intensive NLP tasks. In: *Advances in Neural Information Processing Systems (NeurIPS)*; 2020.
- [16] Klesel M, Wittmann HF. Retrieval-augmented generation (RAG). *Business & Information Systems Engineering* 2025;67:551–561. <https://doi.org/10.1007/s12599-025-00945-3>
- [17] Es S, James J, Espinosa Anke L, Schockaert S. RAGAs: Automated evaluation of retrieval augmented generation. In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations (EACL)*; 2024. p. 150–158. <https://doi.org/10.18653/v1/2024.eacl-demo.16>
- [18] Matzka S. Explainable artificial intelligence for predictive maintenance applications. In: *2020 Third International Conference on Artificial Intelligence for Industries (AI4I)*; 2020. <https://doi.org/10.1109/AI4I49448.2020.00023>