Proc. of the European Safety and Reliability Conference (ESREL 2026)
Edited by José C. Matos, Paulo B. Lourenço, Michael Beer, Daniel Oliveira, Edoardo Patelli, Hélder S. Sousa, Rui A. Silva and Monica Santamaria

doi: 10.3850/ESREL2026061419_esrel26-p27820-cd

Agentic AI Framework for Automated Work Order Generation in Mining Drill-Rig Maintenance

Melwin Xavier
Luleå tekniska universitet, Sweden. E-mail: melwin.xavier@ltu.se

Ramin Karim
Luleå tekniska universitet, Sweden. E-mail: ramin.karim@ltu.se

High availability is essential for mining drill-rig operations, and unplanned downtime in critical subsystems directly disrupts production. Although predictive analytics can flag anomalies in streaming sensor data, converting those alerts into actionable, CMMS-ready maintenance work orders remains largely manual. Maintenance planners must interpret alerts, consult documentation, and encode decisions in CMMS with uneven consistency. This paper presents an agentic framework that automates work-order drafting through structured collaboration among four specialised agents: a Workflow Orchestrator, an Anomaly Detection Agent, a Diagnostic Agent, and a Work Order Generation Agent. Diagnostic reasoning is grounded in a domain Knowledge Graph and augmented with Retrieval-Augmented Generation over technical documentation, enabling evidence-linked recommendations. The prototype is evaluated on five representative failure scenarios spanning major drill-rig subsystems, and generated work orders are compared with procedure-derived ground truth using automated metrics (field completeness, diagnosis accuracy, action coverage, part recall). A hydraulic pump seal-wear walkthrough illustrates agent interactions and the resulting structured work order. The framework establishes a traceable and reproducible interface between predictive maintenance outputs and industrial work management systems.

Keywords: Agentic AI, Knowledge Graph, RAG, Predictive Maintenance

1. Introduction

Mobile drill rigs are essential to mining production, and the availability of their subsystems largely determines blast-cycle throughput. Degradation in pumps, feed cylinders, variable-frequency drives, or control modules can rapidly escalate into unplanned downtime, with substantial economic consequences and safety implications (Muminov et al., 2022). Condition-based monitoring and machine-learning anomaly detection have advanced to the point that pressure, temperature, vibration, and flow telemetry can be analysed in near real time to flag incipient faults (Compare et al., 2019). However, operational benefit depends on what follows an alert: the deviation must be interpreted, equipment documentation consulted, component criticality assessed, and a work order authored within a Computerised Maintenance Management System (CMMS). This post-alert translation remains labour-intensive, variable in quality, and frequently delays corrective action.

LLM-based agentic architectures indicate that tool-mediated, multi-step reasoning can coordinate complex decision workflows in modular and auditable forms (Yao et al., 2022). Retrieval-Augmented Generation (RAG) improves factual grounding by conditioning generation on retrieved documentation (Lewis et al., 2020), while constraining reasoning with knowledge-graph structure can further strengthen traceability from component context to recommended actions in knowledge-intensive domains (Pan et al., 2024). These capabilities motivate automated work-order drafting, provided that evidence transparency and human oversight are embedded in the workflow.

The present study proposes an agentic framework that links condition-monitoring outputs to CMMS-ready maintenance work orders for mining drill-rig maintenance. The workflow coordinates four specialised agents a Workflow Orchestrator, an Anomaly Detection Agent, a Diagnostic Agent, and a Work Order Generation Agent via structured, typed message passing. Diagnosis is grounded in a domain Knowledge Graph (KG) aligned with ISO 14224 equipment taxonomy and complemented by RAG over technical documentation, ensuring that recommendations carry traceable evidence links. Generated work orders are mapped to standard CMMS field schemas and routed for human review before execution. The approach is evaluated across five representative failure scenarios spanning major drill-rig subsystems and is validated against procedure-derived ground truth using automated metrics. Section 2 reviews related work; Section 3 presents the methodology; Section 4 reports the case study; Section 5 discusses limitations; and Section 6 concludes.

2. Related Work

Predictive maintenance in mining and heavy industry has shifted from periodic, calendar-based inspection toward telemetry-driven strategies, but translating detection outputs into operational planning and CMMS work orders remains challenging (Muminov et al., 2022; Compare et al., 2019). CMMS databases constitute a large, underexploited repository of maintenance knowledge. Work-order records contain free-text descriptions of faults and corrective actions authored by technicians, yet empirical studies consistently report data-quality issues missing fields, inconsistent vocabulary, and ambiguous task descriptions that hinder reliable analytics (Conte et al., 2021). NLP and text-mining methods have been applied to classify, cluster, and extract patterns from these records. Li et al. (2024) developed an NLP pipeline that automates analysis and assignment of maintenance work orders in hospital facilities and reported improved classification accuracy relative to manual triage. Naqvi et al. (2024) proposed a technical language processing pipeline using BERT-based semantic search to extract insights from noisy industrial maintenance text, with specific attention to abbreviations, jargon, and terse writing styles that challenge general-purpose NLP models. These approaches improve processing of existing work orders but do not solve the upstream problem of generating a new, structured order directly from sensor anomaly evidence.

Large language models have created new possibilities for maintenance decision support because they can synthesise heterogeneous information, reason over multi-step procedures, and produce human-readable outputs. However, standalone inference is susceptible to hallucination plausible but incorrect assertions which is unacceptable in safety-critical settings. Retrieval-Augmented Generation mitigates this risk by anchoring generation in retrieved passages from a curated document store, improving factual grounding (Lewis et al., 2020). Combining RAG with structured knowledge graphs can further strengthen traceability and consistency: knowledge graphs encode validated relationships among equipment, failure modes, symptoms, and corrective actions, enabling constrained traversals that restrict candidate hypotheses to domain-valid options (Pan et al., 2024). Bahr et al. (2025) demonstrated a KG-enhanced RAG approach for Failure Mode and Effects Analysis and showed that graph-structured retrieval outperforms flat vector search for tasks that require relational reasoning over engineering artefacts.

Multi-agent architectures powered by LLMs have also attracted attention as a mechanism for decomposing complex industrial tasks into specialised, verifiable sub-tasks. Gong et al. (2025) introduced a difficulty-aware multi-agent framework for sensor-fault reasoning in smart manufacturing, where a routing module selects either direct resolution or deliberative debate among multiple LLM agents, achieving substantial gains over single-model baselines. Park et al. (2025) deployed a four-agent system on commercial CNC machine tools that interpreted alarms, retrieved relevant manual passages via RAG, and executed corrective actions, demonstrating the feasibility of LLM-based autonomous maintenance in production environments. Despite these advances, an end-to-end, auditable pipeline that orchestrates anomaly detection, KG- and RAG-grounded diagnosis, and CMMS-schema-compliant work-order generation within a unified agentic workflow remains underdeveloped. The present study addresses this gap.

3. Methodology

The workflow is designed as an event-driven, human-in-the-loop pipeline that converts condition monitoring evidence into structured maintenance work orders. Figure 1 outlines four specialised agents coordinated by a Workflow Orchestrator. Time-stamped sensor streams (e.g., pressure, temperature, flow, vibration) are ingested with basic validation and assessed by the Anomaly Detection Agent using configurable statistical baselines and failure signatures. When a deviation exceeds magnitude thresholds and persists for at least a minimum duration, the agent emits an AnomalyAlert that records quantified sensor deviations, severity, and confidence. The Diagnostic Agent consumes the alert and performs root-cause inference by combining two knowledge sources: (i) a domain Knowledge Graph (KG) encoding equipment–failure mode–symptom relationships and maintenance actions, and (ii) Retrieval-Augmented Generation (RAG) over unstructured technical documentation. The agent returns a DiagnosisReport with ranked hypotheses, evidence links, and recommended confirmation steps. Finally, the Work Order Generation Agent transforms the report into a CMMS-compatible WorkOrder draft that specifies priority, required actions, resources, scheduling guidance, and safety considerations. Draft work orders are routed for human review and approval rather than being executed automatically.

The framework is intentionally loosely coupled: agents interact solely through typed messages. Messages and knowledge references are appended to an audit trace to enable after-action review and governance. This design supports modular replacement of components, traceability through persisted alerts and evidence pointers with confidence values, and explicit oversight via draft WorkOrder artifacts.

Although an end-to-end model could be constructed, the present work adopts a multi-agent decomposition to (i) support stage-level verification (detection, diagnosis, generation), (ii) separate governance and audit concerns (attribution of intermediate artifacts), (iii) enable specialised optimisation (statistical detection vs. KG/RAG-grounded reasoning vs. templated order formatting), and (iv) insert human review gates between stages without redesigning the full pipeline.

3.1. *Agent Specifications*

Inter-agent communication is implemented through typed schemas that specify inputs and outputs, bounded enumerations (e.g., severity and priority), and explicit confidence fields for uncertainty propagation. Messages include identifiers, timestamps, and correlation metadata so the Workflow Orchestrator can enforce policy and maintain an end-to-end trace.

3.1.1. *Workflow Orchestrator Agent*

The Workflow Orchestrator routes messages between agents, maintains per-asset case state, and applies workflow policy (validation, timeouts/retries, escalation, and human-review gates). The orchestrator implements the event-driven state machine and persists an append-only audit trace of inputs, outputs, and evidence pointers.

3.1.2. *Anomaly Detection Agent*

The Anomaly Detection Agent monitors multivariate sensor streams and emits AnomalyAlert messages when deviations exceed configured magnitude and persistence thresholds. Alerts report per-sensor deviations (value, baseline, unit, Z-score), overall severity and confidence, and optional multi-sensor signature labels (Compare et al., 2019).

An anomaly confidence score $C_{\mathrm{anomaly}} \in [0, 1]$ is computed as a weighted combination of deviation magnitude, multi-sensor agreement, and signature matching:

$$C_{\mathrm{anomaly}} = \alpha \cdot \frac{|Z_{\max}|}{Z_{\mathrm{thr}}} + \beta \cdot \frac{n_{\mathrm{exc}}}{n_{\mathrm{tot}}} + \gamma \cdot s_{\mathrm{sig}}, \qquad \alpha + \beta + \gamma = 1, \tag{1}$$

Here, $Z_{\max}$ denotes the maximum absolute Z-score across monitored sensors, Z_{thr} is the alert threshold, n_{exc} counts sensors exceeding the threshold among n_{tot} available channels, and $s_{\mathrm{sig}} \in [0, 1]$ quantifies the match to known failure signatures (e.g., correlated pressure,temperature,level

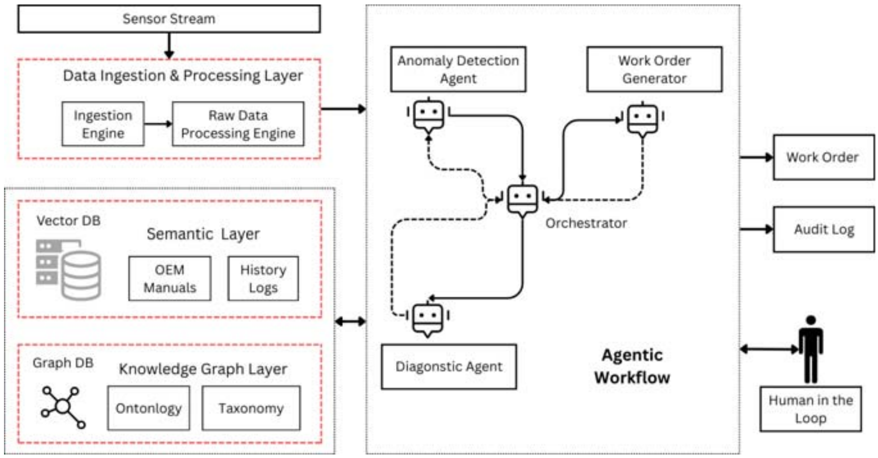

Fig. 1. Agentic Framework for Work Order Creation.

deviations). The computed value is clipped to $[0, 1]$ prior to propagation.

3.1.3. *Diagnostic Agent*

The Diagnostic Agent outputs a DiagnosisReport containing ranked hypotheses, confidence values, evidence pointers (kg_paths,rag_references), and confirmation steps. Candidate causes are generated via KG-guided traversals (component→failure mode→symptoms) and complemented by RAG retrieval over manuals, bulletins, and logs; the language model synthesises and ranks candidates while validating that proposed failure modes are applicable to the component class.

Diagnosis confidence $C_{\text{diagnosis}} \in [0, 1]$ fuses evidence from the KG, the retrieved documentation, and symptom consistency:

$$C_{\text{diagnosis}} = w_{\text{kg}} \cdot s_{\text{kg}} + w_{\text{rag}} \cdot s_{\text{rag}} + w_{\text{sym}} \cdot c_{\text{sym}}, \quad w_{\text{kg}} + w_{\text{rag}} + w_{\text{sym}} = 1, \tag{2}$$

where s_{kg} is a normalized relevance score for KG traversals supporting a hypothesis, s_{rag} is the average semantic similarity of the top-k retrieved passages, and c_{sym} is symptom coverage (fraction of expected symptoms present). The weights can be calibrated using the ground-truth evaluation described in Section 4.

Table 1. Indicative mapping between generated fields and common CMMS schemas.

Generated field	SAP PM	Maximo
`work_order_id`	AUFNR	WONUM
`priority`	PRIOK	WOPRIORITY
`equipment.asset_id`	EQUNR	ASSETNUM
`equipment.location`	TPLNR	LOCATION
`problem_description`	KTEXT	DESCRIPTION
`required_actions[]`	AFVC	WOTASK
`resources.parts[]`	RESB	WPMATERIAL
`resources.personnel`	ARBPL	CRAFTRATE

3.1.4. *Work Order Generation Agent*

The Work Order Generation Agent converts diagnoses into CMMS-compatible WorkOrder drafts for planner review. Priority is derived from safety impact, equipment criticality, and diagnosis confidence; ordered actions are assembled from procedure templates and KG-linked maintenance actions; the draft summarises required parts and labour; and a scheduling window is recommended with supporting justification. Outputs set human review required to enforce oversight. To illustrate deployability, Table 1 maps core WorkOrder fields to representative CMMS schemas; field names remain indicative and may vary by site configuration.

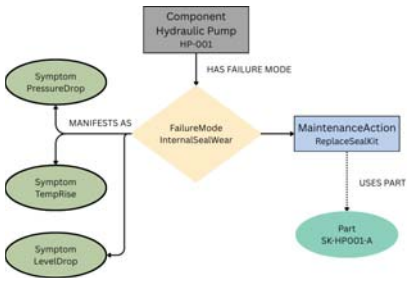

Fig. 2. Example KG subgraph linking equipment, failure mode, symptoms, maintenance actions, and required parts.

3.2. *Knowledge Integration Layer*

3.2.1. *Knowledge Graph Design*

The KG serves as the structured grounding layer for diagnosis and action selection. The ontology is aligned with ISO 14224 equipment taxonomy (Franciosi et al., 2025; Wang et al., 2024) and uses node types Equipment, Component, FailureMode, Symptom, MaintenanceAction, and Part. Core edge types include HAS_COMPONENT, HAS_FAILURE_MODE, MANIFESTS_AS, REQUIRES_ACTION, and USES_PART. Figure 2 illustrates a sample subgraph for a hydraulic pump seal-wear scenario, linking symptoms to a failure mode and to a recommended action with required parts.

During diagnosis, the KG is queried through constrained traversals that restrict candidates to failure modes valid for the identified component class. Returned paths are serialized as node sequences (stored in kg_paths) so the causal chain (component→failure mode→symptoms) remains inspectable during review. Edge attributes can encode expected symptom direction, onset latency, and historical prevalence, enabling weighted scoring rather than simple reachability. For work-order generation, the KG also links failure modes to recommended maintenance actions and required parts, supporting consistent action sequencing and parts planning.

The KG is populated from the asset registry (equipment hierarchy and metadata), FMEA documentation (failure modes, symptoms, severity/criticality attributes), and standard maintenance procedures (actions and parts). Table 2 summarises the failure distribution used to prioritise subsystems during model development and case-study selection.

Table 2. Illustrative failure distribution across major systems and dominant cause types.

System Category	Count	Primary Cause
Hydraulic System	78	Wear (67%)
Feeder System	32	Comp. failure (44%)
Media (Air/Water)	24	Wear (58%)
Electrical System	22	Comp. failure (45%)
Control System	19	Comp. failure (53%)
Drill Machine	13	Wear (46%)

Table 2 reports a synthetic distribution, created for illustration and aligned with ISO 14224 taxonomy; counts are not derived from an operational CMMS export.

3.2.2. *RAG Pipeline*

RAG complements the KG by enabling semantic retrieval over OEM manuals, service bulletins, and historical logs. During diagnosis, a query derived from symptoms and component metadata retrieves the top-k passages; passage identifiers are stored as rag_references so outputs remain auditable and grounded. In the prototype implementation, passages are chunked into 512-token segments, embedded with a GPT embedding model, and retrieved with $k = 5$ using cosine similarity.

4. Case Study

To assess generality beyond a single subsystem, the framework is evaluated on five representative failure scenarios spanning major drill-rig systems (Table 3). For each scenario, simulated sensor deviations are generated to match the expected failure signature and are mapped to the KG ontology described in Section 3.

4.1. *Validation Against Ground Truth*

A structured "gold" work order is derived for each scenario using documented maintenance procedures and troubleshooting guidance (e.g., isolation steps, inspection checks, required parts). Generated

Table 3. Prototype evaluation scenarios across major subsystems.

ID	System	Failure mode	Key symptoms
S1	Hydraulic	Pump internal seal wear	Pressure drop, case-drain temp rise, reservoir level drop
S2	Electrical	VFD overheating	Current spike, thermal alarm, reduced motor speed
S3	Feeder	Feed cylinder rod seal leak	Feed rate drop, position drift, visible external leak
S4	Control	PLC sensor input fault	Erratic readings, comm. timeout, diagnostic code
S5	Drill machine	Rotation motor bearing wear	Vibration increase, temperature rise, audible noise

outputs are compared against this ground truth using automated metrics computed from structured fields: (i) *Field completeness* (% required CMMS fields populated), (ii) *Diagnosis accuracy* (match to the ground-truth failure mode), (iii) *Action coverage* (fraction of standard procedure steps included), and (iv) *Part recall* (fraction of required parts correctly identified).

Across scenarios, field completeness reaches 100% and diagnoses match the ground-truth failure modes. Action coverage ranges from 0.78–0.89, and part recall ranges from 0.60–0.80.

Ablation variants are used to attribute performance to components: KG-only (no retrieval), RAG-only (no KG), and the full system. The KG-only configuration strengthens applicability checks and CMMS-aligned terminology, whereas RAG-only produces richer procedural text. The combined system provides the strongest traceability by linking explicit KG paths with retrieved documentation excerpts.

4.2. *Detailed Walkthrough: Hydraulic Pump Seal-Wear (Scenario S1)*

Scenario S1 considers a main hydraulic pump on a mining drill rig operating underground. The pump supplies hydraulic power for drilling, feed, and boom actuation. Under nominal conditions, the pump outlet pressure is approximately 3200 PSI, the case-drain temperature is around 55 °C, and the hydraulic reservoir level is stable at 100%.

Table 4. Illustrative sensor readings for hydraulic pump seal wear.

Time	Press. (PSI)	Case temp (°C)	Oil (%)	Status
T0	3200	55	100	Normal
T2 (+12 h)	2650	72	88	Exceeded

The simulated failure mode is progressive internal seal wear, which increases internal leakage from the high-pressure side to the case drain. In operational settings, early indications commonly include pressure loss under load accompanied by elevated case-drain temperature due to increased recirculation and heating. As leakage advances, hydraulic reservoir level decreases and oil accumulation may appear in the hydraulic bay, creating reliability and housekeeping/safety concerns.

Table 4 reports representative readings at two time points: a baseline state (T0) and an anomalous state (T2) in which thresholds are exceeded. Thresholds are configured relative to the learned asset baseline: a pressure drop $>$15%, a case-drain temperature increase $>$10 °C, or a reservoir level decrease $>$10% triggers an alert.

At T2, a sensor snapshot is submitted to the Workflow Orchestrator. Agents are invoked in sequence, and intermediate artifacts are persisted to support auditability. Key outputs are summarized below.

Step 1 (Detection). Using the T2 snapshot, the Anomaly Detection Agent computes deviations (e.g., Pressure $z = -3.8$, Case temp $z = +4.2$, Level $z = -3.1$) and emits an AnomalyAlert AL-2026-02-05-HP001-001 with severity CRITICAL (confidence 0.91).

Step 2 (Diagnosis). The Diagnostic Agent combines KG traversal (e.g., HP-001$\rightarrow$InternalSealWear) with RAG retrieval over manuals to produce a DiagnosisReport selecting InternalSealWear (0.87) and a confirmation check (inspect case-drain flow).

Step 3 (Work order). The Work Order Generation Agent translates the diagnosis into a CMMS-compatible WorkOrder draft with URGENT priority, ordered actions, and resource requirements for planner review.

4.2.1. *Generated Work Order Output*

The draft work order illustrates the structured output format and the traceable linkage to upstream evidence. A schema-valid WorkOrder message is returned:


```
{
  "work_order_id": "WO-2026-0152",
  "priority": "URGENT",
  "equipment": {
    "asset_id": "HP-001",
    "location": "Drill Rig DR-42, Hydraulic Bay"
  },
  "problem_description": "AI-diagnosed internal
       seal wear causing hydraulic oil leak (
      pressure drop, case-drain temperature
      rise, reservoir level loss).",
  "required_actions": [
    "Isolate hydraulic pump from system (lock-
        out/tag-out).",
    "Replace seal kit (SK-HP001-A).",
    "Reassemble, pressure-test, and verify leak-
        free."
  ],
  "resources": {
    "parts": ["SK-HP001-A", "HYD_OIL_ISO_VG46_L
        "],
    "personnel": {
      "skill_required": "Hydraulic Technician",
      "headcount": 2,
      "estimated_hours": 4.0
    }
  },
  "human_review_required": true
}
```


5. Discussion

This paper introduces an agentic framework that provides a reproducible path from sensor-level anomaly evidence to CMMS-ready work-order drafts in mining drill-rig maintenance. Coordinating four specialised agents through structured messages yields intermediate artifacts (AnomalyAlert, DiagnosisReport, WorkOrder) annotated with confidence values and evidence links, supporting auditability and downstream integration. The modular decomposition allows components (e.g., detectors or retrieval stacks) to evolve behind stable interface contracts while preserving governance and human-review gates.

Several limitations and deployment considerations remain. The evaluation relies on simulated sensor deviations and procedure-derived ground truth, and scaling requires continued KG and documentation engineering together with integration testing for site-specific CMMS configurations. Key risks include LLM hallucination (mitigated through KG/RAG grounding, schema validation, and human-review gates), adversarial or faulty sensor inputs (addressed through sensor validation and plausibility checks against historical bounds), KG incompleteness (fallback to retrieval-only mode with a confidence penalty), and latency/connectivity constraints in underground environments (edge deployment and asynchronous queuing).

6. Conclusion

This paper presents an agentic architecture for automating work-order generation in mining maintenance. By coordinating specialised agents via typed messages and grounding generation in domain knowledge (KG and RAG), the framework produces structured, evidence-linked work orders from sensor anomalies. Prototype evaluation across five failure scenarios demonstrates generality, and automated comparison against procedure-derived ground truth provides quantitative validation without requiring expert scoring. Future work will prioritise validation with operational data, expansion of the failure-mode library, and benchmarking against maintenance planners in production CMMS workflows.

References

Bahr, L., C. Wehner, J. Wewerka, J. Bittencourt, U. Schmid, and R. Daub (2025). Knowledge graph enhanced retrieval-augmented generation for failure mode and effects analysis. *Journal of Industrial Information Integration 45*, 100807.

Compare, M., P. Baraldi, and E. Zio (2019). Challenges to iot-enabled predictive maintenance for industry 4.0. *IEEE Internet of things journal 7*(5), 4585–4597.

Conte, A., C. Bolland, L. Phan, M. Brundage, and T. Sexton (2021). The impact of data quality on maintenance work order analysis: A case study in historical hvac maintenance work orders. In *PHM Society European Conference*, Volume 6, pp. 11–11.

Franciosi, C., Y. Eslami, M. Lezoche, and A. Voisin (2025). Ontologies for prognostics and health management of production systems: overview and research challenges. *Journal of Intelligent Manufacturing 36*(4), 2223–2253.

Gong, W., S. Qiao, C. Cao, S. Tan, J. Ye, H. Liu, S. Chen, and X. Wang (2025). Harnessing collective intelligence of multi-agent llm systems for sensor failure reasoning in smart manufacturing. *Journal of Industrial Information Integration*, 101012.

Lewis, P., E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems 33*, 9459–9474.

Li, Y., Y. Liu, J. Zhang, L. Cao, and Q. Wang (2024). Automated analysis and assignment of maintenance work orders using natural language processing. *Automation in Construction 165*, 105501.

Muminov, R., D. Kuziev, V. Zotov, and E. Sazankova (2022). Performability of electro-hydro-mechanical rotary head of drill rig in open pit mining: A case-study. *Eurasian Mining 1*, 76–80.

Naqvi, S. M. R., M. Ghufran, C. Varnier, J.-M. Nicod, K. Javed, and N. Zerhouni (2024). Unlocking maintenance insights in industrial text through semantic search. *Computers in Industry 157*, 104083.

Pan, S., L. Luo, Y. Wang, C. Chen, J. Wang, and X. Wu (2024). Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering 36*(7), 3580–3599.

Park, J., S. Cho, Y. Chae, S. N. Durgunlu, and J. Um (2025). Llm-based multi-agent system for autonomous maintenance process of machine tools. In *PHM Society Asia-Pacific Conference*, Volume 5.

Wang, Y., Y. Cheng, Q. Qi, and F. Tao (2024). Ids-kg: An industrial dataspace-based knowledge graph construction approach for smart maintenance. *Journal of Industrial Information Integration 38*, 100566.

Yao, S., J. Zhao, D. Yu, N. Du, I. Shafran, K. R. Narasimhan, and Y. Cao (2022). React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.