A Knowledge-Graph-Guided Diagnostic Pipeline for Root Cause Analysis of Equipment Failures in Nuclear Power Plants

D. Mandelli[1] and C. Wang[1]

[1] *Idaho National Laboratory, Idaho Falls, ID 83415, USA*
diego.mandelli@inl.gov
congjian.wang@inl.gov

ABSTRACT

The root cause analysis of complex equipment failures in nuclear power plants requires the integration of heterogeneous data sources into traceable causal explanations. It is a process that can be time-consuming, expert-dependent, and difficult to audit across plant lifetime. This paper presents a structured diagnostic reasoning framework designed to co-pilot system engineers to perform root cause analyses, improving analysis speed, consistency, and traceability while preserving full engineering interpretability. We have formally decomposed a causal reasoning architecture that integrates five independent but interacting reasoning dimensions (i.e., structural, temporal, evidence-based, governance, and historical) allowing each dimension of the diagnostic process to be explicitly validated. The framework integrates several heterogeneous inputs such as telemetry signals, plant documentation, system architecture models, and operational context within a unified knowledge representation. Structural reasoning constrains the causal search space by identifying physically plausible failure modes based on system connectivity and component dependencies. Temporal reasoning characterizes the relationships between observed anomalies and the main event using interval-based representations to distinguish candidate causes from coincident or downstream effects. Evidence-based reasoning retrieves and evaluates relevant historical records, classifying them as supporting, contradicting, or contextual with respect to each candidate hypothesis. Candidate causes are generated and evaluated across multiple dimensions, including structural plausibility, temporal consistency, telemetry alignment, and documentary evidence strength, with explicit tracking of uncertainty and conflicting evidence. A synthesis stage produces an assessment that includes ranked hypotheses, supporting evidence with traceable references, identified uncertainties, and recommended follow-up actions. A key design principle of the framework is that causal conclusions are only promoted when multiple independent sources of evidence align, reducing the risk of premature convergence on incomplete or biased explanations. The framework is evaluated through mechanism-level validation across three representative nuclear plant failure scenarios covering condenser degradation, reactor trip temporal gating, and surveillance check valve leakthrough.

1. INTRODUCTION

A nuclear power plant processes thousands of condition reports (CRs) every year. Each CR documents an equipment anomaly, degradation finding, or functional failure and must be dispositioned through a corrective action program (CAP) that includes, for significant items, a formal root cause analysis (RCA). The RCA must identify not only what failed but why it failed (i.e., tracing from proximate cause through contributing factors to systemic organizational root causes) and must document the evidence chain supporting each conclusion.

Analysts performing this work manually execute three cognitive tasks that are both critical and error-prone. First, they must bound the hypothesis space: given a reported component, determine which failure modes are physically plausible based on equipment type, system configuration, and upstream-downstream dependencies. Second, they must align signal timing: determine which process anomalies preceded the failure, which were contemporaneous, and which were downstream effects that should not be treated as causes. Third, they must retrieve analogous records: search historic CRs, work orders (WOs), procedures, and fleet operating experience (OE) databases for similar failure histories that inform the current analysis.

The International Atomic Energy Agency TECDOC-1756 program review of operating experience utilization identifies three recurring failure modes in manual RCA practice: investigative fixation (premature convergence on the first plausible hypothesis, suppressing alternatives), causal coverage gaps (incomplete search across causal categories), and incon-

sistent use of plant history (high analyst-to-analyst variability in evidence retrieval and weighting) (IAEA, 2014). These are not primarily technical failures; they are information-retrieval and reasoning failures that structured automation can address.

DACKAR (Digital Analytics, Causal Knowledge Acquisition and Reasoning) [1] is a 10-stage deterministic pipeline designed around three core objectives. First, it constrains the hypothesis space to what the plant equipment model supports, generating failure mode candidates from a structured knowledge graph (KG) (Hogan et al., 2021) of components, failure modes, topology relationships, and safety function definitions (ensuring every hypothesis is physically grounded and traceable rather than inferred from unstructured text or analyst intuition). Second, it enforces systematic coverage of all required investigative dimensions: for each event, the pipeline must either populate or explicitly rule out each of twelve defined causal categories (ranging from equipment-internal causes through human performance and organizational factors), with gaps surfaced as structured attention flags rather than silently omitted. Third, it integrates the plant's documentary and historical record directly into the investigation (indexing CRs, WOs, prior RCA findings, and OE documents in a vector store and retrieving them per candidate to characterize whether a failure mode is a novel occurrence or a recurrence of a known pattern). The pipeline produces two primary outputs: an `rca_card` presenting ranked hypotheses, evidence citations, and corrective action recommendations organized at three causal depths (proximate, contributing, and root cause); and a `run_manifest` recording the complete audit trail of available data, degraded inputs, and open analyst review items.

The composite scoring function integrates five reasoning dimensions: structural (KG-derived topology priors and failure mode plausibility), temporal (Allen interval alignment and failure mode and effects analysis -FMEA- latency qualification), evidence-based (documentary retrieval with source-tier weighting), governance (preventive maintenance -PM- compliance and maintenance program adequacy), and historical (past event similarity and recurrence characterization). These dimensions directly address the three RCA failure modes identified above: structural and temporal reasoning constrain investigative fixation and coverage gaps; evidence-based, governance, and historical reasoning address inconsistent use of plant history.

This paper makes three contributions: first, a 10-stage pipeline with typed interstage artifacts that enable unit testing and analyst inspection at any point in the diagnostic chain. Second, a deterministic composite scoring architecture that integrates the five reasoning dimensions, decoupling candidate ranking from language model synthesis and producing fully reproducible scores. Third, an initial mechanism validation across three representative failure scenarios demonstrating that each of the five reasoning dimensions operates as designed, with counterfactual confirmation for the ranking-determinative mechanisms.

2. RELATED WORK

Ishikawa diagrams, TapRoot, and fault and event trees impose discipline on hypothesis generation and provide a documented evidence chain (Latino & Latino, 2006; Rooney & Vanden Heuvel, 2004). These methods are systematic but static: the analyst must supply all hypotheses, retrieve all evidence, and apply all causal logic manually. They do not scale to high-CR-volume environments and produce analyst-dependent results.

Bayesian network approaches model causal uncertainty with principled probability distributions and support formal sensitivity analysis (Weber, Medina-Oliva, Simon, & Iung, 2012; Cai, Huang, & Xie, 2017). Their primary limitation in operational settings is the maintenance burden: conditional probability tables require quantification from historical data or expert elicitation, and the network structure must be updated as the asset fleet evolves. Sparse failure data in nuclear applications further weakens the learned distributions.

Autoencoder-based reconstruction error detectors, isolation forests, and LSTM sequence models applied to plant historian data are effective at flagging unexpected signal behavior (Hu, Zhang, Yang, & Chen, 2020). A broad review of AI and ML applications in operating nuclear plants (U.S. Nuclear Regulatory Commission (NRC) & Idaho National Laboratory (INL), 2022) confirms that these methods are most mature for anomaly flagging and condition monitoring. They do not, however, produce causal explanations: anomaly flagging is a necessary input to RCA but is not a substitute for it.

NLP pipelines applied to nuclear event reports and maintenance narratives have demonstrated the ability to extract causal relationships from operating experience text (Kim, Kim, Lee, Kim, & Diesner, 2024; Wang, Mandelli, & Cogliati, 2024). These approaches address the evidence retrieval problem but do not enforce temporal consistency, do not generate hypotheses from a structured equipment model, and do not produce reproducible ranked conclusions. LLM-based diagnostic tools generate plausible causal narratives from failure descriptions (Lin, Zhang, Fu, & Liu, 2025) but lack temporal rigor, do not enforce causal category coverage, and produce non-reproducible outputs whose variability complicates compliance with regulated CAP audit trail requirements. Hybrid retrieval-augmented generation approaches exist in general-domain question answering (Lewis et al., 2020) but have not been applied to the formal multi-level RCA problem in nuclear operations.

[1] GitHub repository: https://github.com/idaholab/DACKAR

The IAEA guidance on AI deployment in nuclear (IAEA, 2025) recommends architectures that anchor data-driven methods to plant topology and structured knowledge models rather than end-to-end black-box predictors, framing AI support as advisory and human-supervised with explicit auditability. DACKAR is designed within this framework: hypothesis generation is governed by the plant KG, scoring is fully deterministic, and the natural language processing components (causal extraction, named entity recognition -NER-) feed into the deterministic scoring layer rather than driving conclusions directly. The result is a pipeline that simultaneously enforces temporal gating, multisource evidence alignment with structured provenance, and a deterministic audit trail independent of any language model (three properties that existing methods address individually but have not, to the authors' knowledge, been combined in a single auditable pipeline for the formal multi-level RCA problem in regulated nuclear operations).

3. PIPELINE ARCHITECTURE

3.1. Subsystem Overview

DACKAR is organized around five subsystems that exchange typed artifacts through the 10-stage pipeline. The 10 stages define the execution workflow: the order in which inputs are consumed, candidates are generated, and outputs are produced. The five reasoning dimensions (structural, temporal, evidence-based, governance, and historical) define the scoring framework that operates within that workflow: each dimension contributes a normalized score to the composite ranking computed at Stage E. The two views are complementary, not alternative descriptions of the same thing. Each subsystem directly addresses one or more of the three RCA reasoning failure modes identified in the introduction: investigative fixation, causal coverage gaps, or inconsistent use of plant history. The KG subsystem is a Neo4j graph database encoding plant equipment topology, component-level failure mode libraries, upstream-downstream system dependencies, and safety function assignments. It bounds the hypothesis space and provides structural prior scores for each candidate failure mode. The Chroma evidence store subsystem indexes CRs, WOs, PM procedures, completed RCAs, and fleet OE records for hybrid semantic and lexical retrieval. The causality engine generates and scores causal candidates by integrating evidence from all five reasoning dimensions into a composite score. The synthesis module accepts the scored candidate list and generates a structured narrative report, with language model generation as an optional augmentation layer. The orchestrator manages stage sequencing, interstage artifact validation, attention flag accumulation, and conditional branching at the auto-reentry stage.

3.2. 10-Stage Pipeline

Figure 1 shows the pipeline architecture. The five subsystems that drive the stages (KG, causality engine, evidence store, synthesis module, and orchestrator) are described in detail in Sections 4–5.5; their stage assignments are summarized in the figure caption. Each stage produces a named, schema-validated artifact consumed by subsequent stages; the typed-artifact system is described in Section 4.3.

Stage A (input validation) accepts the raw event record (equipment tag, failure description, event timestamp, and available signal references) and performs temporal alignment across historian sources. It applies a KG governance check to confirm the reported equipment tag exists in the graph and that the failure mode vocabulary is consistent. Outputs are a validated event record and a set of attention flags for any alignment gaps.

Stage B (KG context build) traverses the neighborhood of the reported equipment node in the KG, collecting upstream dependencies, downstream dependents, failure modes associated with the component type, and linked safety functions. The output is a `kg_context` artifact that constrains all subsequent hypothesis generation to physically plausible candidates.

Stage B.5 (signal evidence build) queries the plant historian for process variables associated with the equipment and its KG neighbors within the analysis time window. It applies anomaly extraction to produce a set of labeled signal intervals with estimated start and end times. The output is a `signal_evidence` artifact consumed by temporal reasoning in Stage E.

Stage C (candidate generation) produces an initial candidate set from the KG context. A candidate is a (failure mode, causal category) pair: failure modes are drawn from the plant FMEA library ingested into the KG, and each mode is assigned to the causal category it most plausibly occupies; a single failure mode can generate candidates in more than one category when it is plausible at multiple causal depths. Coverage against the 12-category causal taxonomy (Section 3.3) is enforced; every category must be populated with at least one candidate or explicitly ruled out with a documented rationale. Uncovered categories generate attention flags. When an LLM is used to assist candidate generation, its output is constrained to KG-valid failure modes and causal categories; all downstream scoring and ranking remain fully deterministic.

Stage D (evidence retrieval) executes per-candidate hybrid queries against the evidence store. For each candidate failure mode, it retrieves a ranked list of CRs, WOs, procedures, and OE records using combined semantic similarity and lexical matching. Retrieved snippets are classified as

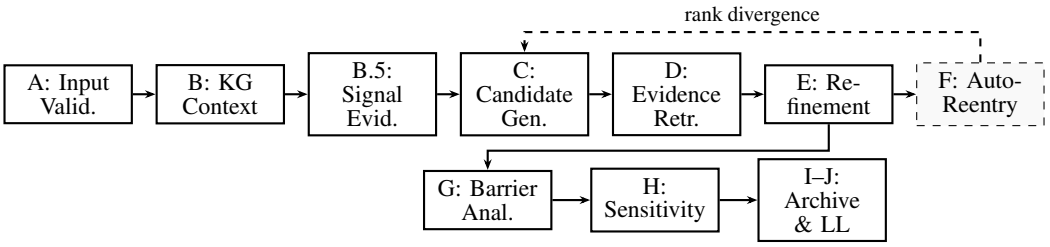

Figure 1. DACKAR 10-stage pipeline. Stages A–F form the primary analysis sequence: the KG drives Stages B and B.5 (context and signal evidence build); the causality engine drives Stage C (candidate generation) and Stage E (composite scoring); the evidence store drives Stage D (evidence retrieval); the orchestrator drives Stage F (auto-reentry). Stage F conditionally triggers a second pass of Stages C–E when rank divergence is detected (dashed feedback arc). Stages G–J complete the analysis under the Synthesis Module: barrier analysis (G), sensitivity analysis (H), archiving and lessons learned (I–J).

supporting, contradicting, or contextual. The output is an `evidence_bundle` indexed by candidate identifier.

Stage E (candidate refinement) computes composite scores by integrating the five reasoning dimensions for each candidate. Evidence-based re-weighting adjusts initial structural priors upward or downward based on the retrieved evidence bundle. Contradiction gating applies an attenuation multiplier to candidates with contradicting evidence. The output is a ranked list of refined candidates with per-dimension score breakdowns and a `scoring_evolution` artifact recording the pre- and post-refinement ranks.

Stage F (auto-reentry) is a conditional stage. If the rank-order divergence between the pre- and post-refinement candidate lists exceeds a site-configurable threshold, the orchestrator expands the KG traversal depth by one hop and reruns Stages C–E with the enlarged candidate set; the reentry terminates after at most one additional pass regardless of subsequent divergence. This prevents premature convergence when the initial KG scope was too narrow. Stage F was not triggered in any of the three validation examples.

Stages G.1–G.3 (barrier analysis and run manifest) apply the AP-913 equipment reliability framework to assess whether the failure involved degraded or bypassed defense-in-depth barriers. The `run_manifest` and `rca_card` artifacts are described in Section 4.3.

Stage H (sensitivity analysis) perturbs each dimension score by a fixed delta and records the resulting rank changes. This produces a `sensitivity_table` that quantifies which reasoning dimensions most influence the final ranking (critical information for the analyst who will review and potentially override the automated conclusion).

Stages I and J (archive and lessons learned) archive the current evidence bundle and scored candidates, enabling future runs to retrieve this event as an analog. Novel pattern detection compares the current failure signature against existing clusters; if no sufficiently similar prior event exists, the event is flagged for expedited lessons-learned processing. The `signal_lessons_learned` artifact documents any new signal-failure mode associations identified.

3.3. Causal Taxonomy

The 12-category taxonomy is derived from the multi-level causal accountability framework required by Institute of Nuclear Power Operations (INPO) AP-913 (Institute of Nuclear Power Operations (INPO), 2018) and reinforced by IAEA TECDOC-1756 guidance (IAEA, 2014): the three depth layers mirror the graded cause analysis approach mandated for significant equipment events, which requires identifying not only the proximate physical failure but also contributing human and organizational factors and the systemic root cause. The twelve categories operationalize this framework by enumerating the specific investigative dimensions an analyst must address or explicitly rule out for each event.

Stage C enforces coverage against this taxonomy spanning three depth layers:

1. *Proximate causes* (Categories A–F) include equipment degradation, support system loss, upstream influence, downstream influence, operating context, and external hazards.
2. *Contributing causes* (Categories G–K) include human execution error, design deficiency, configuration baseline deviation, inspection program gap, and supply chain defect.
3. *Root cause* layer (Category L) addresses systemic organizational or programmatic weakness.

Every category must be addressed or explicitly ruled out; coverage gaps produce structured attention flags that persist in the run manifest.

4. TEXT PROCESSING AND KNOWLEDGE EXTRACTION

DACKAR operates on four categories of input text: CR narratives (i.e., failure descriptions and observations), WOs (i.e., maintenance actions, findings, and technician notes), completed RCA documents (structured causal conclusions with

evidence citations), and fleet operating experience records from INPO databases. These documents are processed through two text analytics stages (named entity recognition and causal extraction) before entering the Chroma evidence store. The typed-artifact system that structures interstage data exchange is described in Section 4.3.

4.1. Named Entity Recognition

Nuclear maintenance text contains domain-specific entity types that general-purpose NER models trained on news or encyclopedic corpora do not recognize. DACKAR uses a hybrid NER pipeline (Wang et al., 2024) that combines a gazetteer-based lookup against a curated plant vocabulary, noun-phrase extraction over the dependency parse, and embedding-based retrieval to extract six entity classes from CR and WO text. Overlapping span candidates are resolved by priority order; a final compatibility check verifies that each extracted failure mode is physically consistent with the extracted equipment type according to the KG. Entities are normalized and linked to canonical KG identifiers.

The six entity classes are:

- Equipment identifiers: plant tag numbers and KKS-style identifiers, used as primary keys for KG node lookup in Stage B and equipment similarity filtering in Stage D.
- Failure modes: mechanism terms (e.g., "seat erosion," "packing leak," "bearing spall") drawn from the plant's failure mode vocabulary, used to seed Stage C hypothesis generation.
- Temporal markers: time expressions and duration phrases (e.g., "over the past 14 days," "47 days overdue") and lag values, used to validate failure mode latency windows in temporal reasoning.
- Physical measurements: process parameter values and units (e.g., "vacuum at 28.5 in-Hg," "dissolved oxygen at 12 ppb"), used to flag out-of-boundary anomalies.
- Alarm identifiers: control room alarm tags and setpoint references, linked to historian channel identifiers for the signal evidence build in Stage B.5.
- Location references: spatial and system-level references used to expand or constrain KG traversal scope.

Two additional features are computed alongside entity extraction. The *hedge fraction* (i.e., the proportion of modal auxiliary verbs (e.g., "may," "could," "possibly," "might") in the document relative to total verb count) provides a document-level uncertainty signal that modulates evidence confidence. The *lag extractor* identifies explicit duration claims associated with failure mode entities and uses them to validate the temporal plausibility of candidate anomaly onset windows against the FMEA specified latency windows.

4.2. Causal Extraction from CRs

Causal extraction identifies (cause, effect) pairs expressed in maintenance text. CRs might contain causal statements at multiple depths: direct mechanism statements at the proximate level ("the check valve failed due to seat erosion"), contributing condition statements ("high-frequency pressure cycling contributed to seat wear over the maintenance interval"), and organizational weakness statements at the root cause level ("the PM interval was insufficient to detect progressive degradation before functional failure").

Extraction proceeds in two steps. First, CAUSAL_TRIGGER entities mark candidate causal spans in the text. Second, for each trigger, a relation extraction model identifies the syntactic head of the cause argument (the entity or clause to the left of the trigger) and the effect argument (the entity or clause to the right). Both arguments are resolved to the nearest FAILURE_MODE, EQUIPMENT_TAG, or condition phrase in the containing sentence using the NER output.

The resulting (cause, effect) pairs serve three functions in the pipeline. They are matched against the 12-category causal taxonomy to classify which depth layer each extracted pair occupies, providing category coverage grounded in documentary evidence rather than relying on rule-based generation alone. They condition Stage C hypothesis generation; the LLM receives both the KG context from Stage B and the extracted causal pairs from the current CR as structured input, constraining generation to failure modes supported by both the plant model and the source text. They are stored in the evidence bundle for direct citation in the `rca_card` output, providing explicit textual provenance for each causal claim in the final report.

4.3. Pipeline Data Structures and Typed Artifacts

Each pipeline stage produces a named, schema-validated Python object (i.e., a typed artifact) consumed by downstream stages. This discipline provides three operational benefits: individual stages can be unit tested in isolation without end-to-end pipeline execution; analysts can inspect intermediate results at any stage boundary; and artifacts from prior runs can be stored and retrieved as evidence analogs for future events, accumulating institutional knowledge across operating cycles.

The pipeline's two primary analyst-facing outputs are the `rca_card`, which presents ranked causal hypotheses with evidence citations and corrective action recommendations organized at three causal depths (proximate, contributing, and root cause), and the `run_manifest`, which records the complete audit trail of available data, degraded inputs, quality multipliers, and open analyst review items. Taxonomy coverage is enforced at Stage C: its schema requires each of the 12 causal categories to be either populated with at

least one scored candidate or explicitly ruled out with a structured rationale; gaps propagate as attention flags to the final `rca_card`.

4.4. Event and Equipment Similarity

Two similarity engines extend the evidence scope beyond the immediate failing component: the event similarity engine, which retrieves historical precedents for the current failure, and the equipment similarity engine, which identifies sister components whose operating experience is relevant to fleet-level causal hypotheses.

The event similarity engine matches the current event against historical records using five structured criteria: asset type (component class as defined in the KG), component identifier (exact tag for same-component recurrence, relaxed to type for cross-unit matching), failure mode (KG failure mode ID or free-text description), event type (functional failure, degraded condition, near miss, or actuation), and operating conditions at the time of failure (mode, power level, environmental context). Retrieved events are assigned a tier confidence multiplier: plant records (same facility, same component class) carry a multiplier of 1.00; fleet records (same design, different facility) carry 0.80; and industry records from INPO, Nuclear Regulatory Commission, and Electric Power Research Institute databases carry 0.60. This weighting ensures that a direct plant recurrence is scored more heavily than a generically similar industry case.

For each match, the engine produces a recurrence characterization, whether this is a first occurrence, a known recurring pattern, or a recurrence after a previously applied corrective action. When a prior RCA report exists for the same failure mode and component, the engine surfaces the prior root cause finding and flags whether the corrective actions were closed and marked effective. Recurrence after a supposedly effective corrective action is a direct signal for Category L (systemic organizational weakness) scoring. An optional semantic recurrence mode performs embedding-based matching against stored event descriptions, enabling the engine to surface similar events that share the same underlying failure mechanism but differ in terminology or component nomenclature. This mode requires a populated semantic event archive and is disabled by default.

Some failure hypotheses (particularly design deficiency [Category H] and supply chain defect [Category K]) are fleet-level phenomena that cannot be adequately assessed from the single failing component's history alone. The equipment similarity engine identifies sister components (i.e., nominally identical equipment at other units or trains) using three matching criteria drawn from the KG component record: component type (equipment class label), service conditions (operating pressure range, fluid type, temperature class, safety classification), and manufacturer and model (vendor name and model designation where available). Matches are scored by the number of criteria satisfied, weighted by specificity; a manufacturer-and-model match is a closer sister than a type-only match. Sister component documents are added to the Stage D query plan and appear in the `evidence_bundle` alongside direct-component documents, tagged with their source tier. Categories H and K benefit most from this expansion; for proximate equipment-origin categories (A–F), the incremental value is limited.

4.5. PM Compliance Engine

The PM compliance engine assesses whether PM was current on the affected equipment at the time of the event. Its output feeds the governance scoring dimension and provides the primary evidence for the Category J (inspection and testing program inadequacy) assessment. Inputs are PM schedule records, CMMS maintenance history (completed WOs with execution dates and as-found and as-left condition fields), and condition monitoring trend data retrieved via the configured CMMS adapter.

The engine applies five sequential assessments:

1. *Schedule alignment:* identifies which PM tasks on the affected equipment and its KG neighbors were overdue at the time of the event and by how many days.
2. *Scope analysis:* maps each PM task to the failure modes it covers, identifying failure modes with no associated preventive task (each such gap is a potential Category J contributor).
3. *Effectiveness analysis:* computes the fraction of historical PM executions that recorded a degraded as-found condition, answering whether prior maintenance was detecting degradation before it progressed to failure.
4. *Frequency concern flagging:* compares the actual mean interval between PM executions against the scheduled frequency; significant exceedance is flagged as a potential programmatic deficiency.
5. *Risk roll-up:* combines gap presence, overdue status, and failure mode linkage into summary compliance and risk fields consumed by the composite scoring function.

The engine produces a structured compliance record that feeds two downstream consumers. The governance scoring dimension uses the compliance status and overdue extent to modulate the composite score for candidates in equipment-origin categories. The Category J assessment uses the scope gap findings directly: a failure mode with no covering PM task, combined with a confirmed degradation finding, constitutes the primary evidence pattern for an inspection program inadequacy conclusion. When CMMS data is incomplete (missing closure dates, absent as-found condition fields), the engine applies conservative fallback values and records the data gap in the run manifest as an open analyst review item.

The entity types extracted by NER (Section 4.1) and the causal pairs extracted from CRs (Section 4.2) feed directly into Stage C hypothesis generation and provide the evidence formula inputs consumed by the reasoning mechanisms described in the next section.

5. REASONING MECHANISMS

5.1. Structural Reasoning via KG

The KG encodes plant equipment as a directed property graph with node types for components, systems, failure modes, safety functions, and maintenance activities. Edges encode physical connectivity (upstream and downstream flow paths), failure mode association (component type to failure mode library), and safety function dependency (component to safety function). The graph is populated from the plant's design basis documentation and updated from CMMS records as component configurations change.

Stage B traverses the KG neighborhood of the reported equipment to a configurable depth (default: two hops), collecting all nodes reachable through physical connectivity and failure mode association edges. This traversal produces the `kg_context` artifact, which defines the bounded hypothesis space for Stage C. The practical effect is that only failure modes documented in the plant's design basis for components physically connected to the reported equipment can reach the scoring stages.

Without this structural filter, a language model generating failure mode hypotheses from parametric knowledge alone would produce candidates that are generically plausible for the component class but physically impossible given this specific plant's configuration (e.g., failure modes associated with components that were removed during a refueling outage or bypassed by a design modification). The KG enforces plant-specific physical plausibility as a precondition for hypothesis evaluation.

5.2. Temporal Reasoning via Allen Interval Algebra

A process signal anomaly is represented as a closed time interval $[s_i, e_i]$, where s_i is the estimated anomaly onset and e_i is the estimated anomaly resolution. The failure event is represented as the interval $[s_F, e_F]$ where s_F is the event initiation time and e_F is the time of restoration to normal operation.

Allen's interval algebra (Allen, 1983) defines 13 mutually exclusive and jointly exhaustive relations between any two intervals. DACKAR applies a reduced set of six relations with associated causal support scores, as shown in Table 1. The support score reflects the strength of the causal inference implied by each temporal relationship in the nuclear equipment failure context.

Table 1. Allen interval relations applied to signal anomaly intervals $[s_i, e_i]$ relative to failure event $[s_F, e_F]$, with causal support scores.

Relation	Score	Nuclear Interpretation
OVERLAPS	0.90	Anomaly onset precedes event; strong causal candidate
CONTAINS	0.85	Anomaly spans entire event window; persistent degradation
PRECEDES	0.75	Anomaly fully resolved before event; contributing factor
DURING	0.30	Anomaly occurs within event; co-symptom or consequence
FOLLOWS	0.00	Anomaly begins after event; excluded (hard gate)
EQUALS	0.85	Anomaly and event co-occur; highly correlated

The FOLLOWS relation implements a hard gate; any signal whose anomaly interval begins after the failure event resolution time e_F is automatically excluded from the candidate pool regardless of semantic similarity to any failure mode description. This prevents the systematic error of treating downstream process responses (which appear abnormal because the failure has already occurred) as causal contributors to that failure.

In practice, historian data from multiple plant systems is time-synchronized with a 30-minute simultaneity epsilon $\varepsilon = 30$ min. Two event times t_a and t_b are treated as simultaneous if $|t_a - t_b| \leq \varepsilon$, preventing false FOLLOWS classifications due to clock drift or logging latency across different data acquisition systems.

Before Allen scores are assigned, each signal window is classified by duration and severity into one of three tone bands. *Transient:* anomaly duration < 5 min AND severity < 0.85; receives a reduced scoring weight. *Trip-band persistent:* duration ≥ 2 min AND severity ≥ 0.85; classified as a safety system response to an initiating event and excluded from causal scoring. *Standard:* all other windows; Allen scoring applied without penalty. Both thresholds are site-configurable. Tone classification is necessary because fast-transient events can cause multiple plant systems to respond within seconds of the initiating signal; without tone filtering, correctly functioning protective logic would receive high Allen support scores as spurious causal precursors.

The base Allen score is further qualified by two additional checks. First, the signal pattern is compared against stored historical episode records for the same component type: if the expected pattern is systematically absent from historical events for that failure mode, the recurrence quality is flagged low and the analyst is prompted to verify whether the evidence corpus is representative. Second, the score is qualified against the FMEA latency window specified for the failure mode in the KG. The temporal score is highest when the observed lead time falls within the expected latency range for

that failure mechanism. A PRECEDES relation where the observed lag is 30 minutes, but the failure mode typically requires 12 hours of progressive degradation receives a penalized latency alignment score, even though the Allen relation itself is causal.

5.3. Documentary Evidence Reasoning

The evidence store contains CRs, WOs, PM procedures, completed RCAs, and fleet OE records from the INPO IRIS database. For each candidate failure mode, Stage D issues a hybrid query combining semantic and lexical retrieval, reranking results by a weighted combination of both scores.

Retrieved snippets are classified into three categories by a rule-based classifier that examines the snippet's modal language, negation patterns, and factual claims relative to the candidate description: *supporting* (the snippet reports a similar failure occurring under similar conditions), *contradicting* (the snippet explicitly reports that this failure mode was ruled out or did not occur under similar conditions), and *contextual* (relevant background information without direct causal bearing).

Evidence integration follows an *attenuation* model rather than a zeroing model. A contradicting snippet multiplies the candidate's evidence component score by an attenuation factor $\alpha \in (0, 1)$, preserving the candidate in the ranked list while reducing its score. The rationale is that a single contradicting analog does not logically preclude the failure mode (it only weakens the evidence case). The analyst receives both the attenuated score and the contradicting evidence for their review, rather than having the pipeline silently eliminate a plausible candidate.

Source-tier weighting assigns authority multipliers to evidence by provenance across six tiers: plant instance records (1.00), plant procedures (0.80), FMEA documents (0.70), fleet family records (0.50), INPO IRIS fleet OE (0.40), and NRC ADAMS industry references (0.30). This reflects the decreasing relevance of evidence as plant-specific configuration differences increase.

The NER pipeline described in Section 4.1 is applied to retrieved snippets at query time to extract PART_NUMBER and LOT_NUMBER entities. These are cross-referenced across the full evidence corpus to detect supply chain failure patterns (cases where multiple failures across a fleet trace to the same manufacturing batch or vendor specification deficiency).

The composite score S_k for candidate k integrates all five reasoning dimensions:

$$S_k = w_{\text{struct}} \cdot r_k^{\text{struct}} + w_{\text{temp}} \cdot r_k^{\text{temp}} + w_{\text{evid}} \cdot r_k^{\text{evid}} + w_{\text{gov}} \cdot r_k^{\text{gov}} + w_{\text{hist}} \cdot r_k^{\text{hist}} \quad (1)$$

where each $r_k^{(\cdot)} \in [0, 1]$ is the normalized score for dimension $(\cdot)$ and the weights $w_{(\cdot)}$ sum to unity. Critically, the weight vector is *category-specific*: equipment-origin candidates (categories A–F) use $w_{\text{struct}} = 0.30$, $w_{\text{temp}} = 0.20$, $w_{\text{evid}} = 0.20$, $w_{\text{gov}} = 0.10$, and $w_{\text{hist}} = 0.20$; human/organizational candidates (category G) shift to $w_{\text{evid}} = 0.65$, $w_{\text{gov}} = 0.15$, $w_{\text{struct}} = 0.05$, $w_{\text{temp}} = 0.10$, and $w_{\text{hist}} = 0.05$; inspection program candidates (category J) use $w_{\text{evid}} = 0.55$, $w_{\text{gov}} = 0.30$, $w_{\text{struct}} = 0.05$, $w_{\text{temp}} = 0.05$, and $w_{\text{hist}} = 0.05$. This reflects the different evidential basis appropriate to each causal depth layer: physical failure modes are most constrained by structure and timing, while human and organizational causes are almost entirely resolved through documentary evidence and governance records. The weight values reported here were elicited from domain experts; they are treated as provisional initial assignments and are intended for calibration against a labeled plant dataset in subsequent validation phases. Sensitivity to weight perturbations is quantified by the Stage H sensitivity mechanism (Section 3.2), which is implemented but not exercised in the validation examples reported here. The evidence dimension itself is updated at Stage E as:

$$r_k^{\text{evid}} = \text{clamp}\big(0.30\, E_{\text{prior}} + 0.55\, S_{\text{support}}\, w_{\text{auth}} + 0.15\, E_{\text{context}} - 0.45\, C_{\text{contradict}},\ [0, 1]\big) \quad (2)$$

where E_{prior} is the pre-retrieval evidence score carried uniformly across all candidates before Stage D (equal to 0.413 in Example 3), S_{support} is the aggregated supporting evidence score, w_{auth} is the authority weight of the source tier (plant instance: 1.00; plant procedure: 0.80; FMEA: 0.70; fleet family: 0.50; INPO IRIS: 0.40; ADAMS: 0.30), and $C_{\text{contradict}}$ is the contradiction signal. When historian signal-chain evidence is available, the final evidence score blends documentary and signal-chain components: $r_k^{\text{evid}} = 0.70\, E_{\text{doc}} + 0.30\, E_{\text{chain}}$.

5.4. Log/SOE Temporal Pattern Recognition

This engine extends the temporal reasoning dimension: where Allen interval algebra classifies individual signal anomaly timing, the log and sequence-of-event (SOE) pattern recognition characterizes the alarm sequence leading to the event as a whole, enabling detection of multi-signal precursor patterns. The engine accepts alarm log records and SOE records, segments them into temporally coherent episodes, and scores each historical episode against the current event's alarm signature using set overlap, sequence-order similarity, and temporal distribution similarity. The output is a ranked list of analogous past events that weights evidence retrieval in Stage D; a novelty flag is set when no prior match exceeds the similarity threshold. This mechanism is implemented but is not exercised in the validation examples; empirical evaluation is deferred to Phase 2.

5.5. Cross-Temporal Pattern Recognition

This engine contributes to both the historical and governance reasoning dimensions: by linking the current event to a preceding multi-event degradation sequence, it surfaces evidence for organizational and programmatic root cause categories (Categories G–K) that single-event analysis may not surface. The linkage algorithm identifies chains of historical alarm/SOE episodes (Section 5.4) where signature intensity increases monotonically toward the current failure, providing evidence that a slowly progressing degradation was allowed to continue uninterrupted. This capability is disabled by default, requires at least two years of historical signal episode data, and is most relevant for Category A latent degradation modes (bearing wear, corrosion trends, insulation resistance decline). Cross-temporal linkage is an implemented architectural capability not exercised in the validation examples; empirical benchmarking is deferred to Phase 2.

The three examples below are representative validation scenarios derived from actual nuclear plant failure mechanisms and anonymized for publication.

6. EXAMPLE 1: PRESSURIZED-WATER REACTOR (PWR) CONDENSER AIR IN-LEAKAGE

Event EVT-U2-2024-0847 records a 14 day multi-signal anomaly window preceding an automatic load runback on Unit 2. Process data shows degraded condenser vacuum, elevated hotwell dissolved oxygen, reduced turbine efficiency, and condenser pressure trending upward over the observation window. The analysis window contains five competing failure mode hypotheses generated at Stage C: air in-leakage through condenser tube sheet joints (FM-AIR-INLEAK), condenser tube fouling (FM-TUBE-FOULING), steam jet air ejector degradation, hotwell pump degradation, and feed system restriction.

Pre-refinement scoring at the end of Stage C assigned FM-TUBE-FOULING a marginally higher structural prior based on KG edge weights encoding the historical base rate. Evidence retrieval at Stage D reversed this ranking; the evidence bundle for FM-AIR-INLEAK contained three supporting CRs documenting identical dissolved oxygen signatures following condenser tube sheet joint rework campaigns, while the FM-TUBE-FOULING bundle contained a contradicting snippet from a completed RCA on a similar unit that ruled out fouling as the primary mechanism when the dissolved oxygen trend preceded the vacuum degradation trend by more than 48 hours.

PM governance scoring at Stage E applied a positive delta to FM-AIR-INLEAK; the associated condenser leak-off test was 47 days overdue against a 90 day PM interval, yielding an overdue fraction of 0.52. This delta, combined with the evidence-based reweighting, produced a final composite score of 0.81 for FM-AIR-INLEAK versus 0.64 for FM-TUBE-FOULING, a margin sufficient to issue a single primary conclusion without triggering the near-tie flag.

A counterfactual run with PM governance scoring disabled reduced the FM-AIR-INLEAK score to 0.72 and the FM-TUBE-FOULING score to 0.68: a margin of 0.04 that would have triggered analyst tie-break review. This demonstrates that the PM governance delta is not merely a minor scoring adjustment but a ranking-determinative mechanism in this example, when evidence retrieval produces a near-tie.

7. EXAMPLE 2: PWR REACTOR TRIP

Event E2026-03-10-001 records a reactor trip with multiple concurrent alarms. Three process signals show anomalies that either preceded or followed the trip initiation time t_F. Two signals (rod position deviation and nuclear power rate) have anomaly intervals that OVERLAP with $[s_F, e_F]$ and receive Allen support scores of 0.90. One signal (CRD position anomaly) has an anomaly interval whose onset is $T+10$s after the trip initiator and therefore receives the FOLLOWS classification with a support score of 0.00, eliminating hypothesis `FM-CRD-MECHANICAL` from the candidate pool with `reason_code="timeline_contradiction"`.

Without temporal gating, the CRD position anomaly would have been scored by semantic similarity alone as a candidate contributing cause, since CRD position deviations are semantically related to reactor trip failure mode descriptions in the evidence corpus. The Allen hard gate excludes it before it enters the scoring pipeline. The run manifest records this exclusion with a structured attention flag documenting the temporal basis: anomaly onset $t_s = t_F + 10$s, relation = FOLLOWS, and action = excluded.

To illustrate what temporal gating addresses, in a post-trip transient, multiple plant systems respond to the trip itself, producing anomaly signatures in historian data that appear causally related to the trip when examined by semantic similarity alone. The Allen interval classification resolves the ambiguity using only timing information, independent of any semantic content.

8. EXAMPLE 3: PWR SERVICE WATER CHECK VALVE LEAKTHROUGH

Event EVT-U1B-2025-0312 records the leakthrough of a service water check valve identified during surveillance testing. This scenario exercises the broadest set of pipeline mechanisms, involving five causal categories across three depth layers and requiring fleet OE retrieval, NER-based cross-referencing, and near-tie detection.

Stage C generated 11 candidates across all 12 taxonomy categories; one category (external hazards) was explicitly ruled out based on KG context indicating the valve was located

in a seismically isolated indoor location, and this ruling was documented as a structured attention flag. At this stage, no documentary evidence has been retrieved; the evidence dimension carries its uniform prior value (0.413) for all candidates. The pre-refinement leader is CHK-SEAT-EROSION (Category A, composite 0.656), driven by a stronger temporal score (0.546 vs. 0.433 for PM-FREQ-NONCONF) reflecting better precursor signal alignment with seat erosion kinematics. PM-FREQ-NONCONF (Category J, composite 0.595) trails at this stage despite identical structural and governance scores, because its temporal evidence is weaker — PM noncompliance does not produce a distinctive precursor signal pattern.

Stage D retrieved seven evidence snippets from seven source documents, including fleet OE analogs from INPO IRIS (source-tier authority weight 0.40; retrieval similarity score 0.85) documenting check valve seat erosion associated with high-frequency pressure cycling. Two WO records directly supporting PM schedule noncompliance raised PM-FREQ-NONCONF's evidence score from the prior 0.413 to 0.561; fleet OE support for CHK-SEAT-EROSION raised its evidence score to 0.493. NER extraction from retrieved snippets identified a common vendor part number across three records; cross-referencing against the current event's WO revealed a matching lot number, flagging a potential supply chain contributing cause.

The rank inversion at Stage E follows directly from the category-specific weight profiles: Category J (surveillance) assigns $w_{\text{evid}} = 0.55$, compared to $w_{\text{evid}} = 0.20$ for Category A (equipment origin). The larger evidence gain for PM-FREQ-NONCONF, amplified by this higher weight, is sufficient to overcome CHK-SEAT-EROSION's temporal advantage. Post-refinement scores converge to a near-tie: PM-FREQ-NONCONF at 0.456 and CHK-SEAT-EROSION at 0.453, a gap of 0.003. The near-tie flag fires when the rank gap is strictly less than 0.01; the 0.003 gap satisfied this condition and the pipeline refused to issue a single primary conclusion. The run manifest records an open item requiring analyst tie-break review.

A separate quality multiplier of 0.682 was applied to the FMEA-dependent scoring components because failure mode and effects analysis linkage data for this valve model were missing from the KG. This degradation was flagged in the run manifest as an attention item; the 0.682 multiplier reflects the reduced confidence in structural priors when FMEA dates are unavailable.

Table 2 summarizes the final scored candidates.

Table 2. Top-ranked candidates for event EVT-U1B-2025-0312. Near-tie flag triggered when rank gap < 0.01.

Rank	Candidate ID	Score	Category
1	PM-FREQ-NONCONF	0.4560	Inspection Gap (J)
2	CHK-SEAT-EROSION	0.4530	Equip. Degradation (A)
3	SUPPLY-LOT-DEFECT	0.3810	Supply Chain (K)
4	DESIGN-SPEC-GAP	0.3250	Design Deficiency (H)
5	OPCTX-PRESSURE	0.2940	Operating Context (E)
Near-tie flag: TRUE	Open items: 2		

9. MECHANISM VALIDATION SUMMARY

The evaluation is scoped as mechanism validation: each entry in Table 3 confirms that a specific reasoning behavior operates as designed under the scenario conditions, using counterfactual runs where applicable to demonstrate that the mechanism is ranking-determinative rather than merely present. The ground truth in all three scenarios is defined by the authors based on documented failure physics, not recovered from completed plant RCA records; this is a necessary limitation of Phase 1 validation. This is distinct from performance benchmarking, which would require a labeled dataset of completed plant RCAs and is identified as the primary Phase 2 objective.

Table 3. Mechanism validation coverage across three scenarios. CF = counterfactual run included.

Mechanism	Ex.	CF
Evidence retrieval rank inversion	1	–
PM governance as ranking-determinative	1	Yes
Allen temporal hard gate (FOLLOWS exclusion)	2	–
Fleet OE retrieval with source-tier weighting	3	–
NER lot-number cross-reference	3	–
Contradiction attenuation (not zeroing)	3	–
Near-tie detection	3	–
12-category causal taxonomy coverage	3	–
Determinism (5 repeated runs)	All	–

Table 3 summarizes the mechanisms validated and whether a counterfactual run was included to demonstrate the mechanism's effect on the final ranking.

10. GRACEFUL DEGRADATION

Nuclear plant data environments are operationally heterogeneous. CMMS integrations may be unavailable during planned outages, historian data may have coverage gaps, OE databases may not be accessible from a given network segment, and FMEA documentation may be incomplete for older component types. DACKAR is designed to produce useful output under each of these degraded-data configurations, with attention flags in the run manifest that tell the analyst exactly what is missing and how it affected the scoring.

Table 4 summarizes the degradation behavior for each data source. In all cases the pipeline completes and produces a scored ranked list; the quality of that list degrades gracefully with the available data, and the degradation is explicitly quantified in the `rca_card` quality multipliers.

The quality multiplier mechanism is central to graceful degra-

Table 4. Graceful degradation behavior when data sources are unavailable.

Unavailable Source	Affected Stage(s)	Fallback Behavior
CMMS / PM records	E (governance)	Static PM priors; attention flag
FMEA linkage	C, E (structural)	Neutral structural prior (0.5); quality multiplier
Fleet / industry OE	D (retrieval)	Internal records only; tier weights adjusted
Alarm / SOE logs	B.5 (signal)	Signal reasoning skipped; temporal scores absent
Plant historian	B.5 (signal)	Signal reasoning skipped; attention flag

dation. Rather than returning an error or refusing to score when data is missing, the pipeline computes a multiplier in $(0, 1]$, reflecting the fraction of expected evidence that was actually available, and applies it to the affected scoring components. The analyst sees both the degraded score and the multiplier, enabling a calibrated interpretation of the pipeline's confidence.

11. DISCUSSION

The five reasoning dimensions implemented in DACKAR are not specific to nuclear power. Any asset-intensive industry with structured equipment topology data, process instrumentation historian data, and documentary maintenance records can support the same architecture.

Structural reasoning requires a formalized equipment hierarchy: present in manufacturing as bill-of-materials and machine-cell connectivity graphs, in offshore energy as process and instrumentation diagrams, and in petrochemical plants as piping and instrumentation diagrams with equipment databases. Temporal reasoning requires timestamped process signals; historian infrastructure is standard in all of these sectors. Evidence-based reasoning requires a corpus of maintenance and operational records with sufficient volume for retrieval to be meaningful; large plants in these sectors accumulate tens of thousands of WOs and inspection reports per year. Governance reasoning requires structured PM schedule records and work order histories linked to equipment failure mode libraries; reliability-centered maintenance programs in manufacturing and energy provide an equivalent foundation. Historical reasoning requires a corpus of past failure events with documented outcomes; any organization maintaining a corrective action or maintenance management system accumulates the necessary records over time.

The causal taxonomy enforced in Stage C is the most domain-specific component of the architecture. Replacing the 12-category nuclear taxonomy with an industry-appropriate taxonomy (e.g., a reliability-centered maintenance failure class structure for manufacturing) is a configuration change that does not require modifying the pipeline's scoring logic. Domain adaptation is therefore expected to be primarily a knowledge engineering task (populating the KG with domain-specific equipment failure mode relationships and configuring the taxonomy) rather than a machine learning retraining task.

12. CONCLUSION

DACKAR is a 10-stage deterministic pipeline for the RCA of equipment failures. It automates hypothesis bounding through KG neighborhood traversal, temporal consistency enforcement through Allen interval algebra, and evidence retrieval and scoring through a hybrid semantic-lexical evidence store with source-tier weighting. The composite scoring function integrates five reasoning dimensions; scoring and candidate ranking are fully deterministic and independent of language model availability.

Three representative validation scenarios demonstrated the following mechanisms in isolation: PM governance scoring as a ranking-determinative factor in Example 1 (condenser air in-leakage, with counterfactual confirmation); Allen temporal hard gating excluding a post-event signal from the candidate pool as designed (reactor trip); and fleet OE retrieval with NER-based lot-number cross-referencing, contradiction attenuation, near-tie detection triggering analyst review, and complete 12-category causal taxonomy coverage enforcement (service-water check valve leakthrough). Determinism was confirmed across five repeated runs.

Three implemented mechanisms (i.e., auto-reentry (Stage F), sensitivity analysis (Stage H), and cross-temporal pattern recognition) were not exercised in the validation examples and are reserved for Phase 2 evaluation together with a performance benchmark against completed historical plant RCA records. The primary next steps are an evaluation against completed historical RCAs from actual plant CAPs, live CMMS integration in an operational network environment, and calibration of the composite score weights against a labeled plant-specific dataset. The generalizability of the architecture to manufacturing, offshore energy, and petrochemical diagnostics will be explored through an application to asset-specific KG configurations using the same pipeline implementation.

ACKNOWLEDGMENT

This work was supported by the U.S. Department of Energy, Office of Nuclear Energy, under the Light Water Reactor Sustainability (LWRS) program. AI tools assisted with the development of the presented framework and the revision of this paper. All technical analyses, designs, results, and conclusions were developed and verified by the authors.

REFERENCES

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, *26*(11), 832–843.

Cai, B., Huang, L., & Xie, M. (2017). Bayesian networks in fault diagnosis. *IEEE Transactions on Industrial Informatics*, *13*(5), 2227–2240.

Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutierrez, C., ... Zimmermann, A. (2021). Knowledge graphs. *ACM Computing Surveys*, *54*(4), 1–37. doi: 10.1145/3447772

Hu, D., Zhang, C., Yang, T., & Chen, G. (2020). Anomaly detection of power plant equipment using long short-term memory based autoencoder neural network. *Sensors*, *20*(21), 6164. doi: 10.3390/s20216164

IAEA. (2014). *Root cause analysis following an event at a nuclear installation: Reference manual* (Tech. Rep. No. TECDOC-1756). International Atomic Energy Agency.

IAEA. (2025). *Considerations for deploying artificial intelligence applications in the nuclear power industry* (Tech. Rep. No. NR-T-1.26). Vienna, Austria: International Atomic Energy Agency.

Institute of Nuclear Power Operations (INPO). (2018). *AP-913: Equipment reliability process description* (Revision 6 ed.; Tech. Rep. No. AP-913). Atlanta, GA: INPO.

Kim, J., Kim, J., Lee, A., Kim, J., & Diesner, J. (2024). LERCause: Deep learning approaches for causal sentence identification from nuclear safety reports. *PLOS One*, *19*(8), e0308155. doi: 10.1371/journal.pone.0308155

Latino, R. J., & Latino, K. C. (2006). *Root cause analysis: Improving performance for bottom-line results*. CRC Press.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th international conference on neural information processing systems* (pp. 9459–9474).

Lin, L., Zhang, S., Fu, S., & Liu, Y. (2025). FD-LLM: Large language model for fault diagnosis of complex equipment. *Advanced Engineering Informatics*, *65*, 103208. doi: 10.1016/j.aei.2025.103208

Rooney, J. J., & Vanden Heuvel, L. N. (2004). Root cause analysis for beginners. *Quality Progress*, *37*(7), 45–56.

U.S. Nuclear Regulatory Commission (NRC), & Idaho National Laboratory (INL). (2022). *Exploring advanced computational tools and techniques with artificial intelligence and machine learning in operating nuclear plants* (Tech. Rep. No. NUREG/CR-7294). Washington, DC: U.S. Nuclear Regulatory Commission.

Wang, C., Mandelli, D., & Cogliati, J. J. (2024). Technical language processing of nuclear power plants equipment reliability data. *Energies*, *17*(7), 1785. doi: 10.3390/en17071785

Weber, P., Medina-Oliva, G., Simon, C., & Iung, B. (2012). Overview on Bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, *25*, 671–682.

BIOGRAPHIES

Diego Mandelli is an R&D Scientist in the "Plant Optimization" Department at Idaho National Laboratory (INL). His areas of expertise include risk, reliability, and system health management, and his research focuses on developing probabilistic methods based on machine learning, data mining, and optimization algorithms. He is currently employing these methods to perform a state-of-the-art simulation-based safety assessment (also known as dynamic probabilistic risk assessment), system health management, and stochastic optimization. His developed methods include data pre-processing, system modeling, system analysis, data mining and visualization, and decision-making. He holds a Ph.D. degree in nuclear engineering from The Ohio State University (2011).

Congjian Wang is a computational nuclear scientist at INL. He received his B.E. degree in engineering physics and his Ph.D. degree in nuclear engineering and carried out postdoctoral research focused on machine learning, advanced sensitivity and uncertainty quantification, verification, validation, and data assimilation. He has more than 10 years of experience in developing and implementing artificial intelligence algorithms which include but are not limited to physics-based reduced-order modeling, surrogate modeling, data mining techniques, and deep neuron networks. He is the main developer of the Risk Analysis Virtual ENvironment (RAVEN) framework and is mainly responsible for developing and implementing supervised and unsupervised learning algorithms, cross validations, and verification and validation metrics within it.