

EVIDENCE-GROUNDED DECISION SUPPORT FOR AIRCRAFT LINE MAINTENANCE USING CONFORMAL PREDICTION AND RETRIEVAL- AUGMENTED NLP FROM TECHNICAL LOG RECORDS

Arthur Dela Peña^{*}  0000-0003-2519-2128

Jefferson Clariza¹  0000-0001-9487-5606

Mary Ann Aballiar-Vista²  0000-0003-0794-5237

¹ Aviation Maintenance Technology, National Aviation Academy of the Philippines,
Basa Air Base Campus, Pampanga, Philippines

² Aviation Information Technology, National Aviation Academy of the Philippines,
Pasay City, Philippines

ABSTRACT

Aircraft line maintenance requires rapid triage of short defect narratives despite variable documentation quality, which can weaken coding consistency and traceability. This study develops and evaluates an evidence-grounded decision-support pipeline using secondary records from the Federal Aviation Administration Service Difficulty Reporting System. The corpus comprised 4,430 deduplicated events from 2021 to 2023 across 10 receiving regions and 290 aircraft make-model combinations, with a heavily imbalanced distribution of Joint Aircraft System/Component (JASC) codes as supervised labels; broader Air Transport Association chapter groupings were used only for post hoc aggregation. Validation used a chronological 70/10/20 train/calibration/test split, with calibration reserved for conformal tuning, plus a station hold-out protocol to assess cross-site transferability. The pipeline combined a lightweight text classifier for JASC triage, term frequency-inverse document frequency (TF-IDF) retrieval to identify label-consistent precedents, and split-conformal prediction with a review-required abstention rule. On the temporal test set, the TF-IDF representation with a linear support vector machine classifier achieved top-1 accuracy of 0.509 and top-3 accuracy of 0.683, while retrieval achieved Recall@10 = 0.688 and mean reciprocal rank = 0.466. Conformal prediction attained empirical coverage of 0.919 at a 90% target. Performance declined directionally under station shift, highlighting the need for monitoring, governance, and human oversight in deployment. Overall, the study contributes a deployment-aligned evaluation protocol integrating retrieval and conformal prediction for auditable line-maintenance triage.

Keywords

aircraft maintenance diagnostics; line maintenance triage; JASC code classification; evidence-grounded retrieval; conformal prediction; abstention.

* Corresponding Author. Email address: artair248@gmail.com
Received 23.01.2026 • Revised 14.03.2026 • Accepted 16.03.2026
© 2026 Author[s]. Article category: research article



1. INTRODUCTION

Aircraft line maintenance is a safety-critical, time-pressured decision environment in which technicians must interpret brief defect narratives, select troubleshooting actions, and restore dispatch readiness under strict regulatory constraints and tight turnaround windows. Modern operators are simultaneously pushing for higher fleet availability and more data-driven maintenance strategies, which increases the value of fast, consistent, and auditable decisions at the ramp and gate [1, 2]. In this setting, uncertainty is not a side issue – it is the default condition: symptoms can be intermittent, aircraft may be operating away from base engineering support, and fault manifestation can be context-dependent under schedule pressure [3].

A major, ubiquitous – but still underutilized – asset for line maintenance decision-making is the aircraft technical log. Technical logs contain write-ups, rectification notes, and operational context, but they are primarily unstructured text with abbreviations, nonstandard phrasing, and variable documentation practices across stations and shifts. This “high-value, low-structure” profile is widely recognized in industrial predictive maintenance as a barrier to systematic reuse of experience at scale [4, 5]. The operational consequences are well-known to line teams: (i) repeat defects that are closed quickly yet recur due to intermittency or incomplete troubleshooting, (ii) inconsistent troubleshooting trajectories across sites and shifts because relevant prior cases are hard to find and compare, and (iii) large time-to-close variability for similar defects, which creates manpower uncertainty and delay exposure [3].

Recent research shows that natural language processing (NLP) and machine learning can extract signals from maintenance logs and support predictive maintenance and reliability objectives [1, 2]. For example, log-based learning has been shown to support rare-event failure prediction using operational maintenance-record datasets, showing that text-rich fields can carry early indicators even when failures are sparse [6]. At the same time, decision-support value depends on documentation quality, robustness, and operational alignment – not only aggregate accuracy metrics [3].

Despite this progress, a decision-ready gap remains for line maintenance. First, much log-NLP work still concentrates on coding/classification (e.g., mapping narratives to categories) without strong decision traceability – yet safety-adjacent users typically need inspectable reasoning and evidence, not just labels [7]. Second, deployed ML systems remain vulnerable to operational drift and dataset shift, which can induce overconfident errors when conditions change across stations, seasons, or documentation styles [8]. Third, while retrieval-augmented generation (RAG) is increasingly used to ground model outputs in enterprise knowledge, auditable retrieval pipelines for maintenance decision support have only recently emerged in the peer-reviewed engineering and industrial informatics literature [9, 10]. Finally, although conformal prediction has matured into a practical framework for distribution-free

uncertainty quantification with formal coverage guarantees, it is not yet routinely integrated into maintenance NLP pipelines as an integral safety layer that supports abstention (via set-valued outputs or reject options) when the model should not be trusted [11–14].

This study develop and empirically evaluate an evidence-based decision-support pipeline for aircraft line maintenance using only technical log text. The objectives are to: (1) build an NLP triage model for defect categorization relevant to line maintenance; (2) implement retrieval-augmented NLP to surface analogous historical cases as auditable evidence; (3) apply conformal prediction to provide calibrated uncertainty and abstention via prediction sets/reject options; and (4) evaluate predictive performance, calibration, and robustness under realistic dataset shift [8, 9, 11]. RQ1 asks how accurately technical-log text predicts defect categories relevant to line maintenance. RQ2 tests whether retrieval augmentation improves decision support, measured by retrieval quality (top- K case relevance) and downstream robustness. RQ3 evaluates whether conformal prediction preserves valid coverage under realistic dataset shift. Accordingly, H1 posits that retrieval augmentation improves top- K relevance and robustness versus NLP-only baselines by grounding outputs in analogous historical cases [9, 10]. H2 posits that conformal prediction achieves target coverage while reducing unsafe overconfident errors through calibrated uncertainty and abstention [11–14].

This work makes three contributions: (i) a line-maintenance-specific, evidence-grounded pipeline integrating NLP triage with auditable retrieval from technical logs; (ii) a conformal uncertainty layer enabling abstention and safety-aligned decision support; and (iii) a robustness evaluation protocol explicitly targeting station/season shift – aligning model assessment with real operational drift risks in deployed systems [8].

2. LITERATURE REVIEW

2.1 Aircraft maintenance text analytics

Text analytics is increasingly applied to unstructured maintenance narratives, making it well-suited to aircraft technical logs, which typically contain dense abbreviations, shorthand, and inconsistent phrasing. Taxonomy-guided methods can infer hierarchical equipment knowledge, including subassemblies and failure mechanisms, directly from free-text maintenance records [15]. In aviation safety corpora, topic modeling, particularly non-negative matrix factorization (NMF), has achieved high topic coherence, demonstrating its value for large-scale theme [16]. More broadly, maintenance-text and technical-document NLP studies show that useful structure can be extracted from unstructured records, while also highlighting persistent challenges such as label scarcity, vocabulary inconsistency, and heterogeneous writing styles across sources and settings [17–19]. Scoping reviews

confirm that topic models such as latent Dirichlet allocation (LDA) remain common tools for uncovering latent structure in technical [17]. NLP pipelines have been validated for extracting structured relations and metadata from technical documents, suggesting transferability to defect coding, clustering, and similarity-based triage of maintenance logs [18, 19]. Adjacent aviation-maintenance research shows that the field is moving toward AI-enabled maintenance optimization, remaining-useful-life prediction from operational data, and operator-oriented maintenance decision support [20, 21]. However, most of this newer work remains centered on sensor-rich engine-health and prognostics applications rather than on short, noisy narrative defect logs, leaving limited empirical evidence on whether text-analytic pipelines can robustly support line-maintenance tasks such as ATA/subsystem triage and repeat-defect screening across stations and documentation styles.

2.2 Retrieval for decision support in maintenance

Retrieval-based decision support mirrors how maintainers work, comparing current symptoms with prior cases and known fixes. Case-based reasoning (CBR) has shown high retrieval similarity and practical utility in maintenance-adjacent scheduling and incident retrieval, demonstrating the value of structured case libraries and similarity metrics for time-critical decisions [22, 23]. Studies of maintenance decision-making argue that data-driven tools should explicitly connect operational history to current actions, positioning retrieval as a core component rather than a secondary feature [24]. More recently, retrieval-augmented generation (RAG) has been proposed as an evidence-grounding mechanism that links model outputs to retrieved knowledge, with graph-based variants using entity relations to improve retrieval quality and controllability in knowledge-intensive settings [25, 26]. Structured retrieval frameworks that encode relationships and temporal constraints closely resemble maintenance histories, in which sequence and context matter [27]. Related aviation-maintenance research reinforces the value of similarity-based use of historical operational data. For example, unsupervised engine-health frameworks group similar degradation states and use prior patterns to estimate remaining useful life, while propulsion-health prognostics rely on trend analysis of historical performance signatures to anticipate deterioration and maintenance needs [28, 29]. However, peer-reviewed applications of RAG/GraphRAG to line maintenance technical logs remain scarce, particularly studies that jointly evaluate retrieval relevance, traceability, and robustness across stations and fleets.

2.3 Uncertainty quantification for safety-critical ML

Uncertainty quantification (UQ) is central to safety-critical and safety-adjacent ML because it enables calibrated confidence estimates and selective prediction, allowing models to abstain when uncertainty is high rather than producing overconfident errors [30]. Reviews show that explicit UQ, together with reject/deferral

options, can substantially improve the trustworthiness of deployed models in complex operational settings [31]. Empirical studies demonstrate that uncertainty estimates help identify rare but clinically or operationally important misclassifications and that selective classification can improve reliability by trading coverage for correctness when appropriate [32]. Accountability frameworks further argue that transparency, interpretability, and explicit UQ are prerequisites for safe decision support instead of opaque automation [33]. Adjacent engineering domains already embed UQ into safety cases, for example, in probabilistic safety assessment for automated driving and uncertainty-aware diagnostics for particle accelerators [34, 35].

Aviation maintenance has not yet operationalized UQ as a deployment control – with explicit abstention thresholds, coverage targets, or calibration-drift monitoring – or shown how these signals feed into line-maintenance workflows such as triage, escalation, and engineering review.

2.4 Dataset shift and robustness in operational ML

Dataset shift – arising from temporal drift, site differences, and evolving workflows – can substantially degrade model accuracy, discrimination, and calibration in operational deployments. Systematic reviews show that temporal shift and concept drift are common, with monitoring and statistical tests widely used for detection. However, no single mitigation strategy, such as refitting, feature updates, or recalibration, performs reliably across all settings [36, 37]. Empirical and longitudinal multi-site studies further show that performance can vary across institutions and over time, and that even when aggregate metrics appear stable, calibration and error profiles may drift in ways that matter for risk-sensitive decision support [36, 38]. Tools that characterize temporal variability and detect shifts in electronic records support more realistic evaluation protocols, whereas research on hidden subgroups shows that unrecognized strata can inflate apparent performance, motivating explicit subgroup and robustness checks [39, 40].

Aviation maintenance decision-support studies rarely adopt station-, season-, or fleet-based holdout protocols, report calibration under shift, define retraining or recalibration triggers, or examine how distributional changes affect retrieval quality in evidence-grounded systems.

3. MATERIALS AND METHODS

3.1 Study design

This study adopted a retrospective observational design using secondary operational records from aircraft line maintenance (i.e., historical technical log entries and related system timestamps/fields generated during routine operations). Because the study analyzed pre-existing records rather than intervening in maintenance activity, it was non-interventional and focused on quantifying patterns

and model performance using operational data [41]. Reporting followed established transparency expectations for observational studies (e.g., clear specification of the data source, eligibility criteria, variables, preprocessing, and the analysis plan) to support auditability and reproducibility [42]. Because the dataset consisted of secondary-use data, the design also emphasized explicit documentation of provenance, governance, and de-identification controls appropriate for reused operational records.

3.2 Data sources and scope

Operational data were obtained from the FAA Service Difficulty Reporting System (SDRS) and treated as secondary maintenance records suitable for retrospective analytics [2]. The observation window spanned 1 January 2021 to 22 December 2023 and, across year-based exports, yielded 4,929 raw records, of which 4,430 unique SDRs were retained after cleaning and deduplication (Table 1A). The retained corpus encompasses 2,204 aircraft and 97 operators, spanning 10 SDRS receiving regions and 290 distinct make-model combinations (Tables 1A–1C). Each record was keyed by the Operator Control Number (deduplication/audit trail), with fleet composition captured via Aircraft Make/Model and operational context proxied by Receiving Region Code. Because SDRS web queries are subject to maximum-record limits, the final dataset reflects the realized outputs of these constrained, year-based exports (Tables 1A–1B).

Eligibility screening was applied to improve analytical consistency and to reduce the noise typical of free-text maintenance datasets. Included records required (i) a valid Difficulty Date and (ii) a defect narrative in the Discrepancy field sufficient for NLP processing; system labels (JASC/ATA codes) were retained for supervised evaluation. Exclusions removed duplicate SDR identifiers, records missing critical fields required for modeling or evaluation, and clearly non-maintenance or administrative entries when identifiable, consistent with prior log-based maintenance analytics practice [6]. Field and label availability after cleaning are summarized in Table 1D.

Overall dataset scale, ATA/JASC label imbalance, and the composition of the train, calibration, and test partitions used for model development and conformal calibration are consolidated in Table 2. This corpus exhibits a skewed ATA/JASC distribution (few high-frequency classes, many low-frequency classes), and the temporal and station-based evaluations preserve this structure while documenting station and fleet coverage for robustness testing (Sections 3.3 and 3.10).

Table 1A. Overall dataset summary (SDRS exports, 2021–2023).

Item	Value
Study period (Difficulty Date)	01 Jan 2021 – 22 Dec 2023
Records extracted (raw rows across files)	4,929
Records included after cleaning (unique SDRs with required fields)	4,430
Excluded (duplicates / missing critical fields / missing narrative)	499
Unique SDR identifiers (Operator Control Number)	4,430
Unique aircraft (RegistryNumber)	2,204
Unique operators (OperatorDesignator)	97
Location/station proxy available	Receiving Region Code
Unique region codes (Receiving Region Code)	10
Narrative availability (Discrepancy present)	4,430 / 4,430 (100%)
Label availability (JASC Code present)	4,430 / 4,430 (100%)

Table 1B. Included records by year.

Year	Included records	Unique aircraft	Unique make-model	Region codes
2021	1,500	811	185	9
2022	1,500	831	164	9
2023	1,430	848	176	8
Total	4,430	2,204	290	10

Table 1C. Fleet composition (top 10 make-model by count).

Rank	Fleet (Aircraft Make + Aircraft Model)	Count
1	BOEING 737H4	226
2	AIRBUS A320232	209
3	DOUG MD11F	166
4	EMB ERJ170200LR	151
5	AIRBUS A321231	134
6	CNDAIR CL6002D24	128
7	CNDAIR CL6002C10	124
8	BOEING 737823	120
9	BOEING 737890	109
10	BOEING 737	106

Table 1D. Field/label availability (after cleaning).

Field	SDRS column	Availability in the included set
Unique control # (dedup/audit trail)	OperatorControlNumber	100%
Event/occurrence date	DifficultyDate	100%
Report/submission date	SubmissionDate	100%
Aircraft ID	RegistryNumber	High (used for unique count)
Fleet composition	AircraftMake, AircraftModel	High
Location/station proxy	ReceivingRegionCode	High
Defect narrative (NLP input)	Discrepancy	100%
System label (JASC code)	JASCCode	100%

Table 2. Dataset scale, JASC label imbalance, and split composition (SDRS, 2021–2023).

Quantity	All data	Train	Calibration	Test
Study period (Difficulty Date)	01 Jan 2021 – 22 Dec 2023	01 Jan 2021 – 14 Jan 2023	15 Jan 2023 – 24 Apr 2023	24 Apr 2023 – 22 Dec 2023
Events (unique SDRs)	4,430	3,101 (70.0%)	443 (10.0%)	886 (20.0%)
Unique aircraft (Registry Number)	2,204	1,605	263	552
Unique operators (Operator Designator)	97	90	34	41
Unique fleets (make–model) (both fields present)	290	245	79	143
Stations (Receiving Region Code)	10	10	8	8
Distinct JASC classes (JASC Code)	306	279	95	160
Majority class count	582	421	53	108
Majority class share (%)	13.14%	13.58%	11.96%	12.19%
Top-10 classes cumulative share (%)	43.68%	43.28%	53.72%	44.58%
Minority class count (min frequency)	1	1	1	1
Majority/minority ratio (max/min)	582:01:00	421:01:00	53:01:00	108:01:00

3.3 Data fields and schema

The SDRS-derived dataset was organized into a standardized event-level schema to support consistent NLP preprocessing, record linkage, and supervised learning from operational maintenance narratives [2, 6]. Each SDR entry was treated as a single maintenance event keyed by the Operator Control Number, which served

as the primary deduplication and audit-trail identifier. Event timing was anchored on the Difficulty Date as the primary occurrence date, with the Submission Date retained as a secondary timestamp for lag checks and validation of chronological splits.

Aircraft identity and fleet composition were represented using Registry Number (pseudonymized/hashed for analysis) and the Aircraft Make and Aircraft Model fields. Operational context was proxied by Receiving Region Code, the most consistently populated location-related attribute. The Discrepancy field provided the main NLP input text, and the JASC code served as the supervised label for the primary classification task when available. Any broader ATA chapter grouping was used only for aggregation, reporting, or visualization, not as the prediction target. Table 3 summarizes each variable’s operational role in modeling and evaluation, while Table 2 reports overall dataset scale and the skewed distribution of JASC codes.

Table 3. Core data fields, operational meaning, and modeling role in the recommended SDRS schema.

Field	Example / Format	Role in analysis	Notes
Aircraft/Tail (hashed)	a3f9...	Entity linkage	Enables repeat-defect and within-aircraft history without revealing identity
Timestamp	ISO datetime; optional bins (e.g., day/week/shift)	Temporal ordering, shift tests	Binning used when required for privacy or stability
Station	ICAO/IATA code	Site effects, station-shift evaluation	Supports cross-station robustness testing
Write-up text	Free text	Primary NLP input	Defect narrative; cleaned/normalized for abbreviations where feasible
Action taken text (if available)	Free text	Evidence/outcome context	Useful for retrieval, resolution summarization, and traceability
JASC code (supervised label)	ATA chapter/subchapter	Supervised label	Target for triage classification when present and reliable
Deferral indicators (if present)	MEL/CDL flag; deferral code	Operational constraint marker	Captures deferral behavior relevant to dispatch and risk
Closure status/time (if present)	Open/closed; minutes/hours	Outcome/efficiency metric	Supports time-to-close variability and operational impact analyses

3.4 Data governance, privacy, and de-identification

Because the dataset consisted of secondary operational technical-log records, the study implemented a privacy-by-design pipeline to reduce re-identification risk while preserving analytic utility. First, direct identifiers embedded in structured fields or free text (e.g., names, staff IDs, signatures, phone numbers, email addresses) were

removed or masked using rule-based patterns plus NLP-assisted detection appropriate for unstructured narratives [43]. Second, quasi-identifiers and linkage keys (e.g., aircraft/tail numbers and any internal identifiers) were pseudonymized via hashing so that repeat events within aircraft could be linked without exposing operational identities; text minimization was applied to avoid retaining unnecessary granularity that could enable linkage-based re-identification [44]. Third, de-identified files were stored in restricted-access repositories with role-based permissions, audit logs, and encrypted storage, and were governed by a defined retention schedule aligned with institutional policy.

The study involved no human participants and analyzed only de-identified operational records generated during routine line maintenance operations [43].

3.5 Problem formulation

This study approached the challenge of aircraft line-maintenance log analysis as a combined supervised classification and evidence-retrieval problem. The primary task (T1) was multi-class text classification, in which the input was the technical-log write-up and the target was the corresponding JASC code recorded in the SDRS data. The classifier was trained on the full observed in-distribution JASC code space within the training partition, rather than on a reduced set of dominant classes plus an “other” category. For interpretability, predictions and errors could be aggregated *post hoc* to ATA chapter groupings derived from the leading digits of the JASC codes, but this aggregation was used only for reporting and visualization and did not define the model’s prediction space. This formulation supports realistic defect categorization under a heavily long-tailed label distribution rather than coarse chapter-level screening [6].

Two secondary tasks complemented T1. First, similar-case retrieval was defined as a top- K retrieval task over historical technical-log cases to surface the most similar prior cases and their recorded outcomes or actions, thereby improving traceability and auditability in operational use [10]. Second, an optional severity- or urgency-proxy classification task was considered only if sufficiently consistent proxy labels were available. In the reported experiments, this optional task was not pursued because proxy labels were not of sufficient quality for reliable supervised evaluation. Accordingly, the reported results are limited to JASC code classification, retrieval evidence, and conformal uncertainty analysis. Given the strong label imbalance, the system is positioned as a decision-support tool rather than a system for autonomous decision-making, with outputs intended to be used alongside retrieved evidence, abstention signals, and human review.

3.6 Text preprocessing and normalization

Technical-log narratives were preprocessed to reduce noise and standardize language before modeling. When mixed-language entries might be present, language

detection was applied to route records through the appropriate normalization pathway. Text was then tokenized and normalized through spelling correction and character-level cleanup to mitigate common log artifacts (e.g., inconsistent casing, punctuation, and shorthand). A domain lexicon was used to expand and standardize maintenance abbreviations, fault codes, and common terms, thereby improving consistency across stations and writers, consistent with prior work showing that domain-aware processing improves extraction from unstructured maintenance text [15]. Structured tokens – such as “ECAM,” “BITE,” and message/fault codes – were preserved as meaningful technical units (rather than treated as generic words) to retain diagnostic signal and improve downstream classification and similarity matching, reflecting established practice in predictive-maintenance NLP pipelines that make use of free-form maintenance reports [45].

3.7 NLP triage model

The NLP triage component was implemented as a supervised text-classification module that assigned a score distribution over candidate JASC code labels for each technical-log write-up. The reported classifier used TF-IDF features with a linear SVM, selected because it is transparent, computationally efficient, and well suited to short, noisy operational text, while supporting reproducible evaluation under the deployment-aligned temporal and station-shift protocols in Section 3.10.

The TF-IDF plus linear SVM configuration was intentionally treated as a transparent baseline rather than a fully optimized mitigation model. No synthetic oversampling, cost-sensitive reweighting, class-weighted optimization, focal or class-balanced loss, or threshold-tuning strategy was applied. The rationale was to establish an interpretable, auditable reference point under realistic deployment conditions before introducing additional imbalance-mitigation or domain-adaptation methods. Accordingly, class imbalance was addressed primarily through macro-sensitive evaluation, shortlist reporting, uncertainty-aware abstention, and evidence-backed human review, rather than algorithm-level rebalancing.

The resulting label scores were used to generate ranked candidate labels for shortlist-based triage and to support downstream uncertainty-aware decision support in operational maintenance analytics [6]. Under this framing, the study emphasizes baseline robustness, retrieval support, and abstention behavior, while mitigation-focused analyses for rare-class and cross-station performance are reserved for subsequent work. A natural next step would be a simple sensitivity analysis – for example, class-weighted linear SVM – to test whether rare-class gains can be achieved without materially degrading calibration, abstention behavior, or auditability.

3.8 Retrieval-augmented NLP (evidence layer)

An evidence layer was implemented by structuring the historical technical log repository into a searchable case corpus. Each prior log entry was treated as a single

case, and when action-taken text was available, an optional resolution snippet was retained as a separate field to support concise evidence display. In the reported experiments, retrieval used term frequency-inverse document frequency (TF-IDF) representations with cosine similarity to identify the top- K most similar historical cases for each new write-up. The retrieved cases were presented as auditable evidence to support triage by identifying label-consistent historical precedents and associated maintenance context. To avoid information leakage, post-event resolution fields were not used as predictive inputs for the classifier and were retained only for controlled evidence display after retrieval [10, 25].

3.9 Conformal prediction layer (calibration and abstention)

A conformal prediction layer was applied on top of the trained models to produce calibrated uncertainty and an explicit abstention mechanism suitable for safety-adjacent decision support. For the multi-class task (e.g., JASC code prediction), the model outputs were converted into conformal prediction sets rather than a single top-1 label, enabling controllable marginal coverage at a pre-specified target of $1-\alpha$ [11]. For the binary repeat-risk task, the same calibration principle was used to provide uncertainty-aware risk outputs aligned with a target error/coverage level [14].

A three-way split was used, with the calibration partition reserved exclusively for conformal calibration. Let $\hat{p}_y(x)$ denote the model-predicted probability for class y on input x . The nonconformity score for the observed class y_i was defined as:

$$s_i = 1 - \hat{p}_{y_i}(x_i) \quad (1)$$

Given calibration scores $\{s_i\}_{i=1}^{n_{\text{cal}}}$, the conformal threshold at the target $1-\alpha$ was computed using the standard split-conformal quantile:

$$\hat{q}_{1-\alpha} = \text{Quantile}_{[(n_{\text{cal}}+1)(1-\alpha)]/n_{\text{cal}}}(\{s_i\}) \quad (2)$$

For a new input x the conformal prediction set was then:

$$\Gamma_{1-\alpha}(x) = \{y \in \mathcal{Y} : 1 - \hat{p}_y(x) \leq \hat{q}_{1-\alpha}\} = \{y \in \mathcal{Y} : \hat{p}_y(x) \geq 1 - \hat{q}_{1-\alpha}\} \quad (3)$$

An operational abstention policy was defined to prioritize traceability over forced automation: outputs were routed for review when the conformal set was insufficiently specific, implemented as a size rule.

$$\text{abstain}(x) = (|\Gamma_{1-\alpha}(x)| > m), \quad (4)$$

where m is a pre-defined maximum acceptable set size (Mortier et al., 2021). The same calibration split and thresholding logic was applied for the binary repeat-risk task to ensure uncertainty-aware outputs consistent with the chosen $1-\alpha$ [10].

3.10 Experimental design and validation strategy

Model evaluation used deployment-aligned validation to reduce optimistic bias from random splits and to test generalization under realistic operational variability. First, a chronological split approximated prospective deployment: models were trained on the early portion of the study window, a held-out calibration subset from the same period was reserved for conformal calibration, and final evaluation was performed on a later test period. This design reflects evidence that temporal drift can affect both discrimination and calibration and therefore should be assessed explicitly rather than masked by random splits [37, 36]. The resulting split sizes were 3,101 training events (70.0%), 443 calibration events (10.0%), and 886 test events (20.0%); the corresponding date boundaries and station/fleet coverage are reported in Table 4, Panel A.

Second, station generalization was evaluated using a region hold-out protocol in which models were trained on reports from a subset of Receiving Region Codes and evaluated on unseen regions (anonymized as S1–S2). Table 4, Panel B reports the composition of this station hold-out split across the full cleaned and deduplicated corpus, including held-out events, fleet overlap with training, and fleets exclusive to the held-out regions. This design was motivated by evidence that performance can vary across sites because of differences in documentation practices, local terminology, and case mix [36, 38].

For transparency, Table 4 reports two different levels of split information: Panel A summarizes the temporal train/calibration/test design, whereas Panel B summarizes the full-corpus composition of the station hold-out design. These counts are therefore not directly comparable to Table 8, which reports results on the smaller station-comparable temporal test subset used for the dataset-shift analysis. Fleets observed only in held-out stations were treated as out-of-distribution in fleet-stratified reporting and were either excluded from within-distribution summaries or explicitly flagged.

Table 4. Split counts and held-out station/fleet composition.
Panel A. Temporal split (chronological 70/10/20 by difficulty date).

Metric	All data	Train	Calibration	Test
Difficulty Date range	01 Jan 2021 – 22 Dec 2023	01 Jan 2021 – 14 Jan 2023	15 Jan 2023 – 24 Apr 2023	24 Apr 2023 – 22 Dec 2023
Events (unique SDRs)	4,430	3,101 (70.0%)	443 (10.0%)	886 (20.0%)
Stations (Receiving Region Code), n	10	10	8	8
Fleets (make-model), n	290	245	79	143
Aircraft (RegistryNumber), n	2,204	1,605	263	552
Operators (Operator Designator), n	97	90	34	41
JASC classes (JASC Code), n	306	279	95	160

Panel B. Station hold-out split (region generalization test).

Metric	Train stations (non-held-out)	Held-out stations (S1-S2)
Stations (ReceivingRegionCode), n	8	2
Events (unique SDRs)	1,766	2,664 (60.14%)
Fleets (make-model), n	177	192
Fleet overlap with training (n)	–	79
Fleets exclusive to held-out (OOD), n	–	113
Aircraft (RegistryNumber), n	935	1,312
Operators (Operator Designator), n	64	49
ATA/JASC classes (JASC Code), n	249	226

Note. Panel B reports the station hold-out composition across the full cleaned and deduplicated corpus, and is not directly comparable to Table 8, which reports performance on the smaller station-comparable temporal test subset.

3.11 Evaluation Metrics

Evaluation followed three layers: (i) triage prediction, (ii) historical-case retrieval, and (iii) uncertainty/abstention. Prediction performance was assessed using macro-F1 and weighted-F1, which capture class-balanced and prevalence-weighted behavior, together with top-1 and top-3 accuracy, which reflect shortlist-based triage use. PR-AUC was also reported, as it is informative under class imbalance [46]. Retrieval quality was evaluated using Recall@K and MRR. Label-consistency@K served as an operational proxy for relevance, defined as the share of retrieved cases with the same target JASC label as the query case; this was interpreted as precedent consistency rather than proof of the same engineering cause [47]. For the uncertainty layer, evaluation included empirical coverage, average prediction-set size, and the error-abstention trade-off. Calibration curves were used only as supplementary checks when probabilistic outputs were available [14]. Where feasible, two operational proxy checks were examined: simulated change in repeat-defect rates and time-to-close class accuracy.

To quantify sampling uncertainty, 95% confidence intervals were estimated for the main proportion-based metrics, particularly top-1 and top-3 accuracy, using binomial proportion methods applied to the relevant evaluation subsets. These intervals summarize sampling uncertainty within the available corpus and do not speak to external generalizability. For subgroup comparisons of top-1 accuracy, a two-sided test for the difference in proportions was applied to compare independent evaluation subsets, specifically the non-held-out versus station-held-out groups in the robustness analysis. The null hypothesis was equality of top-1 accuracy across groups. Approximate 95% confidence intervals for the accuracy difference, with corresponding p -values, are reported for this comparison. Macro-F1, weighted-F1, MRR, and

PR-AUC were retained primarily as point estimates only, because formal inference for these measures would require resampling-based procedures not used in the present study.

3.12 Incremental pipeline analysis

To assess the contribution of the main components of the proposed decision-support pipeline, an incremental analysis was conducted under the deployment-aligned temporal evaluation protocol. Rather than a controlled component-removal ablation, the analysis examined staged additions to the pipeline under evaluation. The configurations were: (i) the NLP triage classifier alone, (ii) the same classifier reported as a top-3 shortlist, (iii) the classifier combined with the TF-IDF retrieval evidence layer, and (iv) the full pipeline with the conformal uncertainty layer and abstention. This design was intended to show how each module contributes to operational decision support by improving shortlist utility, evidence grounding, and uncertainty-aware routing, without reproducing the full metric suite presented in the Results section. Accordingly, this analysis summarizes incremental pipeline contributions rather than lexicon-removal or reranker-removal ablations.

3.13 Implementation details (reproducibility)

Implementation details were reported to enable computational reproducibility and to make results inspectable beyond aggregate performance. The study documented the hardware and software environment used for training and inference, including operating system, CPU/GPU type, memory, and the exact versions of the ML stack (e.g., Python, deep learning framework, CUDA/cuDNN, and key libraries). This is consistent with reproducibility standards that emphasize explicit environment specification and dependency traceability [48]. The main hyperparameters were recorded for each model family, including tokenizer/max sequence length, learning rate and schedule, batch size, epochs, regularization (e.g., weight decay, dropout), class-weighting (where used), and random seeds. The exact training time windows used for temporal validation were also stated to prevent leakage and to make the evaluation replicable. Experiment versioning captured code commits, data snapshot identifiers, and model-artifact hashes so that each reported result could be traced to a specific configuration and dataset state, reflecting recommendations to standardize ML reporting and artifact management for reproducible research [49].

4. RESULTS

4.1 Dataset description

After cleaning and deduplication, the analytic corpus comprised 4,430 unique SDR events dated 1 January 2021 to 22 December 2023, spanning 10 receiving regions, 290 aircraft make-model combinations, 2,204 aircraft, and 97 operators.

The deployment-aligned temporal split yielded 3,101 training events (70.0%), 443 calibration events (10.0%), and 886 test events (20.0%). Across the corpus, 306 JASC classes were observed, of which 279 appeared in training and therefore defined the in-distribution label space for supervised evaluation. Overall corpus scale, class imbalance, and split composition are summarized in Table 2, with validation-specific station and fleet coverage reported in Table 4.

The JASC distribution was heavily imbalanced, with a small number of frequent classes and a pronounced long tail of rare classes. The largest class contained 582 events (13.14%), the top 10 classes accounted for 43.68% of all records, and the minimum class frequency was 1. This is a corpus-scale limitation, rather than a modeling inconvenience: supervision is dense for a few common labels but sparse for many rare ones. Accordingly, the study evaluates prediction over the full observed JASC space rather than a collapsed chapter-level task; later results should be interpreted in light of this sparse long-tail regime. This structure motivates the use of macro-sensitive metrics, shortlist reporting, retrieval support, and uncertainty-aware abstention, while suggesting hierarchical or few-shot extensions as appropriate future directions for rare-label handling.

Data completeness for the core modeling schema was high overall. Difficulty Date, Submission Date, Receiving Region Code, Discrepancy, and JASC Code each had 0% missingness, while aircraft identifiers showed only trace missingness (Registry Number: 12 records, 0.271%; Aircraft Make: 3, 0.068%; Aircraft Model: 5, 0.113%). The retained corpus was therefore effectively complete for the classification, retrieval, and robustness analyses, reported below. A compact summary of missingness is provided in Table 5.

Table 5. Missingness summary for the core modeling schema fields defined in Table 3, in the analytic corpus ($n = 4,430$).

Field	Missing (n)	Missing (%)
Difficulty Date	0	0
Submission Date	0	0
Receiving Region Code	0	0
Registry Number	12	0.271
Aircraft Make	3	0.068
Aircraft Model	5	0.113
Discrepancy	0	0
JASC Code	0	0

4.2 Prediction performance

Prediction performance was evaluated on the temporal test split defined in Section 3.10 using macro-F1, weighted-F1, top-1 accuracy, and top-3 accuracy. To preserve transparency under realistic operational drift, JASC codes not observed in training were treated as out-of-distribution (OOD); the primary discrimination metrics were therefore computed on the in-distribution temporal test subset, with the OOD count reported separately (Table 6). The baseline TF-IDF plus linear SVM showed moderate top-1 accuracy but substantially higher top-3 accuracy, indicating that the correct JASC code often appeared within a triage shortlist even when the top-ranked prediction was incorrect.

Macro-F1 was notably lower than weighted-F1, consistent with the heavy long-tail imbalance in JASC codes and the greater difficulty of predicting minority classes from short, noisy defect narratives. Weighted-F1 is therefore reported as a prevalence-weighted complement to macro-F1 and should not be interpreted as evidence of reliable performance on rare or safety-relevant classes. The classifier was trained and evaluated on the full observed in-distribution JASC code space, not on a reduced set of dominant classes plus a catch-all “other” category. For interpretability, misclassifications were aggregated after prediction into ATA chapter groupings derived from the leading digits of the JASC codes. Figure 1, therefore, provides a post hoc grouped diagnostic view of error structure, not a representation of the model’s training target space or deployed decision-support output. The grouped results show that many errors cluster within or between a small number of operationally dominant ATA chapter groupings, reinforcing the need to interpret aggregate accuracy together with imbalance-aware metrics, shortlist performance, and evidence-backed human review (Table 6; Fig. 1).

Table 6. Prediction performance on the temporal test set after label-based in-distribution filtering (labels observed in training).

Model	Evaluated temporal test subset	Top-1 accuracy	Top-3 accuracy	Macro-F1	Weighted-F1	Excluded test cases with OOD labels (<i>n</i>)
TF-IDF + Linear SVM (baseline)	848/886 (95.7%)	0.5094	0.6828	0.2386	0.4768	38

Note. The temporal test partition contains 886 cases. Of these, 38 carry JASC codes not observed in training and were therefore excluded from this classifier evaluation, leaving 848 in-distribution test cases. This denominator differs from Table 8, which reports a smaller robustness subset used for the station-shift comparison after additional filtering described in Section 4.5.

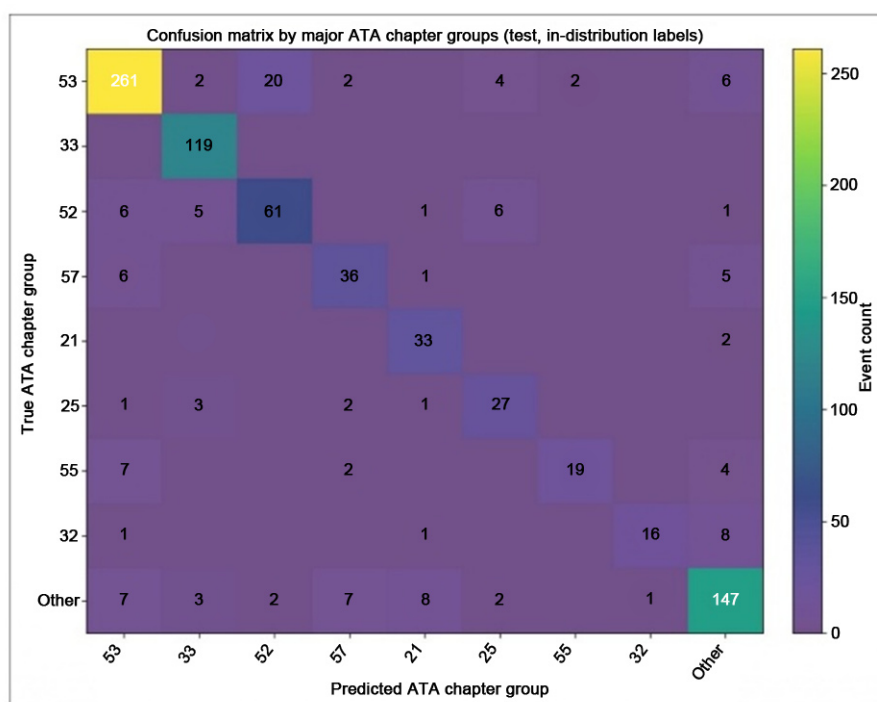


Fig. 1. Post hoc confusion matrix aggregated to major ATA chapter groups for the temporal test split.

4.3 Retrieval quality

Retrieval quality was evaluated using the evidence-layer protocol in Section 3.8 and the retrieval metrics specified in Section 3.11, applied to the temporal test queries defined in Section 3.10 (Table 4, Panel A). For each test event, the system retrieved the top- K most similar historical reports from the training corpus, consistent with a deployment-aligned indexing design. Similarity was computed using a TF-IDF vector-space representation with cosine similarity, without dense-embedding retrieval or reranking. Relevance was operationalized as JASC-code consistency: a retrieved case was considered relevant if its JASC code matched the query's, enabling computation of Recall@ K and MRR across the full temporal test set.

As shown in Figure 2, retrieval achieved Recall@1 = 0.368, increasing to Recall@3 = 0.530, Recall@5 = 0.600, and Recall@10 = 0.688, with MRR = 0.466. These results indicate that although the single nearest neighbor is label-consistent for only about one-third of test queries, a shortlist ($K \approx 5$ –10) often contains at least one JASC-code-consistent historical case, supporting evidence-grounded triage display. The same pattern is shown as a Recall@ K curve in Figure 3, highlighting the monotonic gain in recall as K increases and providing a practical basis for selecting an operational K given the trade-off between evidence breadth and review burden.

To capture the temporal evolution of sensor behaviour, the continuous multivariate telemetry stream is segmented into overlapping fixed-length windows. This segmentation strategy enables the framework to represent both localized disturbances and gradually developing fault patterns within a structured feature space. Each window consists of 50 consecutive samples, while a step size of 10 samples is employed to introduce substantial overlap between adjacent windows. This design choice ensures that short-duration anomalies as well as slowly accumulating subsystem degradations are preserved and consistently represented across multiple windows. Each window is assigned a binary label based on the presence of fault activity within its temporal span. A window is classified as faulty if any sample within the window corresponds to an injected fault event; otherwise, it is labelled as nominal. This inclusive labelling strategy improves sensitivity to brief but high-impact anomalies and avoids dilution of fault signatures across window boundaries, which can occur when faults span only a fraction of a window. As a result, the segmentation approach supports reliable detection of both transient and progressive fault behaviours. Figure 3 illustrates the window-level fault annotation timeline and the segmentation of the multivariate telemetry stream into overlapping analysis windows, highlighting how the adopted windowing strategy preserves temporal continuity while enabling structured downstream feature extraction and classification.

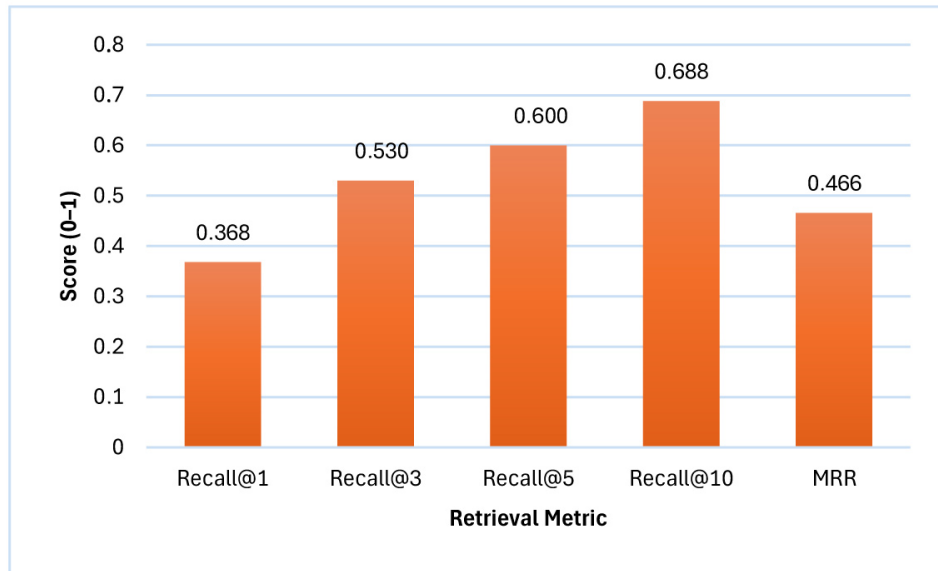


Fig. 2. Retrieval quality metrics on the temporal test set: Recall@K ($K = 1, 3, 5, 10$) together with Mean Reciprocal Rank (MRR). Relevance is defined as JASC-code label consistency between the retrieved case and the query.



Fig. 3. Recall@K curve for the temporal test set, showing the monotonic gain in recall as K increases from 1 to 10, with MRR = 0.47 annotated for reference. Relevance is defined as in Figure 2.

4.4 Uncertainty and abstention performance

Uncertainty behavior was evaluated using the split-conformal procedure and abstention rule defined in Section 3.9, applied to the dedicated calibration split from the temporal validation protocol (Section 3.10; Table 4, Panel A). Primary diagnostics were empirical coverage, mean prediction-set size, and the error-abstention trade-off (Section 3.11). Because the calibration split contains 443 cases across a highly sparse JASC label space, many rare classes have little or no direct calibration support. The reported conformal results should therefore be interpreted as marginal on the evaluation set, rather than as stable class-specific guarantees for very rare labels. Figure 4 shows the expected conformal trade-off: as the target coverage increases (i.e., as α decreases), prediction sets expand to preserve coverage over the temporal test period. Under the present long-tail JASC distribution, this expansion quantifies the additional ambiguity that must be tolerated to maintain a safety-aligned coverage target under class imbalance and temporal drift.

Under the operational abstention policy, in which cases are routed to review when the prediction set is insufficiently specific, stricter thresholds (smaller $S_{\max}$) increase abstention but reduce error among accepted outputs. This behavior is consistent with the intended safety posture of deferring when uncertainty is high. In operational terms, the abstention mechanism is particularly important for rare-label cases, which should default to evidence-backed human review even when overall empirical coverage remains close to the target. Table 7 summarizes coverage and prediction-set size across target coverage levels.

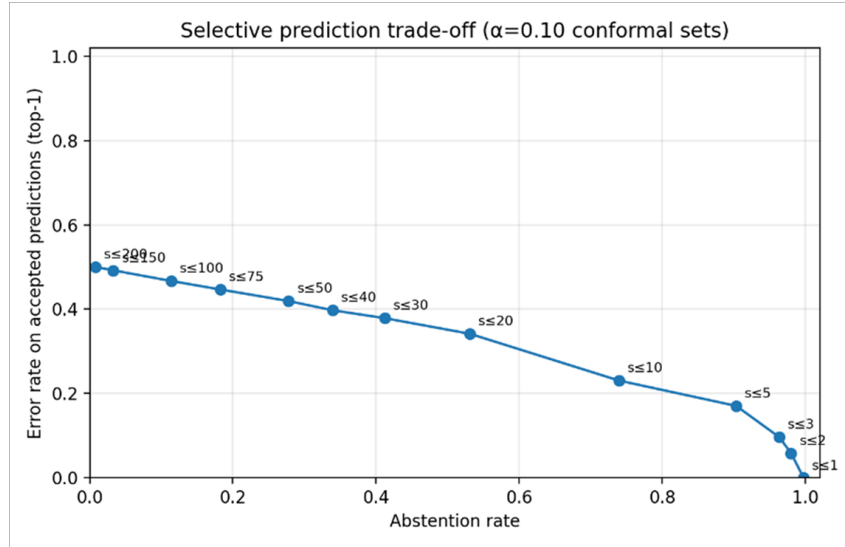


Fig. 4. Selective prediction trade-off on the temporal test set at $\alpha = 0.10$.

Error rate on accepted top-1 predictions is plotted against abstention rate, parameterized by the maximum acceptable prediction-set size m (labeled at each point as $s \leq m$; see equation (4) in Section 3.9). Stricter thresholds (smaller m) move the curve toward higher abstention and lower accepted-case error.

Table 7. Conformal prediction set efficiency on the temporal test set: empirical coverage and prediction-set size summary across target coverage levels (α).

Target coverage	α	$\hat{q}_{hat}$	Empirical coverage	Average set size	Median set size	25th percentile set size	75th percentile set size
0.80	0.20	0.993285	0.828033	9.016490	8	4	13
0.85	0.15	0.995652	0.870436	15.93168	12	6	24
0.90	0.10	0.997414	0.923439	40.87986	23	10	57
0.92	0.08	0.997839	0.943463	60.44405	30	13	85
0.95	0.05	0.998647	0.971731	122.2839	83	22	245
0.98	0.02	0.999292	0.990577	196.5524	266	91	277

4.5 Robustness under dataset shift

Robustness was evaluated using the deployment-aligned protocols described in Section 3.10, separating (i) time-held-out generalization via the temporal split and (ii) station-held-out generalization via a Receiving Region Code hold-out design (Table 4, Panels A and B). Table 6 reports performance on the full temporal test set after label-based in-distribution filtering (848/886 cases retained after excluding 38

cases with unseen ATA/JASC labels), whereas Table 8 reports the station-comparable temporal subset used for shift analysis ($n = 810$), comprising 337 non-held-out-station cases and 473 station-held-out cases.

Performance was lower under station hold-out than in the non-held-out temporal subset (Table 8). Top-1 accuracy was 0.504 (95% CI: 0.451–0.557) for non-held-out stations and 0.452 (95% CI: 0.408–0.497) for station-held-out regions; the corresponding value for the full station-comparable subset was 0.474 (95% CI: 0.440–0.508). The absolute Top-1 difference between the two station-defined subsets was 0.052 (approximate 95% CI: -0.018 to 0.122). A two-sided test for the difference in proportions did not indicate statistical significance at the 0.05 level ($p = 0.144$), suggesting that, although the observed direction is consistent with reduced transportability under station shift, the available sample does not support a strong inferential claim for Top-1 accuracy. Macro-F1 and weighted-F1 showed the same directional pattern, declining from 0.263 to 0.185 and from 0.468 to 0.421, respectively, with the largest absolute drop observed for macro-F1; these F1 differences are interpreted descriptively.

Overall, station-held-out cases appear more challenging, consistent with reduced transportability when models encounter documentation practices, terminology, and operational contexts not represented in the training data. Supporting diagnostics in Table 9 indicate that station-held-out reports tend to be shorter and less similar to the training corpus, consistent with site-specific abbreviations and reporting conventions that reduce classification and retrieval match quality. These findings reinforce the value of deployment-aligned validation: random splits can mask site effects, whereas station hold-out provides a more realistic view of transfer risk under operational variability.

Table 8. Robustness under dataset shift on the station-comparable temporal test subset: top-1 accuracy with approximate 95% confidence intervals, and descriptive F1 metrics for non-held-out stations versus station-held-out regions.

Split	n	Top-1 accuracy (95% CI)	Macro-F1	Weighted-F1
Time-held-out (temporal test, non-held-out stations)	337	0.504 (0.451–0.557)	0.263	0.468
Station-held-out (regions S1 and S2)	473	0.452 (0.408–0.497)	0.185	0.421
Temporal test (all stations, label-in-distribution; station-comparable subset)	810	0.474 (0.440–0.508)	0.209	0.437
Absolute difference (non-held-out – station-held-out)	–	0.052 (-0.018 to 0.122)	0.078	0.047

Note. 95% confidence intervals are reported for Top-1 accuracy only and were approximated using the observed subgroup proportions and sample sizes. Macro-F1 and weighted-F1 are presented as descriptive point estimates. The difference row is computed as non-held-out minus station-held-out.

Table 9. Dataset-shift diagnostics for robustness evaluation: narrative length and similarity-to-training for time-held-out versus station-held-out groups.

Group similarity to training	n	Median tokens in narrative	Median max
Time-held-out (non-held-out stations)	337	36	0.325201
Station-held-out (S1 and S2)	473	29	0.275161

4.6 Ablation results

Ablation analysis isolates the incremental contribution of each pipeline module under the deployment-aligned temporal evaluation protocol (Section 3.10; Table 4, Panel A). Starting from the classifier trained on the temporal training partition, successive modules add (i) a top-3 shortlist, (ii) a TF-IDF retrieval evidence layer, and (iii) a conformal uncertainty layer with an abstention rule (Section 3.9).

Table 10 summarizes compact, decision-relevant indicators for each module without repeating the full metric suite reported in Sections 4.2–4.5. For the classifier-only baseline, performance on the full temporal test set after label-based in-distribution filtering (Table 6) is: top-1 accuracy = 0.509, macro-F1 = 0.239, and weighted-F1 = 0.477. This should be distinguished from the lower top-1 value in Table 8 (0.474), which is computed on the station-based robustness subset rather than the full label-filtered temporal test set. Reporting the same classifier as a top-3 shortlist increases accuracy to 0.683, showing that many top-1 errors remain useful near-misses for human triage (Table 10; Fig. 5).

The retrieval layer supports decision-making without changing the classifier’s predicted label. Consistent with Section 4.3, retrieval achieves Recall@10 = 0.688 and MRR = 0.466 across the full temporal test set, indicating that a short evidence panel ($K \approx 10$) often contains at least one label-consistent historical case (Table 10; Fig. 5).

The conformal layer improves safety alignment by exposing uncertainty. At a 90% target, empirical coverage reaches 0.919, but the mean prediction-set size remains large (40.9 labels), reflecting long-tail ambiguity. Applying the abstention rule $|C(x)| \leq S_{\max}$ with $S_{\max} = 10$ leads the system to abstain on 0.735 of test cases while reducing the error rate among accepted outputs to 0.236. This supports the intended posture of abstaining under high uncertainty and automating only sufficiently specific cases (Table 10; Fig. 5).

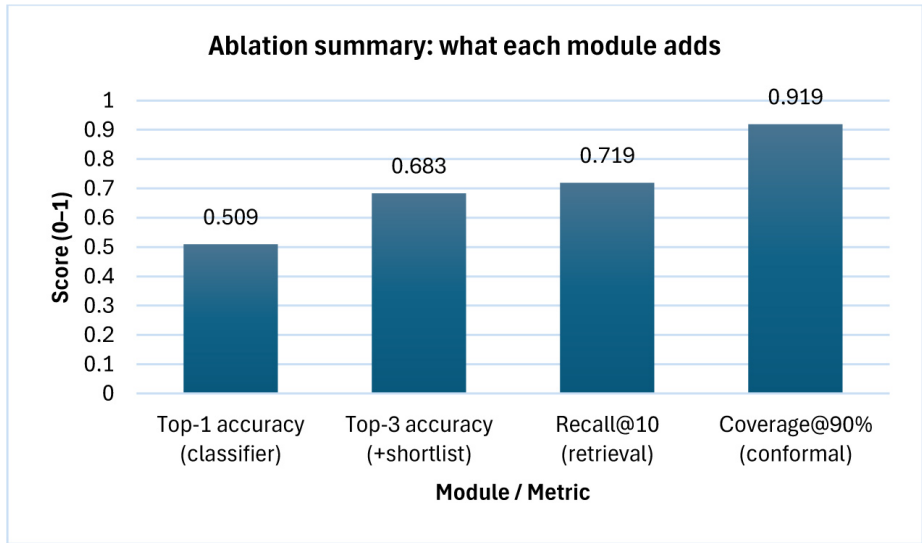


Fig. 5. Ablation summary showing incremental contributions of the classifier (top-1 accuracy), shortlist output (top-3 accuracy), retrieval evidence (Recall@10), and conformal uncertainty layer (coverage@90%), with abstention@Smax=10 and error among accepted cases reported for the conformal setting.

Table 10. Ablation summary showing the incremental contribution of each pipeline module on the temporal test set.

Configuration	Primary metric	Value
Classifier only	Top-1 accuracy	0.509
Classifier + shortlist	Top-3 accuracy	0.683
Retrieval evidence	Recall@10	0.688
Conformal + abstention	Coverage@90%	0.919

5. DISCUSSION

5.1 Key findings linked to RQs

Under the deployment-aligned temporal protocol (Table 4), the classifier shows moderate top-1 performance (0.509) but substantially higher top-3 utility (0.683), supporting shortlist-style triage rather than single-label automation. The macro-weighted F1 gap (0.239 vs. 0.477) indicates long-tail weakness under strong class imbalance (Table 2). The temporal test also includes OOD labels (38), so supervised scoring is reported only on in-distribution labels (Table 6). The evidence layer adds traceable precedent rather than changing predictions. Using the TF-IDF retrieval baseline, performance reaches Recall@10 = 0.688 and MRR = 0.466 (Figures 2–3),

implying that a compact evidence panel ($K \approx 5\text{--}10$) often includes a label-consistent historical case. This retrieval signal should be interpreted as evidence of similarity in historically observed defect-label patterns and precedents, not as a direct inference of the underlying engineering cause. Retrieval usefulness can degrade due to differences in site-specific documentation (Table 9).

Split-conformal behavior follows the expected tradeoff: higher target coverage (lower α) yields larger prediction sets (Table 7). With the operational abstention rule, tightening $S_{\max}$ increases abstention and lowers error among accepted outputs (Figure 4), operationalizing “defer when uncertain.” Efficiency remains the main constraint: at 90% target coverage, the mean set size is large (≈ 41) and abstention at $S_{\max} = 10$ is high (0.735), positioning the system as selective automation plus evidence-backed escalation rather than blanket auto-coding (Table 10). Overall, the pipeline is strongest as human-in-the-loop triage (shortlist + retrieved evidence + calibrated deferral) and weakest under long-tail labels, label novelty, and station-level shift (Tables 8–9).

5.2 Operational implications for line maintenance

The main operational value of the pipeline is improved consistency in the first-pass categorization of free-text technical log write-ups. Although top-1 accuracy is moderate, stronger shortlist performance supports a practical triage workflow in which maintainers are shown a small set of plausible JASC code candidates together with retrieved historical precedents. This can reduce idiosyncratic coding, improve shift-to-shift continuity, and create a more standardized path from narrative write-up to structured troubleshooting, particularly for high-frequency defect families where the model performs best.

The retrieval layer also supports repeat-risk screening by surfacing prior cases with consistent JASC codes and similar symptom wording, subsystem context, or fleet context. In this role, it is best treated as an early-warning aid rather than a deterministic risk score, especially under long-tail labels and station-level documentation differences. Conformal prediction adds an explicit escalation mechanism: when the model cannot be sufficiently specific, it should not force a label. Cases should therefore be routed for review when the conformal set is too large, confidence is diffuse, similarity to the training corpus is low, OOD indicators are present, or the case originates from shifted stations where transferability is weaker. As an additional operational guardrail, outputs should be presented to the maintainer only when shortlist concentration is sufficiently high or top-retrieval similarity exceeds a predefined threshold; otherwise, the case should default to review-required routing. Even in escalated cases, the system can still add value by attaching candidate JASC codes, top- K precedents, and an explicit reason for escalation, thereby keeping automation selective, ambiguity visible, and escalation evidence-backed.

5.2.1 Worked example of decision support in line maintenance

To illustrate the intended operational use of the proposed framework, Table 11 presents a de-identified case from the temporal test period involving an inoperative emergency exit light on a Boeing 737-family aircraft. The model ranked JASC 3350 first, and the retrieval layer returned several closely matched historical precedents with the same label and similar corrective actions, primarily battery pack replacement and post-maintenance testing. Although the evidence was coherent, the conformal prediction set remained large ($n = 72$), so the case would be flagged for review rather than silently auto-coded. Notably, even this coherent case produced a wide conformal set, illustrating that calibrated uncertainty can recommend review when qualitative evidence alone appears decisive. This example shows the intended role of the framework in practice: it does not replace maintenance judgment but provides a shortlist of plausible labels, attaches auditable historical precedents, and makes uncertainty explicit so that first-pass triage becomes faster, more consistent, and better supported.

Table 11. Example of the proposed decision-support output for a de-identified temporal test case.

Current write-up	Top-3 JASC candidates	Retrieved precedents	Conformal output	Recommended workflow action	Final human disposition
Emergency exit light inoperative at R1 door; removed and replaced emergency exit battery pack M1675/STA 322R with a serviceable unit in accordance with B737-800 AMM 33-51-06; test satisfactory.	1. JASC 3350 2. JASC 5240 3. JASC 3442	Case 1: R1 door emergency exit light inoperative; battery pack replaced; operational check satisfactory. Case 2: Aft left emergency exit lights inoperative; battery pack replaced; checks satisfactory. Case 3: Emergency exit light assembly near R1 door removed and replaced; test satisfactory. Case 4: Inoperative emergency exit battery pack replaced; test satisfactory. Case 5: Similar emergency exit battery-pack fault corrected by replacement, followed by a satisfactory test.	Prediction-set size = 72; status = review required.	Review required – the shortlist and retrieved precedents are coherent, but predictive uncertainty remains too high for silent auto-suggestion.	Human reviewer confirms JASC 3350 and records battery-pack replacement with satisfactory operational test.

5.3 Why evidence-grounding matters (auditability)

Evidence grounding makes NLP-based triage auditable in a safety-adjacent maintenance setting. By attaching the top- K retrieved historical cases to each recommendation, the system provides a traceable rationale that maintainers, supervisors, and QA personnel can inspect, challenge, and document when final coding differs from the model suggestion. This shifts the system from opaque prediction to reviewable decision support, which is more consistent with SMS-aligned oversight and post-event accountability.

Retrieved precedents, however, should be treated as inspectable context, not as self-validating proof of correctness, because historical SDRS labels may contain coding errors, local shorthand, or station-specific conventions. Their value lies in supporting QA review, supervisor oversight, post-event reconstruction, and identification of systematic failure modes such as station-specific jargon, long-tail labels, and emerging defect types. As an additional safeguard, frequently top-retrieved cases should be subject to periodic audit, with suspect or disputed precedent labels flagged, down-weighted, or excluded from future retrieval display.

5.4 Safety, limitations, and risk controls

This system is intended for decision support only. It provides candidate labels, retrieved precedents, and uncertainty or abstention signals, but it does not issue maintenance instructions or replace approved maintenance data, MEL/CDL constraints, or QA/SMS processes.

Several risks require explicit control. As noted above, data leakage must be prevented through strict deduplication, temporal split integrity, and leakage audits. Severe ATA/JASC imbalance should be understood as a corpus-scale limitation, since labeled evidence is concentrated in a small number of recurrent classes while many rare labels remain sparsely represented. The reported pipeline was therefore intentionally evaluated as a transparent baseline, not a mitigation study, and does not include class weighting, cost-sensitive learning, focal or class-balanced loss, synthetic oversampling, or station-adaptation methods. The findings should thus be read as a deployment-aligned reference point, with follow-up work testing whether hierarchical classification, few-shot learning, or other rare-class extensions can improve long-tail and cross-station performance without sacrificing auditability or uncertainty control.

The conformal layer also requires caution. Because the calibration split contains only 443 cases across a sparse label space, many rare classes likely have little or no direct calibration support; coverage should therefore be interpreted as marginal evaluation-set coverage rather than stable class-specific guarantees for very rare labels. Rare or potentially safety-critical defects should default to evidence-backed human review. Retrieved precedents should likewise be human-verified, with override logging, periodic QA review, and periodic audit of top-retrieved cases,

including flagging or exclusion of suspect precedent labels, to reduce feedback loops from mislabeled historical cases. Station effects may further limit cross-site transferability, so deployment should be monitored per station rather than assumed universal.

Operational guardrails should also be explicit. Recommendation outputs should be suppressed when shortlist concentration is weak, retrieval similarity falls below the threshold, the conformal set is too large, or when OOD or station-shift indicators are triggered; in such cases, the workflow should move directly to review-required status. Because drift is expected as vocabulary, fleets, documentation practices, and label space evolve, the system should be monitored continuously using recent-window performance, empirical coverage, abstention rate, retrieval relevance, shortlist concentration, similarity to the training corpus, and OOD rates. Deterioration in these indicators should trigger stricter review routing and recalibration or retraining before broader operational use.

5.5 Research and industry implications

Conceptual integration path into MRO/line-maintenance IT. A practical integration approach is to embed a thin decision-support service within existing maintenance IT rather than as a standalone tool. Conceptually, the pipeline can sit behind (or alongside) the operator's eTechLog/MIS as an event-enrichment layer: when a discrepancy is entered, the service returns (i) a shortlist of plausible ATA/JASC codes, (ii) top- K retrieved precedents, and (iii) an uncertainty/abstention flag that drives escalation ("review required"). The resulting outputs can be written back as structured fields (recommended codes, evidence links, uncertainty state) and displayed in familiar work queues for controllers, MCC, QA, or reliability engineering. From an implementation standpoint, this suggests a governance-friendly workflow: API-based integration, role-based access, immutable audit logs of model outputs and human overrides, and monitoring dashboards tracking abstention, OOD rates, and station-specific drift.

Research implications: moving from text classification to reliability outcomes. The results also point to the next research step: connecting NLP outputs to downstream operational and reliability metrics rather than treating classification accuracy as the terminal objective. Future work can incorporate ACMS/FDM/engine health signals (e.g., fault codes, exceedances, trends, flight-phase context) as complementary features to reduce ambiguity in long-tail labels and improve station transferability by anchoring text in sensor-derived context. In parallel, the pipeline should be evaluated against outcomes that matter to operators – repeat removals, repeat write-ups, delay minutes, dispatch reliability, MEL deferrals, and MTBUR/MTBF proxies – so that "evidence-grounded triage" can be assessed as an intervention that changes maintenance system performance. This integration would also enable closed-loop learning: retrieval could be conditioned not only on label consistency but on *outcome similarity* (e.g., cases that led to repeats or delays),

and conformal abstention thresholds could be tuned to optimize safety-aligned utility (minimizing avoidable errors while maintaining manageable review load).

6. CONCLUSIONS

This study demonstrates that operational free-text maintenance reports can be used to build an evidence-grounded, uncertainty-aware decision-support pipeline for aircraft line-maintenance triage under realistic deployment conditions. Using FAA SDRS narratives from 2021 to 2023, the results show that a lightweight TF-IDF plus linear SVM model can support defect-category triage, but its value is stronger as a shortlist-based aid than as a single-label automation tool, given the gap between moderate top-1 performance and stronger top-3 utility. The retrieval layer added practical value by surfacing label-consistent historical precedents, thereby strengthening interpretability and auditability. The conformal layer provided calibrated uncertainty handling and an explicit review-required pathway, with empirical coverage exceeding the nominal 90% target, although at the cost of large prediction sets under long-tail class imbalance.

Taken together, these findings answer the study questions in a balanced way. Technical-log text can support line-maintenance defect categorization, but chiefly as human-supportive triage rather than as autonomous coding. Retrieval augmentation improves decision support by making recommendations more evidence-grounded and traceable, although the present results show this contribution more clearly in retrieval relevance and auditability than in a definitive gain in robustness over the classifier alone. Conformal prediction performed as intended as a safety layer by maintaining target-level coverage and enabling abstention when predictions were insufficiently specific. Accordingly, the hypothesis concerning retrieval augmentation is partially supported, whereas the hypothesis concerning conformal prediction is supported.

The study makes three main contributions. First, it applies a deployment-aligned evaluation design, including temporal splitting, dedicated calibration, and station hold-out testing, to reduce optimistic bias and assess transferability under drift and site variation. Second, it identifies heavy ATA/JASC imbalance, out-of-distribution labels, and station-related narrative variation as core constraints on predictive reliability. Third, it shows how calibrated abstention can convert uncertainty into actionable escalation logic for governance-sensitive, safety-adjacent use. This paper should therefore be read as a transparent baseline study rather than a final optimized classifier. Future work should test class-weighted or cost-sensitive learning, retrieval-aware reranking, domain adaptation, and hierarchical or few-shot methods to improve rare-class and cross-station performance while preserving calibration, abstention behavior, and auditability. Overall, the findings support the system's use as a human-in-the-loop triage tool, with future operational pilots needed to evaluate both model metrics and maintenance-facing outcomes under drift-monitored, station-stratified deployment.

REFERENCES

- [1] Stanton I, Munir K, Ikram A, El-Bakry M. Predictive maintenance analytics and implementation for aircraft: Challenges and opportunities. *Syst Eng*. 2023;26(2):216–237. <https://doi.org/10.1002/sys.21651>
- [2] Scott MJ, Verhagen WJC, Bieber MT, Marzocca P. A systematic literature review of predictive maintenance for defence fixed-wing aircraft sustainment and operations. *Sensors*. 2022;22(18). <https://doi.org/10.3390/s22187070>
- [3] Shaukat S, Katscher M, Wu C-L, Delgado F, Larraín H. Aircraft line maintenance scheduling and optimisation. *J Air Transp Manag*. 2020;89:101914. <https://doi.org/10.1016/j.jairtraman.2020.101914>
- [4] Kála M, Lališ A, Vojtěch T. Analyzing aircraft maintenance findings with natural language processing. *Transp Res Procedia*. 2022;65:238–245. <https://doi.org/10.1016/j.trpro.2022.11.028>
- [5] Garcia J, Rios-Colque L, Peña A, Rojas L. Condition monitoring and predictive maintenance in industrial equipment: An NLP-assisted review of signal processing, hybrid models, and implementation challenges. *Appl Sci*. 2025;15(10):5465. <https://doi.org/10.3390/app15105465>
- [6] Dangut MD, Jennions IK, King S, Skaf Z. A rare failure detection model for aircraft predictive maintenance modelling. *Neural Comput Appl*. 2023;35:2991–3009. <https://doi.org/10.1007/s00521-022-07167-8>
- [7] Kostopoulos G, Davrazos G, Kotsiantis S. Explainable artificial intelligence-based decision support systems: A recent review. *Electronics*. 2024;13(14):2842. <https://doi.org/10.3390/electronics13142842>
- [8] Pelosi D, Cacciagrano D, Piangerelli M. Explainability and interpretability in concept and data drift. *Algorithms*. 2025;18(7):443. <https://doi.org/10.3390/a18070443>
- [9] Heredia Álvaro JA, Barreda JG. An advanced retrieval-augmented generation system for manufacturing quality control. *Adv Eng Inform*. 2025 Mar;64:103007. <https://doi.org/10.1016/j.aei.2024.103007>
- [10] Ludwig H, Schmidt T, Kühn M. An ontology-based retrieval augmented generation procedure for a voice-controlled maintenance assistant. *Computers in Industry*. 2025;159:104289. <https://doi.org/10.1016/j.compind.2025.104289>
- [11] Mortier T, Wydmuch M, Dembczyński K, Hüllermeier E, Waegeman W. Efficient set-valued prediction in multi-class classification. *Data Mining and Knowledge Discovery*. 2021;35(4):1435–1469. <https://doi.org/10.1007/s10618-021-00751-x>
- [12] Johansson U, Löfström T, Sönströd C, Löfström H. Conformal prediction for accuracy guarantees in classification with reject option. In: *International Conference on Modeling Decisions for Artificial Intelligence (MDAI 2023)*; 2023. p. 133–145. https://doi.org/10.1007/978-3-031-33498-6_9
- [13] Artelt A, Visser R, Hammer B. “I do not know! but why?” – Local model-agnostic example-based explanations of reject. *Neurocomputing*. 2023;558:126722. <https://doi.org/10.1016/j.neucom.2023.126722>
- [14] Campos MM, Farinhas A, Zerva C, Figueiredo MAT, Martins AFT. Conformal prediction for natural language processing: A survey. *Trans Assoc Comput Linguist*. 2024;12:1497–1516. https://doi.org/10.1162/tacl_a_00715

- [15] Bhardwaj AS, Veeramani D, Zhou S. Confidently extracting hierarchical taxonomy information from unstructured maintenance records of industrial equipment. *Int J Prod Res.* 2023;61(23):8159–8178. <https://doi.org/10.1080/00207543.2023.2167013>
- [16] Nanyonga A, Joiner K, Turhan U, Wild G. Does the choice of topic modeling technique impact the interpretation of aviation incident reports? A methodological assessment. *Technologies.* 2025;13(5):209. <https://doi.org/10.3390/technologies13050209>
- [17] Mohd Sharif MM, Maskat R, Baharum Z, Maskat K. A scoping review of topic modelling on online data. *Indones J Electr Eng Comput Sci.* 2023;31(3):1633–1641. <https://doi.org/10.11591/ijeecs.v31.i3.pp1633-1641>
- [18] Blanchy G, Albrecht L, Koestel J, Garré S. Potential of natural language processing for metadata extraction from environmental scientific publications. *SOIL.* 2023;9:155–168. <https://doi.org/10.5194/soil-9-155-2023>
- [19] Lam BD, Chrysafi P, Chiasakul T, Khosla H, Karagkouni D, McNichol M, Adamski A, Reyes N, Abe K, Mantha S, Vlachos IS, Zwicker JI, Patell R. Machine learning natural language processing for identifying venous thromboembolism: Systematic review and meta-analysis. *Blood Adv.* 2024;8(12):2991–3000. <https://doi.org/10.1182/bloodadvances.2023012200>
- [20] Szrama S. Optimizing aircraft engine longevity: A comparative framework for dynamically adaptive predictive maintenance using autoencoders, LSTMs, and Gaussian processes. *Eng Appl Artif Intell.* 2025;156:111199. <https://doi.org/10.1016/j.engappai.2025.111199>
- [21] Szrama S, Lodygowski T. Turbofan engine health status prediction with artificial neural network. *Aviation.* 2024;28(4):225–234. <https://doi.org/10.3846/aviation.2024.22554>
- [22] Park S, Kwon N, Ahn Y. Forecasting repair schedule for building components based on case-based reasoning and fuzzy-AHP. *Sustainability.* 2019;11(24):7181. <https://doi.org/10.3390/su11247181>
- [23] Wu H, Zhong B, Medjdoub B, Xing X, Jiao L. An ontological metro accident case retrieval using CBR and NLP. *Appl Sci.* 2020;10(15):5298. <https://doi.org/10.3390/app10155298>
- [24] Bousdekis A, Lepenioti K, Apostolou D, Mentzas G. A review of data-driven decision-making methods for Industry 4.0 maintenance applications. *Electronics.* 2021;10(7):828. <https://doi.org/10.3390/electronics10070828>
- [25] Peng B, Zhu Y, Liu Y, Bo X, Shi H, Hong C, Zhang Y, Tang S. Graph retrieval-augmented generation: A survey. *ACM Trans Inf Syst.* 2025;44(2):Article 35, 1–52. <https://doi.org/10.1145/3777378>
- [26] Rojas L, Hernandez B, Garcia J. A systematic review of intelligent agents, language models, and recurrent neural networks in industrial maintenance: Driving value creation for the mining sector. *Int J Intell Syst.* 2024;2025(1):9953223. <https://doi.org/10.1155/int/9953223>
- [27] Wang B, Wu J, Shi Y, et al. Structured reflective reasoning for precise medical knowledge graph retrieval augmented generation. *Health Inf Sci Syst.* 2025;13(1):76. <https://doi.org/10.1007/s13755-025-00390-2>
- [28] Lodygowski T, Szrama S. Unsupervised classification and remaining useful life prediction for turbofan engines using autoencoders and Gaussian mixture models: A comprehensive framework for predictive maintenance. *Appl Sci.* 2025;15(14):7884. <https://doi.org/10.3390/app15147884>

- [29] Szrama S, Szymański G, Mokrzan D. Aircraft propulsion health status prognostics and prediction. *Adv Sci Technol Res.* 2025;19(5):321–335.
<https://doi.org/10.12913/22998624/202232>
- [30] Kompa B, Snoek J, Beam AL. Second opinion needed: Communicating uncertainty in medical machine learning. *NPJ Digit Med.* 2021;4(1):4.
<https://doi.org/10.1038/s41746-020-00367-3>
- [31] Nemani V, Biggio L, Huan X, Hu Z, Fink O, Tran A, et al. Uncertainty quantification in machine learning for engineering design and health prognostics: A tutorial. *Mech Syst Signal Process.* 2023;205:110796. <https://doi.org/10.1016/j.ymssp.2023.110796>
- [32] Loftus TJ, Shickel B, Ruppert MM, Balch JA, Ozrazgat-Baslanti T, Tighe PJ, Efron PA, Hogan WR, Rashidi P, Upchurch GR, Bihorac A. Uncertainty-aware deep learning in healthcare: A scoping review. *PLOS Digit Health.* 2022;1(8):e0000085. <https://doi.org/10.1371/journal.pdig.0000085>
- [33] Singh Y, Hathaway QA, Keishing V, Salehi S, Wei Y, Horvat N, Vera-Garcia DV, Choudhary A, Mula Kh A, Quaia E, et al. Beyond post hoc explanations: A comprehensive framework for accountable AI in medical imaging through transparency, interpretability, and explainability. *Bioengineering.* 2025;12(8):879. <https://doi.org/10.3390/bioengineering12080879>
- [34] de Gelder E, Op den Camp O. How certain are we that our automated driving system is safe? *Traffic Inj Prev.* 2023;24(sup1):S131–S140.
<https://doi.org/10.1080/15389588.2023.2186733>
- [35] Convery O, Smith L, Gal Y, Hanuka A. Uncertainty quantification for virtual diagnostic of particle accelerators. *Phys Rev Accel Beams.* 2021;24(7):074602.
<https://doi.org/10.1103/PhysRevAccelBeams.24.074602>
- [36] dos Santos Silva GF, Barcellos Filho FN, Wichmann RM, da Silva Junior FC, Porto Chiavegatto Filho AD. Strategies for detecting and mitigating dataset shift in machine learning for health predictions: A systematic review. *J Biomed Inform.* 2025;170:104902. <https://doi.org/10.1016/j.jbi.2025.104902>
- [37] Guo LL, Pfohl SR, Fries J, Posada J, Fleming SL, Aftandilian C, Shah N, Sung L. Systematic review of approaches to preserve machine learning performance in the presence of temporal dataset shift in clinical medicine. *Appl Clin Inform.* 2021;12(4):808–815. <https://doi.org/10.1055/s-0041-1735184>
- [38] Cabanillas Silva P, Sun H, Rezk M, Roccaro-Waldmeyer DM, Fliegenschmidt J, Hulde N, von Dossow V, Meesseman L, Depraetere K, Stieg J, Szymanowski R, Dahlweid FM. Longitudinal model shifts of machine learning-based clinical risk prediction models: Evaluation study of multiple use cases across different hospitals. *J Med Res.* 2024;26:e51409. <https://doi.org/10.2196/51409>
- [39] Sáez C, Gutiérrez-Sacristán A, Kohane I, García-Gómez JM, Avillach P. EHRtemporalVariability: Delineating temporal dataset shifts in electronic health records. *GigaScience.* 2020;9(8):giaa079. <https://doi.org/10.1093/gigascience/giaa079>
- [40] Allgaier J, Pryss R. Practical approaches in evaluating validation and biases of machine learning applied to mobile health studies. *Commun Med.* 2024;4(1):76. <https://doi.org/10.1038/s43856-024-00468-0>
- [41] Gardner AL, Charlesworth M. How to write a retrospective observational study. *Anaesthesia.* 2023;78:521–525. <https://doi.org/10.1111/anae.15831>

- [42] Ghaferi AA, Schwartz TA, Pawlik TM. STROBE reporting guidelines for observational studies. *JAMA Surg.* 2021;156(6):577–578. <https://doi.org/10.1001/jamasurg.2021.0528>
- [43] Negash B, Katz A, Neilson CJ, Moni M, Nesca M, Singer A, Enns JE. De-identification of free text data containing personal health information: A scoping review of reviews. *Int J Popul Data Sci.* 2023;8(1). <https://doi.org/10.23889/ijpds.v8i1.2153>
- [44] Lulamba TE, Mutemaringa T, Tiffin N. Ten quick tips for protecting health data using de-identification and perturbation of structured datasets. *PLOS Comput Biol.* 2025;21(9):e1013507. <https://doi.org/10.1371/journal.pcbi.1013507>
- [45] Usuga-Cadavid JP, Lamouri S, Grabot B, Fortin A. Using deep learning to value free-form text data for predictive maintenance. *Int J Prod Res.* 2022;60(14): 4548–4575. <https://doi.org/10.1080/00207543.2021.1951868>
- [46] Opitz J. A closer look at classification evaluation metrics and a critical reflection of common evaluation practice. *Trans Assoc Comput Linguist.* 2024;12:820–836. https://doi.org/10.1162/tacl_a_00675
- [47] Valcarce D, Bellogín A, Parapar J, Castells P. Assessing ranking metrics in top-N recommendation. *Inf Retr J.* 2020;23:411–448. <https://doi.org/10.1007/s10791-020-09377-x>
- [48] Heil BJ, Hoffman MM, Markowetz F, Lee SI, Greene CS, Hicks SC. Reproducibility standards for machine learning in the life sciences. *Nat Methods.* 2021;18(10):1132–1135. <https://doi.org/10.1038/s41592-021-01256-7>
- [49] Pineau J, Vincent-Lamarre P, Sinha K, Larivière V, Beygelzimer A, d’Alché-Buc F, Fox E, Larochelle H. Improving reproducibility in machine learning research (a report from the NeurIPS 2019 reproducibility program). *J Mach Learn Res.* 2021;22(164):1–20. <https://doi.org/10.48550/arXiv.2003.12206>