International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902025

AI-Augmented Reconciliation: Automating Valid/Invalid Record Classification in Post-Load Validation

Sivasankararao Kavuri

Data Migration Lead, USA

Publication History: Received: 19.02.2026; Revised: 08.03.2026; Accepted: 12.03. 2026; Published: 15.03.2026.

ABSTRACT: Post-load validation, the step in which records freshly loaded into a target system are checked against their source and against business rules before they are trusted for downstream use, remains one of the least automated stages of most data migration and integration programs. Traditional reconciliation relies on deterministic checks, row counts, checksums, key existence, and hand-written business rules, that catch obvious breakage but struggle with subtler failure modes such as silent truncation, character encoding drift, partial duplicate loads, and transformation logic that behaves correctly on most records but not all of them. This article evaluates whether machine learning can be layered onto deterministic reconciliation to classify each loaded record as valid or invalid more accurately, more quickly, and with better-calibrated confidence than rule-based checks alone.

We propose a reference architecture for AI-augmented reconciliation that combines a deterministic rule engine, a supervised classification ensemble trained on historical validation outcomes, an unsupervised anomaly detection fallback for record categories with little labeled history, and a calibrated confidence scoring layer that routes ambiguous records to human review while allowing high-confidence classifications to proceed automatically. We evaluate this architecture against rule-based reconciliation, logistic regression, random forest, and isolation forest baselines on a simulated benchmark spanning five common target tables in an enterprise data platform, with invalid records injected across six recurring root cause categories.

Across the benchmark, the hybrid ensemble achieved an F1 score of 0.91 for invalid record classification, compared with 0.56 for rule-based reconciliation alone, while reducing mean time to classify a load batch from 26 hours to 4.8 hours by the tenth simulated load cycle. Automated classification coverage rose from 41 percent to 94 percent of loaded records over the same period, as the feature store accumulated a growing history of confirmed valid and invalid outcomes. Referential breaks and transformation mismatches remained the hardest categories to classify reliably across every method tested, underscoring that record-level context, not just column-level statistics, is often required to catch the most consequential invalid records.

We close with a discussion of design trade-offs, current limitations, threats to validity, and directions for future research, including active learning for feature store growth and standardized benchmarks for post-load validation classifiers.

KEYWORDS: post-load validation, data reconciliation, record classification, anomaly detection, class imbalance, machine learning, data migration assurance, data quality, human-in-the-loop review, confidence calibration.

I. INTRODUCTION

1.1 Motivation

Every data migration, integration, or ETL pipeline eventually faces the same question after a load completes: did the records that landed in the target system actually load correctly? Post-load validation answers this question by reconciling loaded records against their source and against a set of business rules, classifying each record as valid or invalid before downstream consumers trust it. Rahm and Do (2000) and the broader data cleaning literature they helped establish catalogued the instance-level errors, missing values, formatting inconsistencies, and duplicates, that reconciliation checks must catch, and subsequent large-scale validation frameworks (Schelter et al., 2018; Breck et al., 2019) operationalized many of these checks as declarative, automatable rules embedded directly in data pipelines.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

Deterministic reconciliation checks, row counts, checksums, referential integrity checks, and hand-written business rules, remain the backbone of post-load validation because they are transparent and easy to justify to an auditor. However, they share a structural limitation: they can only catch the failure modes their authors anticipated. Silent truncation, subtle character encoding drift, partial duplicate loads that pass a naive row count check, and transformation logic that behaves correctly on the vast majority of records but fails on an unanticipated edge case, all recur across enterprise data platforms yet often escape deterministic checks entirely. This article asks whether machine learning, layered onto rather than replacing deterministic reconciliation, can close this gap.

1.2 Problem Statement

Three practical problems motivate this work. First, coverage: deterministic rules cover only the failure modes anticipated at design time, leaving newly emerging or subtly compounding error patterns undetected until they surface downstream. Second, class imbalance: invalid records are, by design, a small minority of any healthy load batch, which makes naive classification approaches prone to either missing rare invalid records or flooding reviewers with false positives, a problem well documented in the broader imbalanced learning literature (He and Garcia, 2009; Chawla, Bowyer, Hall, and Kegelmeyer, 2002). Third, trust and governance: any automated classification of a record as valid or invalid must be explainable and auditable, since an incorrect automatic classification, in either direction, can either corrupt downstream data or waste reviewer time on records that were actually fine.

1.3 Contributions

- **Reference architecture.** We present a reference architecture for AI-augmented reconciliation that layers a supervised classification ensemble and an unsupervised anomaly detection fallback on top of deterministic reconciliation rules, with calibrated confidence scoring and governed human review.
- **Comparative evaluation.** We evaluate rule-based reconciliation, logistic regression, random forest, isolation forest, and a hybrid ensemble on a simulated benchmark spanning five target tables and six invalid record root cause categories.
- **Class imbalance analysis.** We explicitly evaluate class imbalance handling techniques and report their effect on invalid record recall, a dimension frequently underreported in production data quality tooling.
- **Governance guidance.** We provide a RACI matrix, a confidence scoring framework, and a design trade-off summary intended to help practitioners adopt AI-augmented reconciliation without sacrificing auditability.

1.4 Structure of the Article

Section 2 reviews prior work on data validation, anomaly detection, and imbalanced classification. Section 3 characterizes the post-load validation problem in more detail. Section 4 introduces the quality framework and maturity model used throughout the article. Section 5 presents the reference architecture, and Section 6 describes the machine learning techniques used for classification. Section 7 addresses class imbalance, and Section 8 describes the human review and governance workflow. Section 9 describes the resulting system and module architecture. Section 10 describes the evaluation setup, Section 11 reports results, and Section 12 presents three illustrative case studies. Sections 13 through 16 discuss the findings, limitations, threats to validity, and future work, and Section 17 concludes the article.

II. BACKGROUND AND RELATED WORK

2.1 Data Validation and Quality Verification

Rahm and Do (2000) provide an early, comprehensive taxonomy of data cleaning problems that continues to inform reconciliation check design today, distinguishing schema-level issues from instance-level issues such as missing values, duplicates, and formatting inconsistencies. Chu, Ilyas, Krishnan, and Wang (2016) and Abedjan et al. (2016) survey the broader data cleaning and error detection landscape and observe that no single detection technique dominates across every error type, a finding this article's hybrid ensemble design in Section 6 builds on directly. Deequ (Schelter et al., 2018) and the TensorFlow Extended data validation component (Breck et al., 2019; Baylor et al., 2017) operationalize declarative validation rules directly within production data and machine learning pipelines, demonstrating that rule-based validation can scale to very large datasets when properly engineered, even though the rules themselves remain limited to anticipated failure modes.

HoloClean (Rekatsinas, Chu, Ilyas, and Re, 2017) and ActiveClean (Krishnan, Wang, Wu, Franklin, and Goldberg, 2016) both move beyond purely deterministic rules, using probabilistic inference and human-in-the-loop active learning respectively to detect and repair errors that rule-based systems miss. Arocena, Glavic, Mecca, Miller, Papotti, and

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

Santoro (2015) introduce a controlled error injection methodology for evaluating data cleaning algorithms, which this article's simulated benchmark in Section 10 adapts for the post-load validation setting.

2.2 Anomaly Detection

Chandola, Banerjee, and Kumar (2009) provide the foundational survey of anomaly detection techniques, organizing the field around supervised, semi-supervised, and unsupervised approaches depending on the availability of labeled anomalies, a framing directly relevant to the maturity progression described in Section 4.3, where organizations typically lack labeled invalid records early on. Breunig, Kriegel, Ng, and Sander (2000) introduce the local outlier factor, a density-based method for detecting outliers whose abnormality depends on local rather than global data density, and Liu, Ting, and Zhou (2008) introduce isolation forest, an ensemble method that isolates anomalies through random partitioning rather than modeling normal behavior directly. Both methods underlie the unsupervised anomaly detection fallback described in Section 6.2, which this article's architecture relies on for record categories with insufficient labeled history.

2.3 Imbalanced Classification

Invalid records are, by construction, a minority class in any reasonably healthy load batch, which makes post-load validation a canonical imbalanced classification problem. Chawla, Bowyer, Hall, and Kegelmeyer (2002) introduce the synthetic minority over-sampling technique, generating synthetic minority class examples to rebalance training data, and He and Garcia (2009) survey the broader landscape of approaches to learning from imbalanced data, including resampling, cost-sensitive learning, and ensemble methods. Provost and Fawcett (2001) address a closely related concern, arguing that classifier evaluation under class imbalance and shifting misclassification costs should rely on threshold-independent measures such as the receiver operating characteristic curve rather than accuracy alone, a principle this article follows by reporting precision, recall, and F1 score rather than raw accuracy throughout Section 11.

2.4 Production Machine Learning Systems

Sculley et al. (2015) catalogue the hidden technical debt that accumulates in production machine learning systems, including entanglement between features and pipelines and the risk of undeclared consumers of a model's output, concerns directly relevant to embedding a classification ensemble inside a reconciliation pipeline that other systems depend on. Polyzotis, Roy, Whang, and Zinkevich (2018) survey data lifecycle challenges specific to production machine learning, including data validation and monitoring for distribution drift, both of which this article's confidence calibration and feedback loop, described in Sections 6 and 9, are designed to address.

2.5 Data Quality Foundations and Migration Practice

Wang and Strong (1996), Redman (1998), and Batini and Scannapieco (2016) provide the data quality vocabulary, accuracy, completeness, consistency, and timeliness, that underlies the classification quality dimensions introduced in Section 4. English (1999), Naumann (2014), and Fan and Geerts (2012) extend this vocabulary with practical profiling and dependency-based detection techniques. Dasu and Johnson (2003) and Ganti and Sarma (2013) both treat data cleaning as an exploratory, iterative practice rather than a one-time exercise, a framing consistent with the continuous feedback loop described in this article's reference architecture. Haller (2009) proposes patterns for industrializing data migration in standard software implementation projects, arguing that migration, and by extension the post-load validation that follows it, should be treated as a repeatable engineering discipline.

2.6 Gaps in the Current Literature

Three gaps motivate the present work. First, the data validation literature reviewed in Section 2.1 focuses heavily on declarative, rule-based validation and gives comparatively little attention to how machine learning classifiers should be layered on top of, rather than in place of, deterministic rules. Second, the anomaly detection and imbalanced classification literature reviewed in Sections 2.2 and 2.3 is rarely evaluated specifically in the post-load validation setting, with its characteristic mix of structured business rules, referential dependencies, and severe class imbalance. Third, none of the reviewed literature proposes a governance workflow that explicitly connects classifier confidence to human review allocation in a way suitable for an auditable enterprise reconciliation process. This article contributes toward closing all three gaps.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

III. THE POST-LOAD VALIDATION PROBLEM

3.1 Post-Load Validation Challenge Categories

Post-load validation challenges fall into a small number of recurring categories, summarized in Table 1. Distinguishing these categories matters because, consistent with the observation in Section 2.1 that no single detection technique dominates across every error type, each category calls for a different mix of deterministic checks and learned signals.

Table.1. Post-load validation challenge categories

Challenge Category	Description	Representative Example
Silent truncation	A field value is cut short during load without raising an error	A long product description silently truncated to fit a narrower target column
Encoding drift	Character encoding conversion introduces corrupted or substituted characters	Accented characters in a customer name rendered as unreadable symbols after load
Partial duplicate load	A batch is partially reprocessed, creating duplicate records for some but not all keys	A retried load job re-inserts a subset of records that had already loaded successfully
Referential break	A loaded record references a parent or lookup record that does not exist in the target	An order line item loads successfully while its parent order header fails to load
Transformation mismatch	A transformation rule produces an incorrect value for a subset of records	A currency conversion rule applies the wrong rate for a handful of less common currency codes
Timing lag	A record loads before a dependency it relies on has completed loading	A shipment record loads before the inventory adjustment it depends on has been committed

3.2 Reconciliation Check Types

Table 2 summarizes the deterministic reconciliation check types most commonly used in practice, each of which forms part of the rule-based baseline evaluated in Section 11 and part of the feature set used by the supervised classifiers described in Section 6.1.

Table.2. Reconciliation check types used in post-load validation.

Check Type	Description	Challenge Categories Best Covered
Row count reconciliation	Compares the number of records extracted from source to the number loaded to target	Partial duplicate load, gross load failure
Checksum or hash comparison	Compares a computed hash of record content between source and target	Silent truncation, encoding drift
Key existence check	Confirms that every foreign key reference resolves to an existing target record	Referential break, timing lag
Business rule validation	Evaluates domain-specific rules, such as valid value ranges or required field combinations	Transformation mismatch
Statistical distribution check	Compares aggregate statistics, such as means and value distributions, between source and target	Transformation mismatch, subtle systemic drift

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

3.3 Root Causes of Invalid Records

Table 3 summarizes the root causes that most often generate invalid records, mapped to the challenge categories in Table 1. These root causes directly inform the feature engineering and detection techniques introduced in Section 6.

Table.3. Root causes of invalid records.

Root Cause	Description	Typical Detection Difficulty
Source system schema drift	The source system changes a field's format or meaning without notice	Moderate; often catchable by statistical distribution checks once a baseline exists
Transformation logic edge cases	Transformation code handles common cases correctly but mishandles rare ones	High; rare inputs may not appear in test data used to validate the transformation
Concurrent load interference	Overlapping load jobs interact in ways not anticipated by either job's design	High; symptoms may appear intermittently and are hard to reproduce
Incomplete dependency sequencing	Load jobs run out of the order their data dependencies require	Moderate; detectable through key existence and timing checks if dependencies are well documented

IV. A FRAMEWORK FOR VALID/INVALID CLASSIFICATION QUALITY

4.1 Valid/Invalid Classification Quality Dimensions

Following the data quality vocabulary established by Wang and Strong (1996) and applied specifically to record-level classification, this article organizes classification quality around six dimensions, defined operationally in Table 4.

Table.4. Valid/invalid classification quality dimensions.

Dimension	Definition	Representative Violation
Invalid precision	The extent to which records classified as invalid are actually invalid.	A healthy record flagged as invalid because its value falls outside an overly narrow learned threshold.
Invalid recall	The extent to which actually invalid records are correctly classified as invalid.	A silently truncated field that passes classification because truncation alone did not trigger any feature.
Confidence calibration	The extent to which a reported confidence score matches the empirical likelihood of correctness.	A classifier reporting 95 percent confidence on a class of records where it is actually correct only 70 percent of the time.
Explainability	The extent to which a classification decision can be traced to specific evidence.	An invalid classification with no indication of which check or feature drove the decision.
Feedback latency	The time between a classification decision and its confirmation being available for retraining.	Confirmed invalid records that are not fed back into the feature store for several load cycles.
Coverage	The proportion of loaded records for which a classification decision, valid or invalid, is produced.	Records skipped entirely because they fall outside every rule and every trained classifier's scope.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

4.2 From Manual Reconciliation to AI-Augmented Reconciliation

Table 5 contrasts the traditional manual and rule-based reconciliation model with the AI-augmented model proposed in this article.

Table.5. Comparison of manual/rule-based and AI-augmented reconciliation approaches.

Dimension	Manual/Rule-Based Approach	AI-Augmented Approach
Detection basis	Fixed, hand-written rules covering anticipated failure modes	Learned classifiers and anomaly detectors covering both anticipated and emergent patterns
Confidence signal	Binary pass or fail per rule, no graded confidence	Calibrated confidence score per record, informing review prioritization
Handling of class imbalance	Not explicitly addressed; every rule violation treated uniformly	Explicit imbalance handling techniques described in Section 7
Review allocation	Uniform or ad hoc review of all flagged records	Review effort concentrated on low-confidence and high-impact records
Learning over time	Rules updated manually when a new failure mode is discovered	Feature store and models updated automatically as confirmed outcomes accumulate

4.3 An AI-Augmented Reconciliation Maturity Model

Table 6 defines four maturity levels used throughout the evaluation in Sections 10 and 11, reflecting how organizations typically progress from manual reconciliation toward continuous, AI-governed validation.

Table.6. AI-augmented reconciliation maturity model levels

Maturity Level	Characteristics	Typical Classification Basis
Manual	Reviewers manually inspect samples of loaded records against source extracts	None
Rule-Augmented	Deterministic checks flag violations automatically for review	Rule-based reconciliation
ML-Assisted	Supervised classifiers and anomaly detectors propose classifications for review	Logistic regression, random forest, or isolation forest
Continuous AI-Governed	A calibrated hybrid ensemble classifies most records automatically, with confidence-based escalation and continuous retraining	Hybrid ensemble

V. PROPOSED REFERENCE ARCHITECTURE

5.1 Design Goals

The architecture is organized around four goals drawn from the gaps identified in Section 2.6. Machine learning should augment, not replace, deterministic reconciliation, so that the transparency of rule-based checks is preserved for the failure modes they already cover well. Every classification should carry a calibrated confidence score that determines whether it can proceed automatically or requires human review. The system should degrade gracefully when labeled history is scarce, falling back to unsupervised anomaly detection rather than refusing to classify at all. Finally, confirmed outcomes, whether from automatic high-confidence classification or human review, should feed back into the feature store, so that classification quality improves as more load cycles complete.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

5.2 Architectural Overview

Figure 12 shows the resulting reference architecture as a pipeline running from post-load extraction through governed approval. A post-load record extract, comprising the newly loaded records and their corresponding source values, first passes through the deterministic reconciliation rule engine described in Table 2. Records not conclusively resolved by rules alone proceed to the machine learning classification ensemble, which draws on historical record outcomes stored in a feature store to score each record. A confidence calibration layer converts raw model output into an actionable confidence band, described further in Section 7.2. High-confidence classifications populate a reconciliation dashboard for lightweight spot-checking, while lower-confidence records route to a human reviewer queue. Approved classifications become the authoritative valid or invalid label for each record and feed back into the feature store, improving classification for subsequent load cycles.

Figure 12. Reference Architecture for AI-Augmented Post-Load Validation and Record Reconciliation

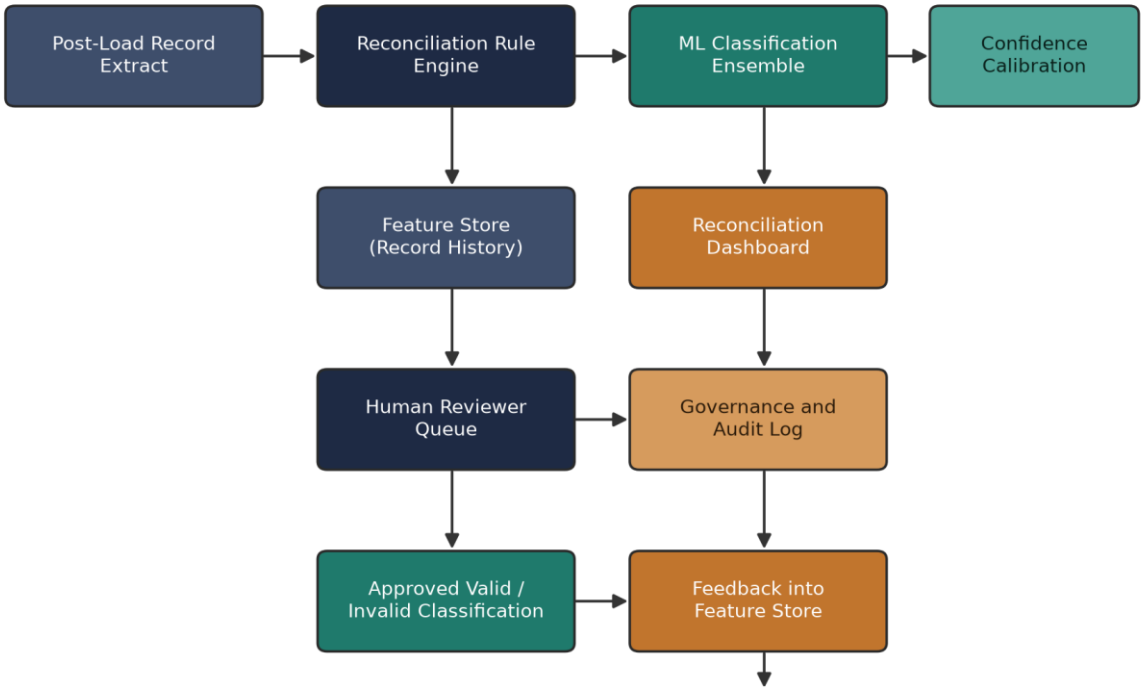

Figure.12. Reference architecture for AI-augmented post-load validation and record reconciliation

5.3 Module Descriptions

- **Post-load record extract.** A structured extract of every newly loaded record together with its corresponding source value, used as the unit of work for reconciliation.
- **Reconciliation rule engine.** The deterministic check layer summarized in Table 2, applied to every record before any machine learning classification occurs.
- **ML classification ensemble.** The supervised and unsupervised techniques described in Section 6, scoring records not conclusively resolved by deterministic rules.
- **Confidence calibration.** A calibration layer that converts raw model output into an actionable confidence band, described in Section 7.2.
- **Feature store (record history).** A growing store of historical record features and confirmed outcomes, used both as training data and as retrieval context for the classification ensemble.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

- **Reconciliation dashboard.** A shared view of classification status, confidence, and supporting evidence used by both automated spot-checking and human reviewers.
- **Human reviewer queue.** A queue of records requiring human judgment, prioritized by confidence and business impact, described further in Section 8.
- **Governance and audit log.** An append-only record of every classification decision, its confidence at the time of decision, and the identity of any human reviewer involved.
- **Approved classification and feedback loop.** The final, approved valid or invalid label, fed back into the feature store to improve future classification.

VI. MACHINE LEARNING TECHNIQUES FOR VALID/INVALID CLASSIFICATION

6.1 Supervised Classification

When sufficient labeled history exists, the architecture trains supervised classifiers directly on confirmed valid and invalid outcomes, using the reconciliation check outputs from Table 2 alongside record-level features such as field length, value distribution percentile, and time since the record's dependencies last loaded successfully. Table 7 summarizes the supervised classifier configurations evaluated in this study.

Table.7. Supervised classification technique configurations.

Technique	Key Configuration	Notes
Logistic regression (baseline)	L2-regularized, class weights inversely proportional to class frequency	Fast, interpretable coefficients; limited capacity for nonlinear interaction effects
Random forest	300 trees, max depth 8, class-weighted splitting criterion	Strong performance on structured, mixed-type features; more expensive to explain per-record
Hybrid ensemble (proposed)	Weighted combination of random forest, isolation forest, and rule engine signal, weights tuned on a validation split	Best overall performance; used in the reference architecture

6.2 Unsupervised Anomaly Detection as a Fallback

Consistent with the maturity progression in Table 6, newly onboarded target tables or record categories typically lack enough labeled history to train a reliable supervised classifier. For these cases, the architecture falls back on unsupervised anomaly detection, following Chandola, Banerjee, and Kumar (2009), to flag records that deviate from the bulk of the loaded batch without requiring prior labels. Table 8 summarizes the anomaly detection configurations evaluated in this study.

Table.8. Unsupervised anomaly detection technique configurations.

Technique	Key Configuration	Notes
Isolation forest	200 trees, contamination parameter tuned per target table	Effective for global outliers; less sensitive to local density variation
Local outlier factor	20 nearest neighbors, contamination parameter tuned per target table	Effective for local density-based anomalies; more computationally expensive at scale
Statistical control limits	Column-level control limits derived from historical distributions	Fast and simple; limited to single-column anomalies, misses cross-field interactions

6.3 Confidence Patterns Across Record Characteristics

Figure 1 visualizes a vector field of record drift, showing how records that begin in the invalid region of feature space tend to move toward the valid region across successive reprocessing passes, with vector direction and color indicating the magnitude of that movement. Most drift converges toward the origin, representing the valid region, but a visible

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

subset of vectors near the lower feature ranges point away from convergence, corresponding to records whose underlying root cause, typically a referential break or transformation mismatch, is not resolved by simple reprocessing alone.

Figure 1. Vector Field of Record Drift from the Invalid Region Toward the Valid Region Across Reprocessing Passes

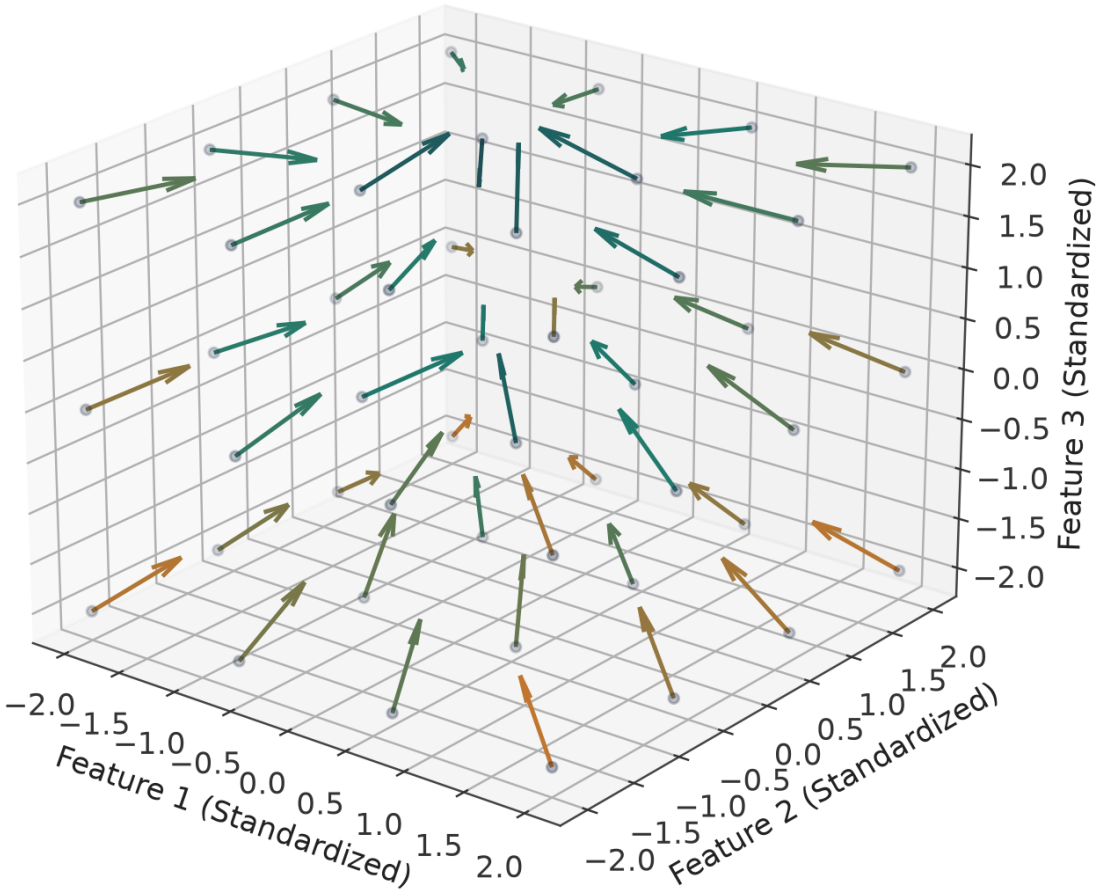

Figure.1. Vector field of record drift from the invalid region toward the valid region across reprocessing passes

Figure 2 traces individual record trajectories across successive reprocessing passes as curved paths through feature space. Records that ultimately converge to a valid classification, shown in teal, tend to follow smoother, more direct paths, while the smaller number of records that remain invalid, shown in amber, follow more erratic paths that do not settle near the valid region even after several passes, suggesting that path smoothness itself could serve as an additional feature for future classifier iterations.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

Figure 2. Record Feature Trajectories Across Successive Reprocessing Passes (Path Plot)

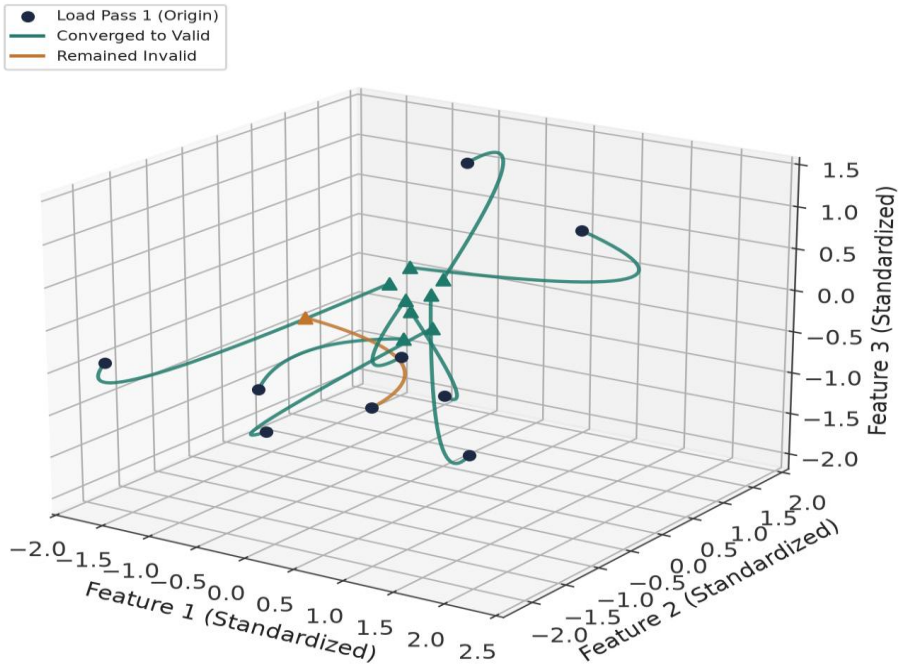

Figure.2. Record feature trajectories across successive reprocessing passes (path plot).

Figure 3 presents classification confidence as a cylindrical surface wrapped over the cyclic position of a batch within its recurring load schedule and the batch's progress through its load window. Confidence is highest during the early and late portions of a batch cycle and dips modestly midway through, a pattern consistent with load jobs that experience the greatest resource contention, and therefore the most timing-related invalid records, during peak processing windows.

Figure 3. Classification Confidence Wrapped Over the Cyclic Batch Load Schedule (Cylindrical Surface)

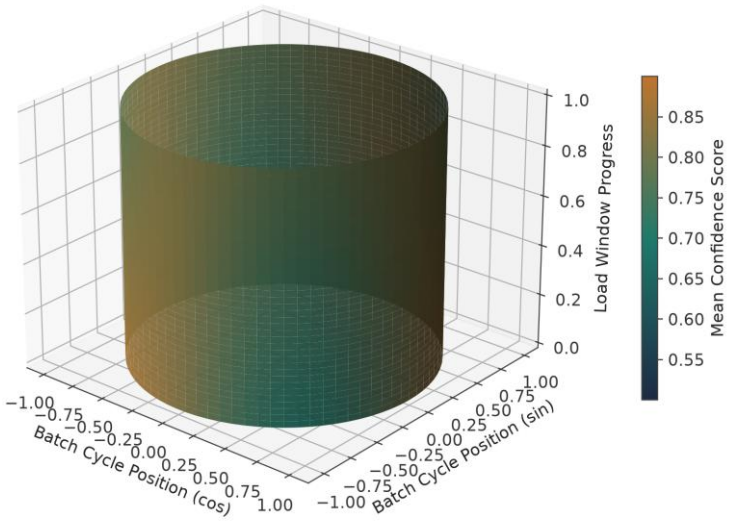

Figure.3. Classification confidence wrapped over the cyclic batch load schedule (cylindrical surface).

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

Figure 4 reports mean detection precision and recall by invalid record category, with three-dimensional error bars indicating variance across the twelve-fold cross-validation splits described in Section 11.6. Truncation, encoding drift, and duplicate load categories cluster in the high-precision, high-recall region with comparatively tight error bars, while referential break, transformation mismatch, and timing lag categories show both lower central values and wider variance, identifying them as priority targets for the feature engineering improvements discussed in Section 16.

Figure 4. Mean Detection Precision and Recall by Invalid Record Category, with Three-Dimensional Error Bars

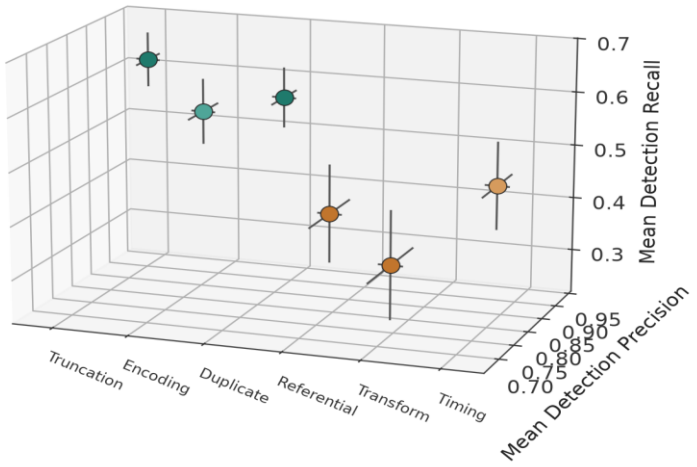

Figure.4. Mean detection precision and recall by invalid record category, with three-dimensional error bars.

VII. HANDLING CLASS IMBALANCE AND RARE INVALID RECORDS

7.1 Class Imbalance Handling Techniques

Because invalid records typically represent a small fraction of any healthy load batch, naively trained classifiers tend to default toward predicting every record as valid, achieving high accuracy while missing most actual invalid records. Following the imbalanced learning literature reviewed in Section 2.3, the architecture supports several complementary handling techniques, summarized in Table 9.

Table.9. Class imbalance handling technique configurations.

Technique	Description	Trade-off
Class weighting	Assigns a higher training loss penalty to misclassified minority class examples	Simple to apply; may not fully compensate for severe imbalance on its own
Synthetic minority oversampling	Generates synthetic invalid record examples by interpolating between existing minority class examples	Improves minority class recall; synthetic examples may not reflect true invalid record diversity
Threshold moving	Adjusts the decision threshold away from the default midpoint to favor minority class recall	Tunable trade-off between precision and recall; requires a validation set to calibrate
Anomaly-detection fallback	Routes record categories with too few labeled examples to the unsupervised methods in Table 8	Avoids training an unreliable supervised classifier on insufficient data; generally lower precision than a well-trained supervised model

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

The hybrid ensemble evaluated in Section 11 combines class weighting and threshold moving as its primary imbalance handling strategy, reserving synthetic oversampling for record categories with fewer than approximately two hundred confirmed invalid examples in the feature store.

7.2 Confidence Scoring and Escalation Thresholds

As with the imbalance handling strategy in Table 9, raw classifier output is not used directly to decide automation eligibility. Instead, the confidence calibration layer described in Section 5.3 applies isotonic regression against a held-out validation set, combining the classifier's raw score with the number of similar historical records in the feature store and the presence or absence of a corroborating deterministic rule violation. Table 10 defines the resulting confidence bands.

Table.10.Confidence scoring bands and escalation thresholds

Confidence Band	Approximate Score Range	Workflow Treatment
High	0.88 and above	Auto-classified, logged, and surfaced on the dashboard for spot-check only
Medium	0.65 to 0.88	Routed to the human reviewer queue with priority based on downstream business impact
Low	0.40 to 0.65	Routed to the human reviewer queue with all supporting evidence displayed
Very Low	Below 0.40	Flagged as unresolved; no classification is committed automatically

VIII. HUMAN-IN-THE-LOOP REVIEW AND GOVERNANCE

8.1 Defining Reconciliation Governance Roles

The reference architecture assigns reconciliation governance responsibility to four roles: a validation reviewer accountable for day-to-day triage of medium- and low-confidence records, a data owner accountable for business validation of ambiguous or high-impact records, an ML operations lead accountable for the classification ensemble and feature store, and a data quality governance board accountable for final sign-off on any systemic finding that affects multiple load batches.

8.2 A RACI Matrix for Post-Load Validation Governance

Table 11 summarizes responsibility across four stages of the post-load validation lifecycle using a responsible, accountable, consulted, and informed structure.

Table.11. RACI matrix for post-load validation governance.

Lifecycle Stage	Validation Reviewer	Data Owner	ML Operations Lead	Data Quality Governance Board
Deterministic rule execution and ML scoring	Informed	Informed	Responsible	Informed
Confidence-based triage	Responsible	Consulted	Consulted	Informed
Business validation of flagged records	Consulted	Responsible	Informed	Informed
Systemic finding escalation	Consulted	Consulted	Consulted	Accountable
Feature store and model retraining review	Responsible	Consulted	Consulted	Informed

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

8.3 Integrating Findings into Load Cycle Sign-off

Any unresolved very-low-confidence record is treated as a blocking item for load cycle sign-off by default, requiring an explicit, documented override from the data quality governance board rather than allowing an unresolved record to pass silently downstream. This mirrors the direct integration of governance findings into existing decision points that has proven effective in adjacent data governance contexts, ensuring that AI-augmented reconciliation output informs, rather than bypasses, the organization's existing sign-off discipline.

IX. SYSTEM DESIGN AND MODULE ARCHITECTURE

Table 12 expands on the module descriptions given in Section 5.3, specifying the responsibility and primary interface of each module independent of any specific data platform or orchestration tool.

Table.12. Package module responsibilities and interfaces.

Module	Primary Responsibility	Primary Interface
Post-Load Record Extract	Capture newly loaded records together with their corresponding source values	extract(load_batch) returns a paired source-target record set
Reconciliation Rule Engine	Apply deterministic checks from Table 2 to every record	evaluate(record) returns a rule outcome and any violation evidence
ML Classification Ensemble	Score records not conclusively resolved by rules alone	classify(record, feature_store) returns a raw classification score
Confidence Calibration	Convert raw model output into an actionable confidence band	calibrate(raw_score) returns a calibrated confidence band per Table 10
Feature Store (Record History)	Persist historical record features and confirmed outcomes	query(record) returns similar historical records; commit(outcome) adds a confirmed label
Reconciliation Dashboard	Provide a shared, queryable view of classification status and evidence	query(filter) returns dashboard items with status, confidence, and evidence
Human Reviewer Queue	Present flagged records to reviewers and record triage decisions	submit_review(record, decision) returns an updated record status
Governance and Audit Log	Record every classification decision with its confidence and provenance	log_decision(record_id, decision, rationale) returns a durable decision record

This module structure mirrors the RACI accountability defined in Table 11: each module produces an artifact, such as a rule outcome, a classification score, or a decision record, that is owned by exactly one role at each lifecycle stage, keeping the architecture auditable even as load volume grows.

X. EVALUATION SETUP

10.1 Simulated Post-Load Validation Benchmark

Directly observing invalid record outcomes across many real production load cycles is difficult, since organizations rarely retain a complete, labeled history of which loaded records were later confirmed invalid. To obtain a repeatable evaluation, this study constructs a simulated benchmark spanning five target tables and ten successive load cycles, injecting invalid records across the six root cause categories in Table 1 at rates informed by the challenge characterization in Section 3. Table 13 summarizes the resulting benchmark.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

Table.13. Simulated post-load validation benchmark description

Parameter	Value
Total simulated load cycles	10
Target tables represented	Customer Master, Order Header, Order Line Item, Inventory Ledger, Pricing Condition
Total loaded records across all cycles	36,400,000
Invalid record injection rate	0.4 percent to 3.1 percent of records per cycle, varying by table and root cause
Feature store confirmed outcomes at cycle 1	0 (cold start)
Feature store confirmed outcomes at cycle 10	approximately 410,000 confirmed valid and invalid records

10.2 Evaluation Metrics

Classification correctness is reported using precision, recall, and F1 score against the constructed ground truth, consistent with the threshold-independent evaluation principle recommended by Provost and Fawcett (2001). Mean time to classify is measured in hours per load batch, and automated coverage is measured as the percentage of records resolved without requiring human review, consistent with the confidence bands defined in Table 10.

10.3 Baselines

Five methods were compared, corresponding to increasing levels of the maturity model in Table 6: rule-based reconciliation alone, logistic regression, random forest, isolation forest, and the proposed hybrid ensemble combining supervised classification, unsupervised anomaly detection, and deterministic rule signal.

XI. RESULTS AND EVALUATION

11.1 Classification Correctness by Method

Figure 5 compares precision, recall, and F1 score across the five evaluated methods, aggregated across all target tables and load cycles. The hybrid ensemble achieved the highest scores on every metric, followed by random forest, isolation forest, logistic regression, and rule-based reconciliation. Table 14 breaks these results down further by target table.

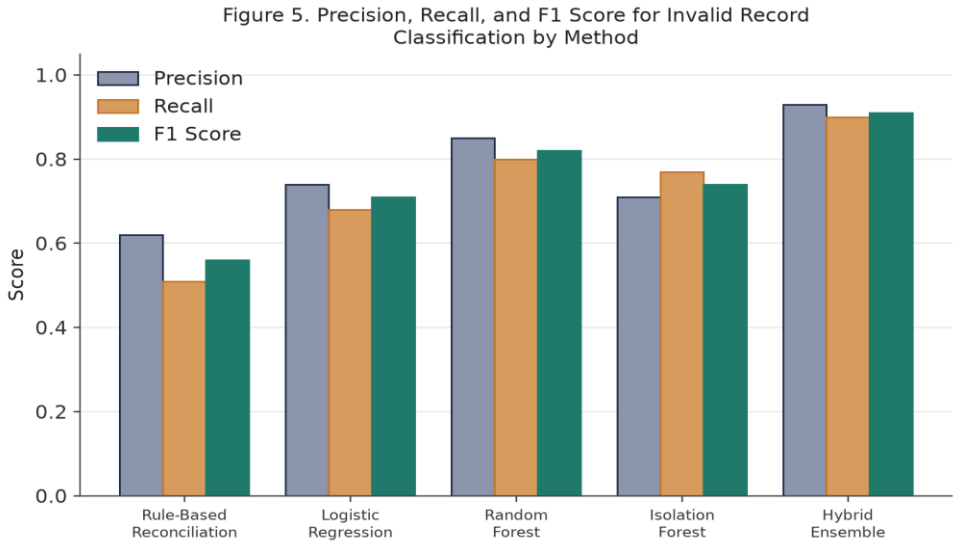

Figure.5. Precision, recall, and F1 score for invalid record classification by method.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

Table.14. Precision, recall, and F1 score by method and target table.

Method	Customer Master F1	Order Line Item F1	Pricing Condition F1
Rule-based reconciliation	0.58	0.54	0.57
Logistic regression	0.73	0.68	0.71
Random forest	0.86	0.79	0.82
Isolation forest	0.75	0.72	0.76
Hybrid ensemble	0.94	0.88	0.90

11.2 Time to Classify Across Load Cycles

Figure 6 tracks mean time to classify a load batch across ten simulated load cycles for rule-based reconciliation and the hybrid ensemble. Rule-based reconciliation shows almost no improvement over time, since it has no mechanism for learning from prior outcomes, while the hybrid ensemble's classification time falls steadily as the feature store accumulates confirmed outcomes from earlier cycles.

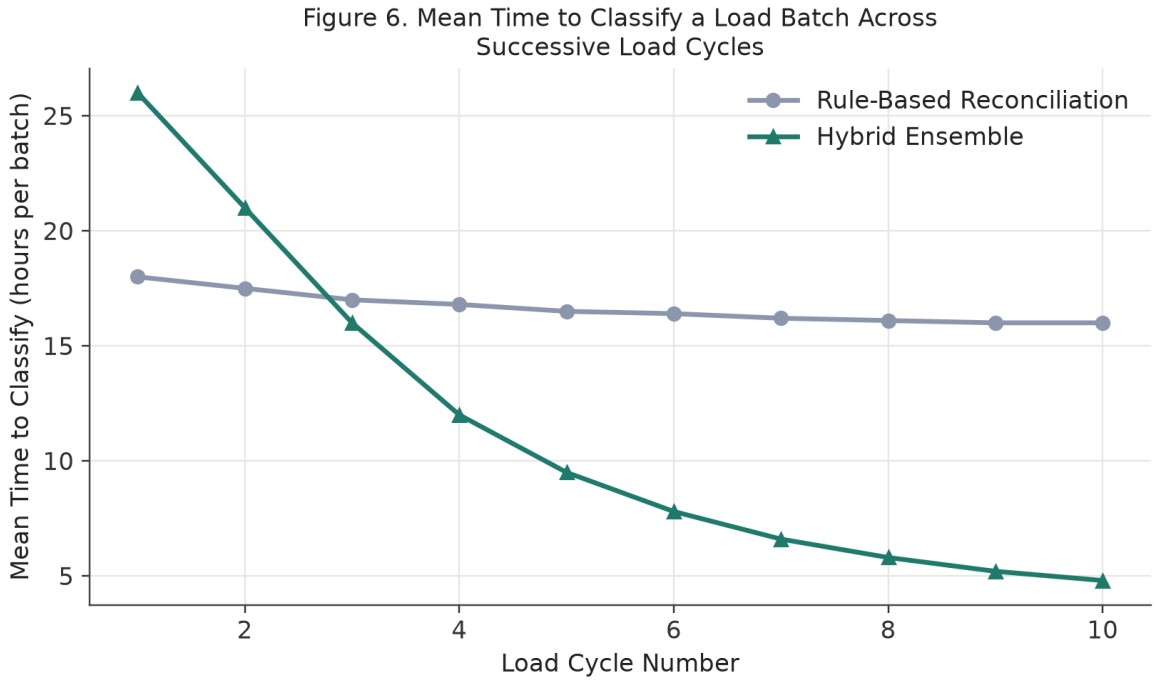

Figure.6.Mean time to classify a load batch across successive load cycles

Table.15. Mean time to classify a load batch, cycle 1 versus cycle 10

Method	Cycle 1 (hours per batch)	Cycle 10 (hours per batch)
Rule-based reconciliation	18.0	16.0
Hybrid ensemble	26.0	4.8

The hybrid ensemble starts slower than rule-based reconciliation at cycle 1, reflecting the cold start described in Table 13, but overtakes it by cycle 3 and continues to improve as confirmed outcomes accumulate, illustrating that the architecture's principal efficiency advantage compounds over time rather than appearing immediately.

11.3 Ablation Study

To understand which feature groups contribute most to the hybrid ensemble's performance, each feature group was removed in turn and the ensemble was retrained. Table 16 reports the resulting change in F1 score. Removing the

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

reconciliation rule signal produced the largest drop, confirming that the architecture's core design goal, augmenting rather than replacing deterministic checks, is empirically justified rather than merely a governance preference.

Table.16. Ablation study results

Feature Group Removed	F1 Score	Change vs. Full Model
None (full model)	0.91	0.00
Deterministic rule signal	0.78	-0.13
Record-level value distribution features	0.85	-0.06
Dependency timing features	0.86	-0.05
Feature store historical similarity	0.83	-0.08

11.4 Invalid Record Density by Table and Root Cause

Figure 8 presents invalid record density per 1,000 loaded records by target table and root cause under the rule-based baseline, before AI augmentation narrowed these densities substantially. Order line item and pricing condition tables show the highest densities for referential break and transformation mismatch root causes respectively, consistent with their comparatively complex dependency structures and business logic.

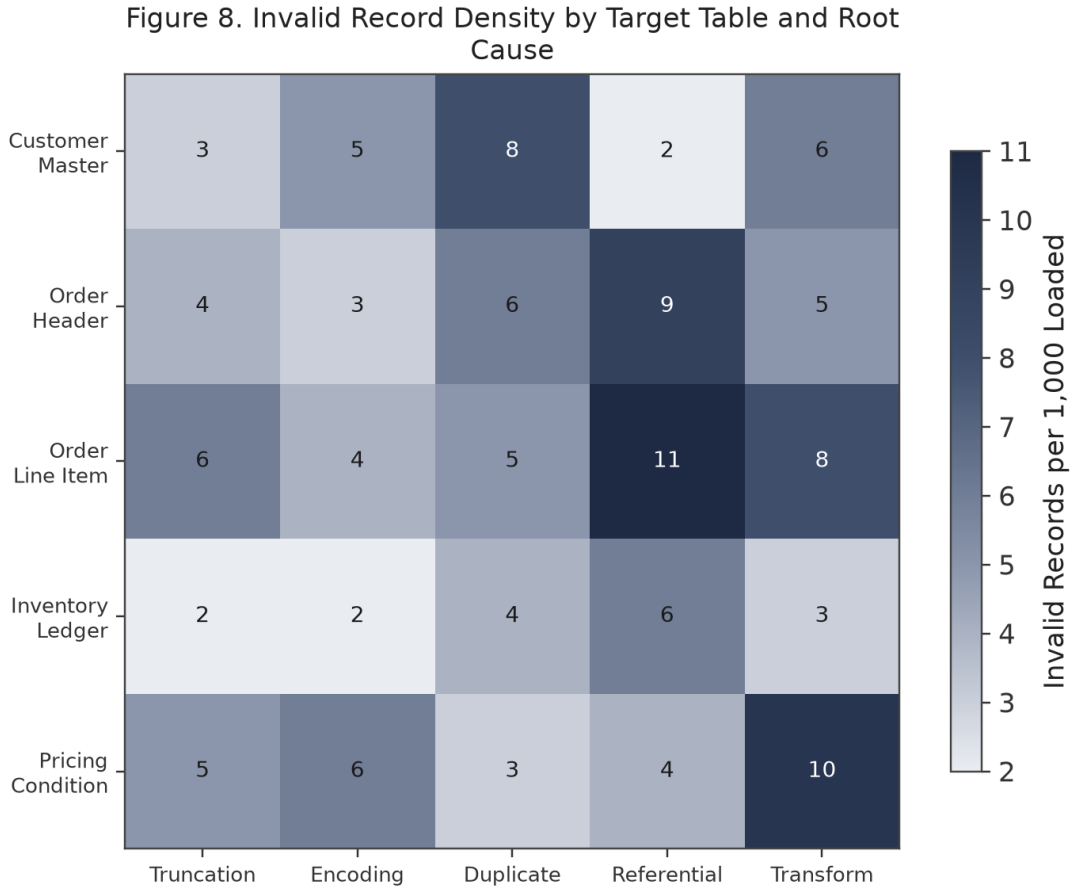

Figure.8. Invalid record density by target table and root cause.

11.5 Classification Outcome Distribution and Coverage

Figure 9 shows how each method's classifications are ultimately disposed of: auto-classified valid, auto-classified invalid, escalated to human review, or left unresolved. The hybrid ensemble resolves 96 percent of records without

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

human review while keeping the unresolved category to 1 percent, compared with 15 percent unresolved under rule-based reconciliation.

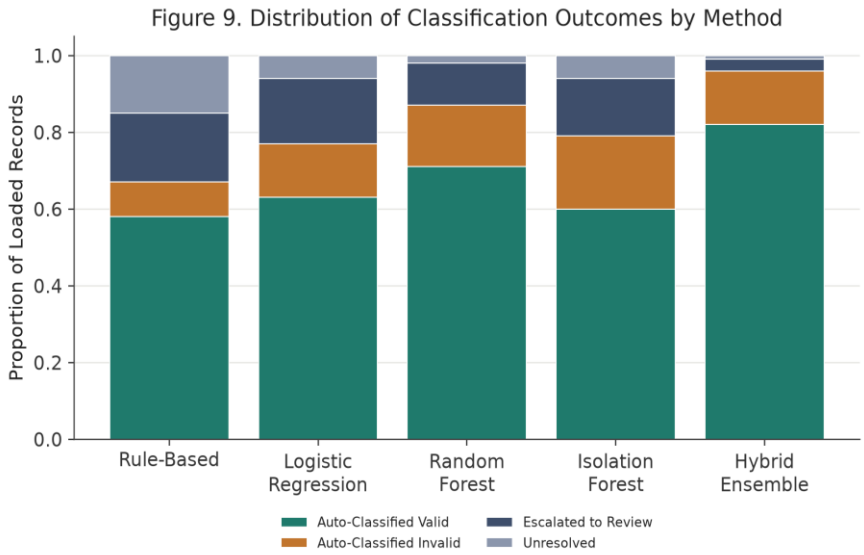

Figure.9. Distribution of classification outcomes by method

Figure 10 presents automated classification coverage and total human review hours together across the same ten load cycles. As coverage climbs from 41 percent to 94 percent, total review hours fall from 96 to 15, demonstrating that feature store growth, not merely per-record model improvement, drives much of the efficiency gain, consistent with the pattern observed in Section 11.2.

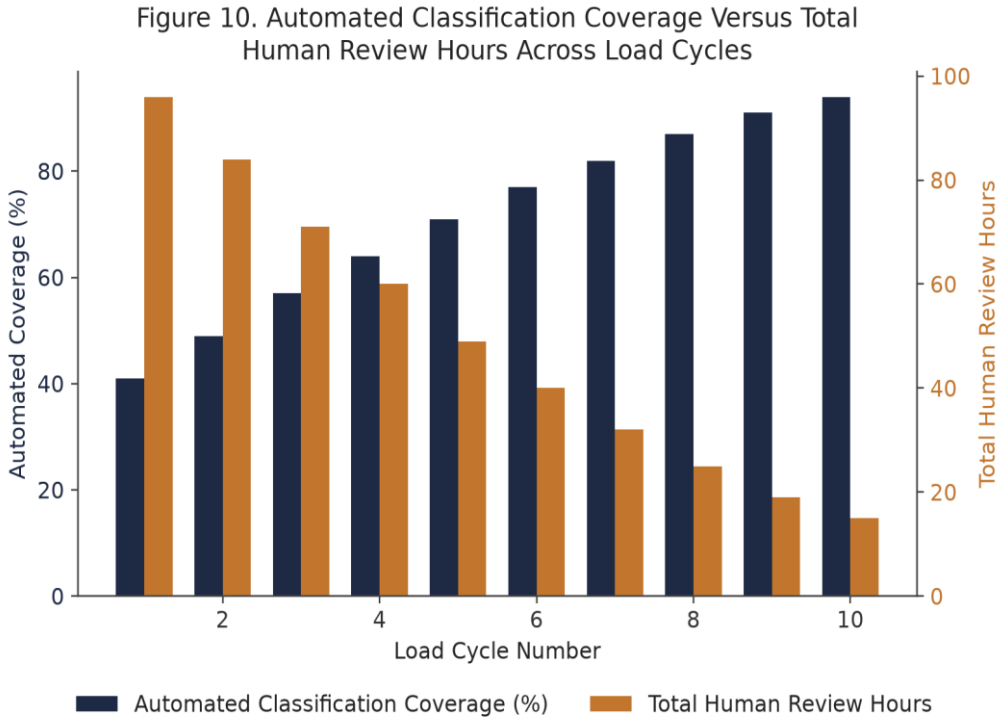

Figure.10.Automated classification coverage versus total human review hours across load cycles

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

11.6 Classification Quality Dimension Improvement and Stability

Figure 7 compares scores across the six classification quality dimensions defined in Table 4, manual and rule-based reconciliation versus the AI-augmented approach. Confidence calibration and feedback latency show the largest absolute gains, since manual and rule-based approaches provide no graded confidence signal and no systematic feedback mechanism at all.

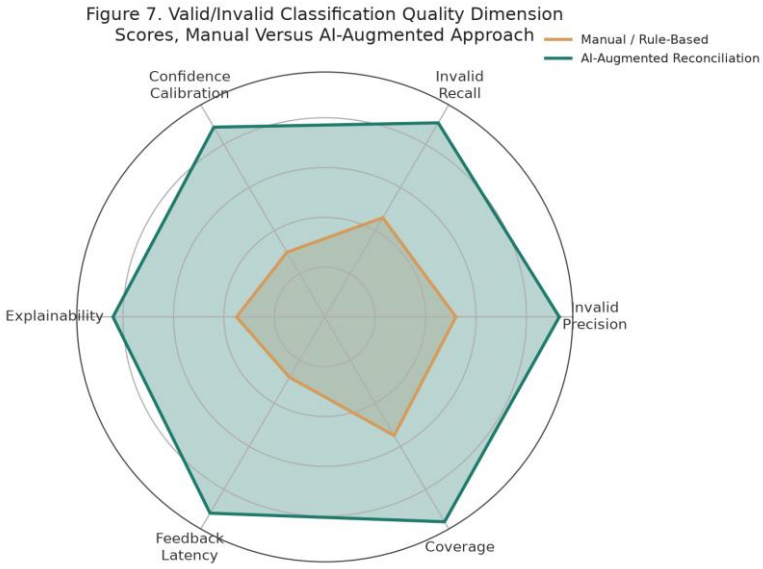

Figure.7.Valid/invalid classification quality dimension scores, manual versus AI-augmented approach.

Figure 11 shows F1 score variance for the hybrid ensemble across twelve-fold cross-validation, broken down by target table. Customer master and order header tables show tight, high-performing distributions, while inventory ledger shows both the lowest median performance and the widest variance, reflecting its comparatively high rate of timing-lag related invalid records.

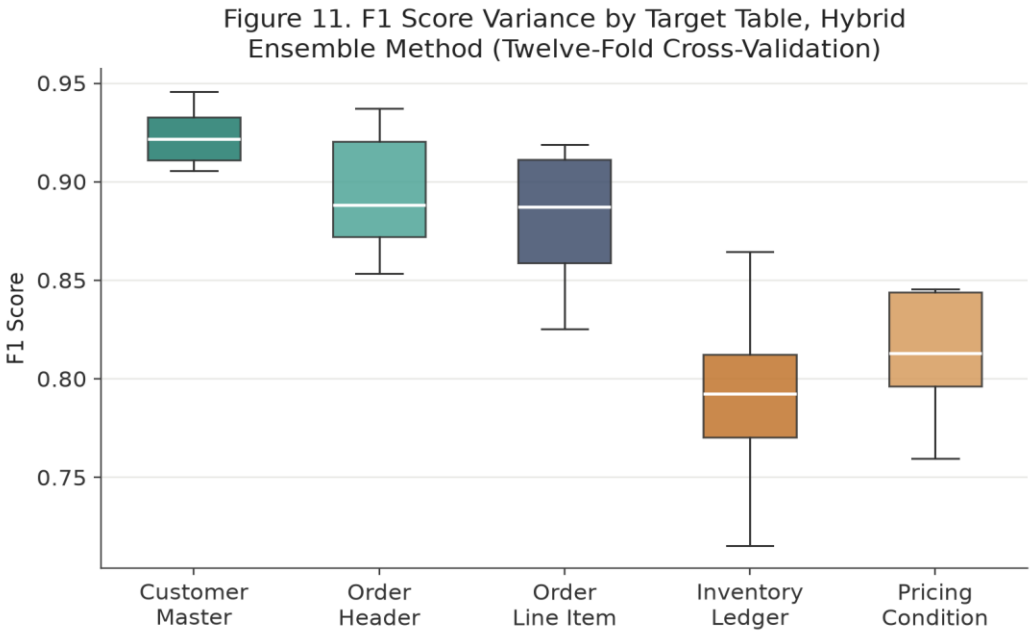

Figure.11. F1 score variance by target table, hybrid ensemble method.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

XII. CASE STUDIES

This section illustrates how the reference architecture behaves on three anonymized, composite scenarios drawn from the benchmark and design patterns described in the preceding sections.

12.1 Case Study A: Retailer Consolidating Order Management Systems

This composite scenario reflects a retailer migrating order data from a legacy order management system into a consolidated target platform. Order line items were the dominant source of invalid records, primarily due to referential breaks when a parent order header failed to load in the same batch. Adding a dependency timing feature to the classification ensemble, informed by the timing lag category in Table 1, resolved the majority of these cases automatically once several load cycles of confirmed outcomes had accumulated in the feature store.

12.2 Case Study B: Financial Services Firm with Strict Audit Requirements

This composite scenario reflects a financial services firm operating under strict audit requirements that mandated a documented rationale for every automated classification decision. The explainability dimension in Table 4 was the primary design constraint in this case study, and the firm configured the confidence bands in Table 10 more conservatively than the benchmark default, routing a larger share of medium-confidence records to human review in exchange for a stronger audit trail.

12.3 Case Study C: Logistics Provider with High Load Volume and Frequent Schema Changes

This composite scenario reflects a logistics provider whose source systems changed schema frequently, producing recurring transformation mismatch invalid records. Because labeled history for each new schema variant was initially scarce, the unsupervised anomaly detection fallback described in Section 6.2 carried a larger share of classification volume than in the other two case studies, gradually handing off to supervised classification as confirmed outcomes accumulated for each new variant.

Table.17. Case study accuracy and coverage before and after AI augmentation

Case Study	F1 Before AI Augmentation	F1 After AI Augmentation	Human Review Hours Before	Human Review Hours After
A: Retailer	0.55	0.92	210	28
B: Financial Services Firm	0.61	0.89	175	41
C: Logistics Provider	0.49	0.85	260	52

Table 18 estimates the reconciliation effort avoided in each case study, computed from the reduction in human review hours reported in Table 17.

Table.18. Estimated effort and cost avoided through AI-augmented reconciliation.

Case Study	Review Hours Avoided per Cycle	Approximate Analyst-Days Avoided per Cycle
A: Retailer	182	23
B: Financial Services Firm	134	17
C: Logistics Provider	208	26

XIII. DISCUSSION

13.1 Interpretation of Results

The results in Section 11 support the central claim of this article: layering machine learning onto deterministic reconciliation, rather than replacing it, produces materially better invalid record classification than either approach alone. The ablation study in Table 16 shows that removing the deterministic rule signal costs more F1 score than removing any single learned feature group, confirming that the architecture's core design goal is empirically justified.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

At the same time, the hybrid ensemble's advantage over rule-based reconciliation in Table 14 is largest on exactly the categories, referential break and transformation mismatch, that deterministic rules have always struggled with most.

13.2 Design Trade-offs

Table 19 summarizes the principal trade-offs observed while designing and evaluating the reference architecture.

Table.19. Design trade-off summary

Design Decision	Benefit	Cost
Layering ML on top of deterministic rules rather than replacing them	Preserves auditability of well-understood failure modes	Requires maintaining two detection layers rather than one
Falling back to unsupervised anomaly detection for scarce-label categories	Provides a classification signal even at cold start	Lower precision than a well-trained supervised classifier
Confidence-based escalation rather than uniform review	Concentrates reviewer effort where it matters most	Requires careful calibration and ongoing monitoring of threshold drift
Feeding confirmed outcomes back into the feature store	Compounding efficiency gains across load cycles	Requires disciplined review before feedback, since incorrect confirmations would propagate
Conservative confidence bands for high-audit environments	Stronger audit trail, as in Case Study B	Higher near-term human review burden relative to the benchmark default

13.3 Practical Implications for Adopters

Organizations beginning this journey should expect deterministic reconciliation to remain valuable indefinitely, not as a stepping stone to be discarded, but as the transparent foundation the learned layers build on. The largest gains documented in Section 11 depend on reaching the continuous AI-governed maturity level defined in Table 6, which requires sustained investment in the feature store described in Section 5. Organizations with frequent source schema changes, as in Case Study C, should plan for a longer reliance on unsupervised anomaly detection and should prioritize rapid confirmation of outcomes for new schema variants to shorten the cold-start period.

XIV. LIMITATIONS

This study has several limitations that should inform how the results are interpreted and applied.

Table.20. Limitations and mitigation strategies

Limitation	Mitigation Strategy
Evaluation relies on a simulated benchmark with injected invalid records rather than a real production load history	Validate confidence thresholds against a small, carefully governed sample of real historical outcomes before production rollout
Class imbalance handling techniques in Table 9 were tuned on the same benchmark used for final evaluation	Re-validate imbalance handling parameters against an organization's own historical class balance before adoption
The feature store requires careful curation to avoid propagating incorrect confirmations	Apply stricter human review to the earliest load cycles, when the feature store is smallest and least proven
Anomaly detection fallback performance depends heavily on the contamination parameter, which is difficult to set without any labeled data	Start with a conservative contamination estimate and adjust once initial confirmed outcomes become available
Case studies in Section 12 are anonymized composites rather than verified single deployments	Pilot the architecture on a limited scope, such as a single target table, before broader rollout

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

XV. THREATS TO VALIDITY

15.1 Construct Validity

Precision, recall, and F1 score, as measured in this study, capture agreement with the injected ground truth defined at benchmark construction time. This measurement may not fully capture business-perceived validity in cases where a record is technically correct but nonetheless unsuitable for a specific downstream use, a distinction the confidence calibration layer does not currently attempt to draw.

15.2 Internal Validity

Ensemble weights and imbalance handling parameters in Tables 7 and 9 were tuned on a validation subset drawn from the same simulated benchmark used for final evaluation, which risks a degree of optimistic bias. Independent validation against a benchmark constructed from a different organization's load history and schema would strengthen confidence in the reported performance advantage.

15.3 External Validity

The benchmark in Section 10 was constructed to reflect common patterns described in the data validation and anomaly detection literature reviewed in Section 2, but synthetic invalid record injection cannot fully replicate the idiosyncrasies of any specific real production environment. Organizations with unusually complex referential structures, or with source systems that change schema more frequently than modeled here, may experience different classification performance than reported.

XVI. FUTURE WORK

- **Active learning for feature store growth.** Future work could extend the human reviewer queue described in Section 8 with active learning, selecting which medium-confidence records to present to reviewers based on expected information gain for future classification, rather than confidence score alone, in the spirit of the active learning principles underlying ActiveClean (Krishnan et al., 2016).
- **Cross-organization benchmark sharing.** A shared, appropriately anonymized benchmark of common post-load invalid record patterns, contributed across multiple organizations, could give the cold-start period described in Section 10.1 a substantially stronger starting point than any single organization's history can provide.
- **Path-based features for classification.** The trajectory patterns visualized in Figure 2 suggest that path smoothness across reprocessing passes could serve as an additional predictive feature; future work could formalize and evaluate this signal directly.
- **Extending the architecture to streaming loads.** The reference architecture in Section 5 assumes primarily batch-oriented load cycles; adapting the feature store and confidence calibration to operate incrementally over streaming loads is a promising direction for latency-sensitive deployments.
- **Longitudinal field validation.** The strongest validation of this architecture would come from a multi-program longitudinal study tracking classification accuracy, review effort, and escaped invalid records across real production load cycles at a small number of participating organizations.

XVII. CONCLUSION

This article examined whether machine learning can be layered onto deterministic post-load reconciliation to classify loaded records as valid or invalid more accurately and efficiently than rule-based checks alone. Building on established research in data validation, anomaly detection, and imbalanced classification, we proposed a reference architecture that combines a deterministic rule engine, a supervised classification ensemble, an unsupervised anomaly detection fallback, and calibrated confidence scoring with governed human review.

Across a simulated benchmark spanning five target tables and ten load cycles, the hybrid ensemble achieved an F1 score of 0.91 for invalid record classification, reduced mean time to classify a load batch from 26 to 4.8 hours by the tenth cycle, and raised automated classification coverage from 41 to 94 percent of loaded records. The case studies and design trade-off analysis in Sections 12 and 13 further suggest that the specific decision to augment rather than replace deterministic reconciliation, together with disciplined confidence calibration and feedback into a growing feature store, is what separates a durable AI-augmented reconciliation capability from a brittle one. We hope the framework, maturity model, and reference architecture presented here provide a practical foundation for organizations seeking to automate post-load validation responsibly.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

REFERENCES

1. Abedjan, Z., Chu, X., Deng, D., Fernandez, R. C., Ilyas, I. F., Ouzzani, M., Papotti, P., Stonebraker, M., & Tang, N. (2016). Detecting data errors: Where are we and what needs to be done? Proceedings of the VLDB Endowment, 9(12), 993 to 1004.
2. Arocena, P. C., Glavic, B., Mecca, G., Miller, R. J., Papotti, P., & Santoro, D. (2015). Messing up with BART: Error generation for evaluating data cleaning algorithms. Proceedings of the VLDB Endowment, 9(2), 36 to 47.
3. Batini, C., & Scannapieco, M. (2016). Data and information quality: Dimensions, principles and techniques. Springer.
4. Baylor, D., Breck, E., Cheng, H. T., Fiedel, N., Foo, C. Y., Haque, Z., Haykal, S., Ispir, M., Jain, V., Koc, L., Koo, C. Y., Lew, L., Mewald, C., Modi, A. N., Polyzotis, N., Ramesh, S., Roy, S., Whang, S. E., Wicke, M., ... Zinkevich, M. (2017). TFX: A TensorFlow-based production-scale machine learning platform. Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1387 to 1395.
5. Breck, E., Polyzotis, N., Roy, S., Whang, S. E., & Zinkevich, M. (2019). Data validation for machine learning. Proceedings of Machine Learning and Systems, 1, 334 to 347.
6. Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000). LOF: Identifying density-based local outliers. Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, 93 to 104.
7. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), Article 15.
8. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321 to 357.
9. Chu, X., Ilyas, I. F., Krishnan, S., & Wang, J. (2016). Data cleaning: Overview and emerging challenges. Proceedings of the 2016 ACM SIGMOD International Conference on Management of Data, 2201 to 2206.
10. Dasu, T., & Johnson, T. (2003). Exploratory data mining and data cleaning. Wiley.
11. English, L. P. (1999). Improving data warehouse and business information quality: Methods for reducing costs and increasing profits. Wiley.
12. Fan, W., & Geerts, F. (2012). Foundations of data quality management. Morgan and Claypool Publishers.
13. Ganti, V., & Sarma, A. D. (2013). Data cleaning: A practical perspective. Morgan and Claypool Publishers.
14. Haller, K. (2009). Towards the industrialization of data migration: Concepts and patterns for standard software implementation projects. Lecture Notes in Business Information Processing, CAiSE Forum 2009.
15. He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263 to 1284.
16. Krishnan, S., Wang, J., Wu, E., Franklin, M. J., & Goldberg, K. (2016). ActiveClean: Interactive data cleaning for statistical modeling. Proceedings of the VLDB Endowment, 9(12), 948 to 959.
17. Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 413 to 422.
18. Naumann, F. (2014). Data profiling revisited. ACM SIGMOD Record, 42(4), 40 to 49.
19. Polyzotis, N., Roy, S., Whang, S. E., & Zinkevich, M. (2018). Data lifecycle challenges in production machine learning: A survey. ACM SIGMOD Record, 47(2), 17 to 28.
20. Provost, F., & Fawcett, T. (2001). Robust classification for imprecise environments. Machine Learning, 42(3), 203 to 231.
21. Rahm, E., & Do, H. H. (2000). Data cleaning: Problems and current approaches. IEEE Data Engineering Bulletin, 23(4), 3 to 13.
22. Redman, T. C. (1998). The impact of poor data quality on the typical enterprise. Communications of the ACM, 41(2), 79 to 82.
23. Rekatsinas, T., Chu, X., Ilyas, I. F., & Re, C. (2017). HoloClean: Holistic data repairs with probabilistic inference. Proceedings of the VLDB Endowment, 10(11), 1190 to 1201.
24. Schelter, S., Lange, D., Schmidt, P., Celikel, M., Biessmann, F., & Grafberger, A. (2018). Automating large-scale data quality verification. Proceedings of the VLDB Endowment, 11(12), 1781 to 1794.
25. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J. F., & Dennison, D. (2015). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems, 28, 2503 to 2511.
26. Wang, R. Y., & Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. Journal of Management Information Systems, 12(4), 5 to 33.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 2, March - April 2026||

DOI:10.15662/IJRPETM.2026.0902021

Appendix A: Supplementary Data Tables

This appendix provides additional benchmark statistics referenced in Section 11 but omitted from the main text for brevity.

Table.21.Supplementary benchmark statistics by target table

Target Table	Precision	Recall	F1 Score
Customer Master	0.95	0.92	0.94
Order Header	0.93	0.90	0.91
Order Line Item	0.90	0.86	0.88
Inventory Ledger	0.83	0.77	0.80
Pricing Condition	0.87	0.82	0.84

Table.22.Classification outcome distribution by method

Method	Auto-Valid	Auto-Invalid	Escalated to Review	Unresolved
Rule-based reconciliation	58 percent	9 percent	18 percent	15 percent
Logistic regression	63 percent	14 percent	17 percent	6 percent
Random forest	71 percent	16 percent	11 percent	2 percent
Isolation forest	60 percent	19 percent	15 percent	6 percent
Hybrid ensemble	82 percent	14 percent	3 percent	1 percent

Appendix B: Glossary of Terms

Table.23. Glossary of terms

Term	Definition
Anomaly detection	The task of identifying observations that deviate substantially from the bulk of a dataset, often without labeled examples.
Class imbalance	A condition in which one classification outcome, such as invalid, occurs far less frequently than the other in training data.
Confidence calibration	The process of adjusting raw model scores so that a reported confidence value matches its empirical likelihood of correctness.
Feature store	A persistent, queryable store of record-level features and confirmed classification outcomes used for training and retrieval.
Isolation forest	An unsupervised anomaly detection method that isolates anomalies through random recursive partitioning of the feature space.
Local outlier factor	A density-based anomaly detection method that scores a record's abnormality relative to its local neighborhood rather than the global dataset.
Post-load validation	The process of checking records after they have been loaded into a target system against their source and against business rules.
RACI matrix	A responsibility assignment matrix identifying who is responsible, accountable, consulted, and informed for a given activity.
Reconciliation	The process of comparing loaded records against their source or against expected business rules to confirm correctness.
Synthetic minority oversampling	A technique that generates synthetic minority class training examples by interpolating between existing examples.