International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

AI-Based Data Quality Monitoring and Anomaly Detection

A Framework for Automated Detection, Diagnosis, and Remediation of Data Quality Issues Using Machine Learning

Mr. Akash Arun Gavali ,

Mrs. Shubhangi Pratik Bomble (Assistant professor)

Department of Master of Computer Application,
JCEI's Jaihind Institute of Management and Research, Vadgaon Sahani
Email: gavaliakash1999@email.com,
gadeshubhangi9@gmail.com

Abstract - Poor data quality is a quiet but persistent threat to analytics, machine learning, and business decision-making, and most organizations still depend on manual rule checks and periodic audits to catch it. This paper proposes an AI-driven framework for ongoing data quality monitoring and anomaly detection, one that brings together statistical profiling, unsupervised machine learning, and deep learning to surface missing values, duplicate entries, schema drift, distributional shifts, and outliers close to real time. The architecture pulls data from both batch and streaming sources, builds profiles against learned baselines, and runs an ensemble of Isolation Forest, autoencoder, and clustering-based detectors to flag anomalies; flagged items are scored, ranked, and passed to human reviewers, whose feedback feeds back into the system to sharpen detection accuracy over time. The framework was tested on structured tabular datasets with deliberately injected anomalies and benchmarked against conventional rule-based methods and single-model baselines. The hybrid ensemble reached a precision of 0.91, recall of 0.89, and F1-score of 0.90 — well ahead of the statistical baseline (F1 = 0.59) and the individual single-model approaches (F1 = 0.76–0.86). These results suggest that blending several detection paradigms with a human-in-the-loop feedback step meaningfully improves both accuracy and the interpretability of the anomalies that get flagged. The paper closes by examining deployment considerations, current limitations, and avenues for future work, including explainable anomaly detection and self-healing data pipelines.

Key Words: data quality, anomaly detection, machine learning, data observability, autoencoder, Isolation Forest, data pipelines, artificial intelligence.

1. INTRODUCTION

Modern organizations lean heavily on data to power automated decisions — credit scoring, fraud detection, personalized recommendations, predictive maintenance, and more. How well these systems perform hinges on the quality of the data feeding them: incomplete records, duplicate entries, mislabeled fields, schema changes, and abrupt distributional shifts can quietly corrupt downstream models and reports long before anyone catches on. Conventional data quality assurance still leans on static, hand-written validation rules (for example, "column X must not be null" or "value must fall between 0 and 100"). These rules work well for constraints that are already known and well understood, but they cannot keep up with the volume, velocity, and variety of today's data pipelines, and they are inherently blind to novel or context-dependent anomalies that never break a predefined rule yet are still clearly abnormal.

Artificial intelligence — and machine learning specifically — offers a different route: rather than coding every conceivable failure mode by hand, a model can learn what "normal" data looks like from historical patterns and flag departures from that norm on its own. This move from rule-based to learning-based data quality monitoring parallels a wider shift already underway in software and infrastructure monitoring, where the goal of "observability" is to surface unknown-unknowns rather than simply check a list of pre-specified failure conditions.

This paper puts forward and evaluates an AI-based framework for data quality monitoring and anomaly detection that is designed to run continuously across both batch and streaming sources. It brings together several complementary detection techniques — statistical profiling, tree-based isolation methods, clustering, and deep autoencoders — inside one ensemble pipeline, and adds a human-in-the-loop feedback mechanism so data stewards can confirm or dismiss flagged anomalies, letting the models improve continuously.

1.1 Problem Statement

Even as investment in data infrastructure keeps growing, data quality problems remain among the top reasons analytics and AI initiatives fail. Manual, rule-based validation simply cannot scale to the size and complexity of today's data ecosystems, cannot catch anomaly types it has never seen before, and tends to either flood teams with false positives (alert fatigue) or miss real issues through false negatives (silent failures) whenever thresholds are not tuned well. What is needed instead is an adaptive, self-improving system that monitors data quality continuously, catches both familiar and novel anomalies, and explains its findings clearly enough for data engineers and analysts to act on.

1.2 Objectives

- Design a modular architecture for AI-based data quality monitoring that works across both batch and streaming data sources.
- Build and compare several anomaly detection techniques — statistical, tree-based, clustering-based, and deep learning-based — for structured data quality problems.
- Propose a hybrid ensemble detector that draws on the strengths of each individual technique to cut down false positive and false negative rates.
- Add a human-in-the-loop feedback mechanism so the system can learn from analyst corrections over time.

- Evaluate the proposed framework against traditional rule-based methods using standard classification metrics: precision, recall, and F1-score.

1.3 Contributions

This paper makes four main contributions: (1) a unified, modular reference architecture for AI-driven data quality monitoring that does not depend on any particular data source; (2) a comparative, empirical evaluation of statistical, machine learning, and deep learning anomaly detectors on structured data quality tasks; (3) a hybrid ensemble detection strategy that beats the individual models it draws from; and (4) a discussion of practical deployment concerns, covering alert design, explainability, and integration with existing data pipelines.

2. LITERATURE REVIEW

Data quality has a long history in the database and information systems community, usually framed around dimensions like accuracy, completeness, consistency, timeliness, validity, and uniqueness. Early work in this area leaned on deterministic rule engines and constraint checking — approaches that still work well for clearly defined business rules but are inherently limited to anomalies the rule writer thought to anticipate.

Anomaly detection as its own machine learning problem has been studied extensively outside the data quality field, most notably in fraud detection, network intrusion detection, and industrial fault detection. Classical statistical techniques such as z-score and interquartile range (IQR) thresholding assume a known or roughly Gaussian distribution, so they tend to perform poorly on skewed, multivariate, or context-dependent anomalies. Distance- and density-based approaches such as k-nearest neighbors and DBSCAN clustering flag outliers by local density instead, but they are sensitive to the choice of distance metric and parameters and scale poorly once data becomes high-dimensional or very large.

Tree-based isolation methods, Isolation Forest chief among them, take a different tack altogether: instead of explicitly modeling what normal behavior looks like, they isolate individual observations through random recursive partitioning, relying on the fact that anomalies are generally easier to isolate than normal points. Because this approach scales well to high-dimensional data and needs no distance metric, it has become a common baseline for tabular anomaly detection.

Deep learning methods — autoencoders and variants such as variational autoencoders and LSTM autoencoders — instead learn a compressed latent representation of normal data and treat high reconstruction error as the signal of an anomaly. These models can capture complex, non-linear relationships between features and tend to perform strongly on high-dimensional or sequential data, though that comes at the cost of needing larger training sets, more compute, and offering less interpretability than simpler methods.

More recently, the field of "data observability" has proposed monitoring data pipelines the way infrastructure engineers monitor software systems — treating freshness, volume, schema, distribution, and lineage as first-class signals and using statistical or learned baselines to catch drift automatically. This line of work motivates the architecture proposed here, which treats data quality monitoring as an ongoing, adaptive process rather than a one-off validation step. Table 1 lays out the strengths and limitations of the major technique families discussed above.

Table -1: Comparison of anomaly detection technique families relevant to data quality monitoring.

Technique	Strengths	Limitations
Statistical / rule-based (z-score, control charts)	Simple, interpretable, low compute cost	Assumes known distribution; misses multivariate and contextual anomalies
Isolation Forest	Scales well, no distance computation, handles high dimensions	Less effective on local/contextual anomalies; needs parameter tuning
Autoencoder / deep learning	Learns complex, non-linear patterns; effective on high-dimensional, mixed data	Requires large training data, less interpretable, higher compute cost
Clustering (DBSCAN, k-means)	Unsupervised, finds density-based outliers	Sensitive to parameter choice; struggles with varying density
LSTM / sequence models	Captures temporal dependencies in streaming data	Needs sequential structure; training and tuning overhead

3. METHODOLOGY

3.1 System Architecture

The proposed framework is organized into five modules. The first, a Data Ingestion Layer, connects to batch sources (databases, data warehouses, file stores) and streaming sources (message queues, event streams) via connectors. The second, a Profiling and Baseline Layer, computes statistical summaries — null rates, cardinality, value distributions, data types — and keeps rolling baselines for each field. The third, a Detection Layer, runs several anomaly detectors in parallel over the profiled data. The fourth, a Scoring and Ranking Layer, combines the detector outputs into a single anomaly score and prioritizes which alerts surface first. The fifth, a Feedback and Retraining Layer, logs analyst decisions — confirm, dismiss, mislabel — and uses them to periodically retrain the detectors, closing the loop between human judgment and model behavior.

3.2 Detection Techniques

Four complementary detectors were built and compared: a statistical baseline applying z-score and IQR thresholds per field; an Isolation Forest model trained on each record's full feature vector; a density-based clustering detector (DBSCAN) run on normalized numerical features; and a deep autoencoder trained to reconstruct normal records, with anomalies flagged whenever reconstruction error exceeds a dynamically calibrated threshold. A fifth detector, the hybrid ensemble, combines the normalized anomaly scores from the Isolation Forest, DBSCAN, and autoencoder models through a weighted voting scheme, with the weights tuned on a held-out validation set.

3.3 Dataset and Anomaly Injection

The framework was tested on structured tabular datasets representative of common enterprise data — customer records, transaction logs, and sensor readings. Since naturally occurring, fully labeled data quality anomalies are hard to come by, synthetic anomalies were injected into a held-out test split at roughly a 5% rate, spanning five categories: missing-value bursts, duplicate records, out-of-range numerical values, schema-inconsistent categorical values, and sudden distributional shifts (concept drift) affecting a subset of features. Injecting anomalies this way allows detection performance to be measured precisely against ground truth while keeping the statistical structure of the real data intact.

3.4 Evaluation Metrics

Detection performance was assessed using precision (the share of flagged records that were genuine anomalies), recall (the share of genuine anomalies that were successfully flagged), and F1-score (the harmonic mean of the two). These metrics were chosen because data quality monitoring is fundamentally a class-imbalanced problem, where plain accuracy would be misleading given how rare true anomalies are relative to normal records.

3.5 Experimental Setup

Every model was trained on an 80/20 train-test split, with the training portion further divided to reserve 15% for validation and threshold calibration. The Isolation Forest used 200 estimators and a contamination parameter set to match the expected anomaly rate. The autoencoder was built as a symmetric fully connected network with a bottleneck layer at 25% of the input dimensionality, trained for 100 epochs with early stopping based on validation reconstruction loss. The hybrid ensemble's weights were chosen via grid search on the validation set to maximize F1-score.

4. RESULTS

Table 2 lays out detection performance across all the methods tested. The statistical z-score baseline came in lowest (F1 = 0.59), reflecting its difficulty with contextual and multivariate anomalies. Isolation Forest did substantially better (F1 = 0.76) by picking up on multivariate structure without needing any distributional assumptions. The autoencoder turned in the strongest individual-model result (F1 = 0.86), particularly on schema-inconsistent and distributional-shift anomalies, which hinge on subtle, non-linear interactions between features. The proposed hybrid ensemble came out on top overall (precision = 0.91, recall = 0.89, F1 = 0.90), which suggests that combining detectors with different failure modes cuts down both false positives and false negatives compared with relying on any single model.

Table -2: Detection performance comparison across models on the held-out test set with injected anomalies.

Model	Precision	Recall	F1-score
Z-score baseline	0.61	0.58	0.59
Isolation Forest	0.79	0.74	0.76
Autoencoder (proposed)	0.88	0.85	0.86
Hybrid ensemble (proposed)	0.91	0.89	0.90

Looking at results by anomaly category, the ensemble model did best on duplicate records and out-of-range values (F1 above 0.93), where the signal is fairly clear-cut, and comparatively worse on subtle distributional-shift anomalies (F1 around 0.80), which call for catching gradual change rather than spotting one deviant record. During a two-week pilot, analyst feedback showed that roughly 12% of ensemble-flagged alerts were dismissed as false positives on first review; feeding that feedback into a single retraining cycle cut the false-positive rate on later alerts by about a quarter, which speaks to the value of the human-in-the-loop step.

5. DISCUSSION

These results back up the central idea that combining several methodologically distinct anomaly detectors produces more dependable data quality monitoring than any one technique alone, which lines up with what the broader anomaly detection literature has found about ensemble methods. The gap between the statistical baseline and the learning-based methods highlights just how limited static rules are at catching multivariate and contextual data quality issues that never trip any single predefined constraint.

That said, the framework's effectiveness rests on having a reasonably clean historical baseline to learn "normal" behavior from; in cold-start situations, or in pipelines going through legitimate structural change, the system may initially throw off more false positives until enough baseline data and analyst feedback build up. Interpretability is a further practical concern — while the autoencoder and ensemble models delivered the strongest raw numbers, their reconstruction-error-based scores are harder for non-technical stakeholders to interpret than a straightforward rule violation, which is why the scoring layer includes per-feature contribution scores and natural-language alert summaries. Finally, running this in production requires attention to computational cost, alert fatigue, and integration with existing data catalog and lineage tools — none of which show up in offline precision/recall numbers alone.

6. CONCLUSION AND FUTURE WORK

This paper presented an AI-based framework for continuous data quality monitoring and anomaly detection that combines statistical, tree-based, clustering, and deep learning techniques into a hybrid ensemble with human-in-the-loop feedback. Experiments on structured data with injected anomalies show the hybrid ensemble clearly outperforming both traditional rule-based validation and single-model machine learning approaches, reaching an F1-score of 0.90 against 0.59 for the statistical baseline. These findings support a broader move away from static, rule-based data quality checks and toward adaptive, learning-based monitoring systems.

Future work will push the framework further in several directions: bringing in explainable AI techniques (such as SHAP-based feature attribution) to make anomaly scores easier for non-technical data stewards to interpret; extending evaluation to real-world, naturally occurring anomalies across multiple industry datasets rather than relying solely on synthetic injection; exploring self-healing pipelines that can automatically propose or apply corrective actions (such as imputation or deduplication) for high-confidence anomalies; and investigating federated or privacy-preserving variants of the framework for organizations that cannot centralize sensitive data for model training.

REFERENCES

1. Battani, C., & Scannapieco, M. (2016). Data and Information Quality: Dimensions, Principles and Techniques. Springer.
2. Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000). LOF: Identifying density-based local outliers. Proceedings of the ACM SIGMOD International Conference on Management of Data, 93–104.
3. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), 1–58.
4. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 226–231.
5. Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. Proceedings of the Eighth IEEE International Conference on Data Mining, 413–422.
6. Malhotra, P., Ramakrishnan, A., Anand, G., Vig, L., Agarwal, P., & Shroff, G. (2016). LSTM-based encoder-decoder for multi-sensor anomaly detection. arXiv preprint arXiv:1607.00148.
7. Redman, T. C. (1998). The impact of poor data quality on the typical enterprise. Communications of the ACM, 41(2), 79–82.

8. Sakr, S., & Elgammal, A. (2016). Towards a comprehensive data analytics framework for smart healthcare services. Big Data Research, 4, 44–58.
9. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., ... & Dennison, D. (2015). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems, 28.
10. Wang, R. Y., & Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. Journal of Management Information Systems, 12(4), 5–33.
11. Zhou, C., & Paffenroth, R. C. (2017). Anomaly detection with robust deep autoencoders. Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 665–674.
12. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), 1–58.
13. Aggarwal, C. C. (2017). Outlier Analysis (2nd ed.). Springer, Cham.
14. 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, 93–104.
15. Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. Proceedings of the 2008 IEEE International Conference on Data Mining, 413–422.
16. Sculley, D., et al. (2015). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems (NeurIPS), 2503–2511.
17. Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. Informatica, 31(3), 249–268.
18. Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann.
19. Kieu, T., Yang, B., Guo, C., Jensen, C. S., & Rundensteiner, E. A. (2018). Anomaly detection in time series data: A survey. ACM Computing Surveys.
20. Pang, G., Shen, C., Cao, L., & van den Hengel, A. (2021). Deep learning for anomaly detection: A review. ACM Computing Surveys, 54(2), 1–38.