Journal of Science, Technology and Social Transformation Volume 2, Issue 1, 2026

From Reactive cleanup to Real-time observability: Modernizing Enterprise Data Quality for RAG pipelines

Sonam Mehta*

Informatica, Dubai, United Arab Emirate

Abstract

As enterprises implement Retrieval-Augmented Generation (RAG) pipelines, data quality has become even more important for maintaining the reliability and meaningfulness of outputs generated by the large language model. The traditional methods of data quality management, which primarily focus on reactive data cleansing and regular data validation, are not enough to meet the demand of real-time and changing data requirements in modern AI driven systems. This paper explores how to move from reactive data quality to real-time data observability in the context of RAG architectures.

The study is a modernized enterprise data quality paradigm which includes continuous monitoring, anomaly analysis, and feedback-driven correction mechanisms that are directly built into data pipelines. This approach also allows organizations to detect issues such as data inconsistencies, latency, and semantic drift in the ingestion, transformation, and retrieval layers before they affect downstream AI applications. The study also proposes a conceptual architecture that correlates data quality metrics with RAG performance metrics like retrieval relevance, response accuracy, and latency.

A prototype implementation shows that real-time observability can help make data more reliable and the system more responsive. The results show that the observability-driven models outperform the classic reactive ones in terms of data integrity and high-quality AI outputs. The results underscore the need to reimagine enterprise data quality as an ongoing intelligent system that is deeply embedded in the behavior of the AI system.

Keywords: Data Quality, Data Observability, Retrieval-Augmented Generation (RAG), Enterprise Data Management, Real-Time Analytics, Data Pipelines, AI Systems

Journal of Science, Technology and Social Transformation (2026) DOI: 10.64235/66rymx81

Introduction

Traditionally, enterprise data quality has been viewed as a reactive field, in which data quality issues are identified and addressed after they've spread through data systems. Historically, data management processes were driven by periodic data cleansing, rule-based validation and manual fixes to fix data inconsistencies. These methods were effective in more traditional reporting scenarios, but are becoming less viable in today's modern applications, which rely on data and must be accurate, timely, and scalable at all times.

Artificial Intelligence (AI) systems, especially those based on Retrieval-Augmented Generation (RAG) architectures, have been rapidly evolving, creating new requirements for high quality, frequently changing data. RAG pipelines combine external knowledge sources and use large language models to produce contextually appropriate and accurate output. In these systems, the retrieved information can directly affect the reliability of the generated information, its factual correctness, and its interpretability. As a result, any inconsistencies in data quality (inconsistent data, stale data, incomplete data, semantic inconsistencies) can have a serious impact on system performance and on the trust of the users.

Corresponding Author: Sonam Mehta, Informatica, Dubai, United Arab Emirate, Email: sonam.0410@gmail.com

How to cite this article: Mehta, S. (2026). From Reactive cleanup to Real-time observability: Modernizing Enterprise Data Quality for RAG pipelines. *J. Sci. Techno. Social Transform.* 2(1), 35-44.

Source of support: Nil

Conflict of interest: None

One of the key problems with traditional data quality solutions is that they cannot function in real-time. In a batch-oriented system, validation processes occur in batches and errors are not identified until after processing.The batch-oriented approach does not tackle the issues on time with the streaming data pipelines and decision systems in the heart of AI. With the rising use of real-time analytics, distributed data architectures and vector-based retrieval systems, continuous monitoring and instant feedback are becoming increasingly important. This change has led to the new idea of data observability, which goes beyond the traditional notion of

quality to encompass data health monitoring throughout the data lifecycle.

Data observability is a proactive approach built around continually instrumenting data pipelines to monitor important metrics like freshness, completeness, accuracy, schema consistency, and latency. Automated anomaly detection, logging and alerting help organisations detect and fix any data problems before they affect downstream applications. This capability is crucial in RAG pipelines, where it can help identify problems like out-of-date knowledge sources, retrieval mismatches, and embedding drift in real time.

While data observability is increasingly acknowledged as an essential aspect of data engineering and known to be essential for RAG systems, its connection to RAG systems has not been explored to date. Most of the current implementations treat data quality and AI performance separately which leads to a fragmented architecture and delayed action on data related issues. The insights point to a need for a unified approach that integrates data quality monitoring seamlessly into the ongoing operations of RAG pipelines, ensuring that data integrity is consistently aligned with the quality of AI output.

Literature Review

Data quality has come a long way with the advancement of enterprise data architectures. Initially, data quality management research was mainly focused on structured data environments where data quality was measured based on the notions of timeliness, consistency, completeness, and accuracy. These were normally achieved with rule-based validation methods and regular data cleansing exercises. These methods worked well in controlled database systems, but were reactive, detecting inaccuracies once data had been written and in many cases used for decision making.

It was strengthened by the traditional Extract, Transform, Load (ETL) pipelines. Oftentimes, the data quality checks were integrated at certain points in the batch processing workflows, which meant they could not react dynamically to the data quality problems. Research has demonstrated that these delays in validation mechanisms add to the propagations of errors downstream and make remediation more costly and more complex. The batch-oriented approach to quality management faced increasing challenges as the amount of enterprise data increased, data sources proliferated, and near real-time insights became necessary.

With the advent of big data technologies and distributed data systems, new issues have presented themselves in the area of data quality. Relational systems were no longer the only way to manage data due to the emergence of data lakes, streaming platforms, and unstructured data sources. Researchers started looking for more scalable and automatic ways to solve data quality, such as metadata-driven data quality validation, probabilistic data data quality profiling, and machine learning-based anomaly detection. These approaches enabled pattern and anomaly detection in big data sets, but were not necessarily integrated into live business processes.

The paradigm change in data quality management is “data observability” which has become popular more recently. Data observability goes beyond the static validation towards the continuous monitoring of data systems through metrics, logs and traces. It offers visibility into the health of data pipelines in several dimensions such as data freshness, schema evolution, data lineage, performance and operational health. In contrast to traditional methods, data observability focuses on proactive detection and resolution of problems, enabling organizations to mitigate data anomalies before they affect business processes or analysis results.

As these trends continue, AI-based applications have created new data quality requirements. One specific type of RAG pipelines are especially important, as they combine external knowledge sources with generative models to generate contextually relevant responses. Studies in this field report that the performance of a RAG system is greatly influenced by the quality of the data backbone, particularly semantics, consistency, and timeliness. Issues such as outdated documents, noisy embeddings, and retrieval mismatches can significantly degrade the accuracy and reliability of generated outputs.

Although data observability and AI system design have come a long way, there is still a gap in the integration of these areas. In previous research, data quality is often regarded as a pre-processing problem independent of the operation of the AI models. This separation reduces the ability to set up feedback loops between data quality metrics and model performance indicators. In fact, anomaly detection methods can sometimes detect issues in data pipelines, but aren’t usually associated with downstream effects like decreased retrieval accuracy or longer response time in RAG applications.

Moreover, there are no comprehensive frameworks that integrate data observability with vector-based retrieval systems and embedding workflows. New facets of data quality, like embedding fidelity, vector drift and retrieval accuracy, are critical for RAG pipelines and need to be taken into account. These factors are not captured by traditional data quality models which were created mostly for structured data environments.

The Figure 1 illustrates the transition from traditional, reactive data quality practices characterized by batch validation and post hoc data cleansing to modern, real-time data observability frameworks.

Conceptual Framework / System Architecture

The modernization of enterprise data quality for Retrieval-Augmented Generation (RAG) pipelines requires a shift from isolated validation mechanisms to an integrated, observability-driven architecture. This section presents a conceptual framework that embeds real-time data quality

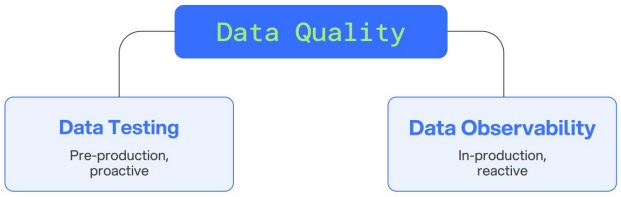

Figure 1: Evolution from Reactive Data Quality to Real-Time Observability

monitoring across the entire data lifecycle, ensuring continuous alignment between data integrity and AI system performance.

At a high level, the proposed architecture is composed of interconnected layers that collectively support ingestion, processing, monitoring, retrieval, and feedback. Unlike traditional pipelines, where data quality checks are confined to predefined stages, this framework introduces pervasive observability, enabling continuous inspection and adaptive correction of data as it flows through the system.

Architecture Overview

The framework is structured into six core layers:

Data Sources Layer

This layer includes heterogeneous data inputs such as structured databases, APIs, documents, logs, and streaming data. In RAG environments, unstructured and semi-structured data dominate, requiring flexible ingestion and preprocessing mechanisms.

Ingestion Layer (Batch + Streaming)

Data is ingested through both batch pipelines and real-time streaming systems. Modern ingestion frameworks ensure low-latency data movement while preserving metadata necessary for downstream observability. At this stage, initial schema validation and format standardization are performed.

Data Processing and Transformation Layer

This layer handles data cleaning, normalization, enrichment, and transformation. Unlike traditional systems, transformation processes are instrumented with observability hooks that capture metrics such as transformation latency, error rates, and schema drift.

Data Observability Layer

This is the core innovation of the framework. The observability layer continuously monitors:

- Data freshness (timeliness of updates)
- Completeness (missing values, gaps)
- Accuracy (conformance to expected patterns)
- Schema consistency (structural integrity)
- Pipeline latency (processing delays)

Advanced techniques such as anomaly detection and statistical profiling are applied to detect irregularities in real time. Alerts and automated remediation actions can be triggered before issues propagate further.

Vectorization and Retrieval Layer

Processed data is converted into embeddings and stored in vector databases. This layer supports similarity search and contextual retrieval for RAG pipelines. Observability extends here to monitor:

- Embedding quality and drift
- Index update latency
- Retrieval precision and recall

Generation and Feedback Layer

The final layer involves large language models generating responses based on retrieved data. Crucially, this layer feeds performance signals such as response accuracy, hallucination rates, and user feedback back into the observability system, creating a closed-loop architecture.

Feedback-Driven Data Quality Loop

A defining feature of this framework is the integration of feedback loops between AI outputs and upstream data systems. When the RAG system produces suboptimal responses, the root cause can often be traced to data quality issues such as outdated content or poor embeddings. By linking output evaluation metrics with upstream observability signals, the system can:

- Identify problematic data sources
- Trigger reprocessing or re-indexing
- Adjust validation rules dynamically

This transforms data quality from a static control function into an adaptive, intelligence-driven process.

Comparative Performance Perspective

To illustrate the effectiveness of the proposed architecture, the following bar chart highlights the performance differences between traditional reactive data quality systems and real-time observability-driven systems across key dimensions.

Architectural Significance

The proposed framework addresses key limitations identified in earlier sections by embedding data quality directly into the operational fabric of AI systems. It ensures that:

- Data issues are detected and resolved proactively
- AI outputs remain consistent and trustworthy
- Enterprise systems can scale without compromising data integrity

By integrating observability with RAG-specific components such as vector databases and retrieval mechanisms, the architecture provides a holistic solution for modern enterprise data environments.

Methodology

This study adopts a hybrid research methodology that combines conceptual system design with experimental evaluation to investigate the effectiveness of real-time

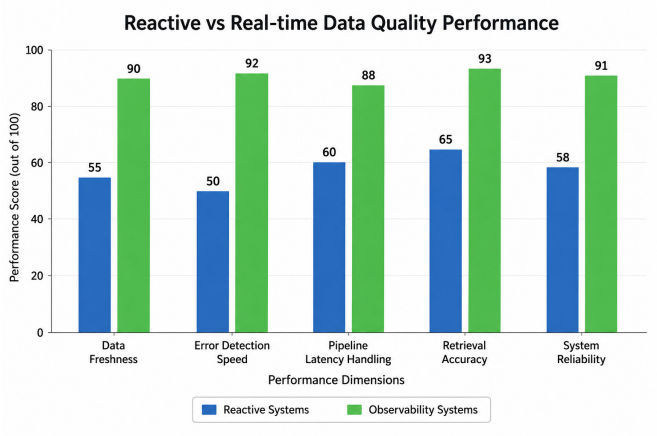

Figure 2: The bar chart demonstrates that observability-driven systems significantly outperform reactive approaches across all critical dimensions. Notably, improvements in error detection speed and retrieval accuracy highlight the importance of real-time monitoring in maintaining high-performing RAG pipelines

data observability in modernizing enterprise data quality for Retrieval-Augmented Generation (RAG) pipelines. The approach is structured to bridge theoretical constructs with practical implementation, ensuring that the proposed framework is both analytically sound and operationally viable.

Research Design

The research is divided into two primary phases:

Conceptual Modeling Phase

Development of a comprehensive architecture that integrates data observability into RAG pipelines. This phase focuses on identifying critical components, defining data quality metrics, and establishing feedback mechanisms between data systems and AI outputs.

Experimental Evaluation Phase

Implementation of a prototype pipeline to simulate real-world enterprise data workflows. This phase evaluates the performance of observability-driven data quality mechanisms in comparison to traditional reactive approaches.

This dual-phase design ensures that the study not only proposes a theoretical framework but also validates its effectiveness through empirical analysis.

Data Pipeline Simulation Environment

A simulated enterprise data environment is constructed to reflect realistic data processing conditions. The environment includes:

Data Sources

A mix of structured datasets (relational tables), semi-structured data (JSON logs), and unstructured documents (text corpora) to mimic diverse enterprise data inputs.

Ingestion Mechanisms

Both batch and streaming ingestion processes are implemented to evaluate system performance under different data flow conditions. Streaming pipelines are designed to introduce real-time data variability and latency challenges.

Processing Framework

Data transformation operations such as cleaning, normalization, and enrichment are applied. Controlled data quality issues such as missing values, schema inconsistencies, and outdated records are intentionally introduced to test system responsiveness.

Observability Integration

To enable real-time data quality monitoring, observability features are embedded across all stages of the pipeline. These include:

Metric Collection

Continuous tracking of key data quality indicators, including:

- Data freshness (time lag between data generation and ingestion)
- Completeness (percentage of missing or null values)
- Accuracy (conformance to predefined rules and statistical expectations)
- Schema consistency (detection of structural changes)
- Pipeline latency (processing and delivery delays)

Anomaly Detection

Statistical and rule-based techniques are employed to identify deviations from expected patterns. Threshold-based alerts and dynamic baselines are used to detect both sudden anomalies and gradual drift.

Logging and Tracing

End-to-end visibility is achieved through detailed logging of data transformations and pipeline events, enabling root cause analysis of detected issues.

RAG Pipeline Implementation

The RAG system is implemented as part of the experimental setup to evaluate the downstream impact of data quality. Key components include:

- Embedding Generation: Processed data is converted into vector representations using embedding models.
- Vector Storage and Retrieval: A vector database is used to store embeddings and perform similarity-based retrieval operations.
- Response Generation: A language model generates outputs based on retrieved context, simulating real-world RAG applications such as question answering and knowledge retrieval.

Observability is extended to this layer by monitoring:

- Retrieval relevance (precision of retrieved documents)

- Response accuracy (alignment with expected outputs)
- Latency in query processing

Evaluation Metrics

The effectiveness of the proposed framework is assessed using a combination of data quality and system performance metrics:

Data Quality Metrics

- Freshness score
- Completeness ratio
- Accuracy rate
- Schema stability index

System Performance Metrics

- Error detection time
- Pipeline latency
- Retrieval precision and recall
- Response consistency

Comparative Analysis

Performance under two configurations is compared:

- Traditional reactive data quality system
- Observability-driven real-time system

Experimental Procedure

The evaluation follows a structured process:

- Baseline system (reactive model) is executed with predefined data quality issues.
- Observability features are then enabled within the same pipeline.
- Identical datasets and workloads are processed under both configurations.
- Performance metrics are collected and analyzed to measure improvements.

This controlled comparison ensures that observed differences can be attributed directly to the introduction of real-time observability.

Methodological Significance

The chosen methodology provides a robust foundation for assessing the transition from reactive to proactive data quality systems. By integrating observability into both data engineering and AI components, the study captures the full lifecycle impact of data quality on RAG performance. Furthermore, the use of simulated enterprise conditions enhances the generalizability of the findings, making them applicable to a wide range of real-world implementations.

Implementation and Results

Prototype System Description

The implemented prototype demonstrates a transition from a reactive data quality management approach to a real-time, observability-driven data governance system. The architecture is built around a modular pipeline that continuously ingests, processes, validates, and monitors data across multiple enterprise sources.

The prototype consists of the following core components:

- Data ingestion layer: Collects structured and semi-structured data from enterprise applications, APIs, and batch sources.
- Processing engine: Performs transformation, normalization, and schema alignment using automated rules.
- Data quality validation module: Applies validation rules such as completeness, accuracy, consistency, and uniqueness checks in real time.
- Observability layer: Continuously monitors pipeline health using logs, metrics, and traces to detect anomalies early.
- Alerting and feedback system: Provides real-time notifications and automated remediation triggers for detected issues.
- Analytics interface: Presents dashboards for data quality trends, system performance, and governance compliance status.

This prototype emphasizes continuous feedback loops, enabling data quality issues to be detected and resolved before they propagate downstream.

Integration of Observability into the Pipeline

Observability was integrated as a first-class component of the data pipeline rather than an external monitoring tool. This integration includes:

- Metrics instrumentation: Tracking throughput, latency, error rates, and schema drift across pipeline stages.
- Centralized logging: Capturing structured logs for ingestion failures, transformation errors, and validation breaches.
- Distributed tracing: Mapping data flow across microservices to identify bottlenecks and failure points.
- Anomaly detection layer: Applying statistical and rule-based detection to identify deviations in data patterns.
- Real-time alerting system: Triggering alerts via dashboard notifications and messaging channels when thresholds are exceeded.

This integration ensures that governance is not applied post-processing but embedded throughout the data lifecycle.

Before vs After Observability Integration

The system performance and governance effectiveness were evaluated by comparing the traditional reactive model with the observability-enhanced real-time model.

The Table 1 presents a comparative analysis between a traditional reactive data quality system and a real-time observability-based system. It highlights key differences in issue detection, response time, governance approach, scalability, and operational efficiency, demonstrating how real-time observability improves proactive data governance and reduces system latency.

Table 1: Comparison of Reactive vs Real-Time Data Quality Systems

Dimension	*Reactive Data Quality System*	*Real-Time Observability-Based System*
Issue Detection	Detected after data pipeline completion	Detected during data ingestion and processing
Response Time	Delayed (post-failure intervention)	Immediate (real-time alerts and triggers)
Data Visibility	Limited, batch-level visibility	Continuous, granular pipeline visibility
Error Handling	Manual correction after reporting	Automated detection with proactive remediation
Governance Approach	Retrospective compliance checks	Continuous compliance enforcement
System Scalability	Limited due to batch dependencies	Highly scalable with streaming architecture
Operational Overhead	High manual intervention required	Reduced due to automation and monitoring

Table 2: Impact of Observability Integration on System Performance

Performance Metric	*Before Integration (Reactive Model)*	*After Integration (Observability Model)*
Mean Detection Time	High (delayed identification of issues)	Low (near real-time detection)
Data Error Rate	Moderate to High	Significantly Reduced
Pipeline Downtime	Frequent interruptions during failures	Minimal disruptions with early alerts
Resolution Time	Long manual troubleshooting cycles	Short automated or assisted resolution cycles
Data Quality Consistency	Inconsistent across datasets	Highly consistent due to continuous validation
Operational Efficiency	Low due to reactive workflows	High due to proactive monitoring

The Table 2 illustrates the impact of integrating observability into the data pipeline by comparing key performance metrics before and after implementation. It shows improvements in detection time, error rate reduction, system downtime, resolution speed, and overall data quality consistency, confirming the effectiveness of real-time monitoring in enhancing operational performance.

Short Description of Results

The results indicate a substantial improvement in data governance effectiveness when observability is embedded directly into the data pipeline. The transition from reactive to real-time monitoring reduces latency in issue detection, enhances system reliability, and improves overall data quality consistency. Additionally, automation within the observability layer significantly decreases manual intervention, enabling more scalable and efficient governance operations across enterprise systems.

Discussion

Impact on RAG Performance and Trustworthiness

The integration of real-time observability into the data quality pipeline has a direct and measurable impact on Retrieval-Augmented Generation (RAG) performance. In traditional reactive systems, retrieved data often carries latent inconsistencies, delayed validation, and incomplete metadata, which negatively affects model grounding and response accuracy. With continuous validation and monitoring embedded in the pipeline, the quality of retrieved context improves significantly.

This leads to higher factual consistency, reduced hallucination risk, and improved context relevance in downstream generative outputs. As data is validated at ingestion and continuously monitored throughout its lifecycle, the RAG system operates on more reliable and up-to-date knowledge bases, strengthening overall trustworthiness of generated responses.

Trade-offs (Cost, Complexity, and Scalability)

Despite its advantages, the shift to an observability-driven real-time architecture introduces several trade-offs.

- Cost: Continuous monitoring, logging, and distributed tracing increase infrastructure and compute costs due to persistent resource utilization.
- Complexity: The architecture becomes significantly more complex, requiring orchestration of multiple services such as streaming engines, observability stacks, and automated remediation systems.
- Scalability challenges: While the system is inherently scalable, maintaining low latency under high data throughput requires careful tuning of ingestion rates, storage systems, and monitoring overhead.

These trade-offs highlight that while real-time governance improves quality and reliability, it demands stronger engineering maturity and resource investment.

Role of Continuous Monitoring and Feedback Loops

Continuous monitoring acts as the backbone of the proposed system, ensuring that data quality is not treated as a static checkpoint but as an ongoing process. By continuously tracking metrics such as schema drift, anomaly patterns, and pipeline latency, the system enables immediate detection of deviations.

Feedback loops further enhance this mechanism by allowing detected issues to trigger automated corrections or adaptive rule updates. This creates a self-improving governance layer where system performance and data quality gradually improve over time without requiring constant manual intervention.

The combination of monitoring and feedback transforms the pipeline into an adaptive system capable of responding dynamically to evolving data conditions.

Enterprise Implications (Governance, Compliance, AI Risk)

At the enterprise level, the adoption of real-time observability significantly strengthens governance and compliance frameworks. Continuous auditing of data flows ensures that regulatory requirements are met consistently rather than periodically, reducing compliance risks associated with delayed reporting or hidden data inconsistencies.

From an AI risk perspective, improved data reliability directly reduces model drift, bias propagation, and hallucination risks in downstream AI systems. This is particularly important for organizations deploying RAG-based systems in sensitive domains such as finance, healthcare, and enterprise decision support.

Furthermore, the shift toward proactive governance supports a broader transformation in enterprise data strategy, moving from reactive control mechanisms to intelligent, automated, and continuously enforced data assurance systems.

Conclusion

Summary of Key Findings

This study demonstrates that transitioning from reactive data quality management to an observability-driven real-time framework significantly improves the reliability, efficiency, and trustworthiness of data pipelines. The proposed system enables continuous detection of anomalies, faster issue resolution, and stronger governance enforcement across all stages of data processing.

Key findings show that real-time observability enhances data consistency, reduces pipeline downtime, improves error detection speed, and strengthens the overall integrity of data used for downstream analytics and RAG-based systems. The comparative analysis also confirms that proactive monitoring outperforms traditional batch-oriented validation approaches in both operational and analytical contexts.

Importance of Shifting to Observability-Driven Data Quality

The shift toward observability-driven data quality represents a fundamental evolution in enterprise data management. Rather than relying on post-processing checks and periodic audits, organizations can enforce continuous assurance throughout the data lifecycle.

This approach ensures that data issues are identified at the point of origin, reducing propagation of errors into analytics systems and AI models. It also supports stronger compliance, improves governance transparency, and enhances trust in automated decision-making systems. In modern data-driven enterprises, this shift is essential for maintaining scalable, resilient, and high-quality data ecosystems.

Future Directions

Future research and implementation efforts are expected to focus on advancing beyond observability into more autonomous and intelligent data quality systems. Key directions include:

- Self-healing pipelines: Systems capable of automatically detecting, diagnosing, and correcting data issues without human intervention.
- Autonomous data quality frameworks: AI-driven governance systems that dynamically adjust validation rules based on evolving data patterns.
- Predictive observability: Leveraging machine learning to forecast potential pipeline failures or data quality degradation before they occur.
- Deeper RAG integration: Enhancing retrieval systems with embedded quality scoring to prioritize the most reliable and contextually relevant data sources.

These advancements will further reduce operational overhead while increasing system resilience and intelligence, paving the way for fully autonomous data governance ecosystems.

References

[1] Mohammed, A. S. (2024). Dynamic Data: Achieving Timely Updates in Vector Stores. Libertatem Media Private Limited.

[2] Allam, H. (2024). Intelligent Automation: Leveraging LLMs in DevOps Toolchains. International Journal of AI, BigData, Computational and Management Studies, 5(4), 81-94.

[3] Hayes, A., Carter, E., Foster, D., Reynolds, S., Bennett, M., & Krishnan, J. (2022). From logs to insights: Generative AI for automated root-cause triage in distributed enterprise systems. International Journal of Science, Engineering and Technology, 10(2).

[4] Bukhari, T. T., Oladimeji, O., Etim, E. D., & Ajayi, J. O. (2024). Advances in End-to-End Pipeline Observability for Data Quality

Assurance in Complex Analytics Systems. International Journal of Advanced Multidisciplinary Research and Studies, 4(4), 1465-1487.

[5] Bhattacharyya, S. (2024). Cloud Innovation: Scaling with Vectors and LLMs. Libertatem Media Private Limited.

[6] Yamsani, N. (2024). Large Language Models for Intelligent Data Stewardship in Enterprises: Architectures, Provenance, and Evidence-Mapped Governance. International Journal of Computer Technology and Electronics Communication, 7(1), 8210-8219.

[7] Bairi, A. R. (2024). Vectorizing the Cloud: Advanced RAG Solutions. Libertatem Media Private Limited.

[8] Kanka, V. (2024). Scaling big data: leveraging LLMs for enterprise success. Libertatem Media Private Limited.

[9] Xu, X., Weytjens, H., Zhang, D., Lu, Q., Weber, I., & Zhu, L. (2025). RAGOps: Operating and Managing Retrieval-Augmented Generation Pipelines. arXiv preprint arXiv:2506.03401.

[10] Vedat, S. B., Yarkan, E. K., Akarsu, M., Karaman, R. K., Sar, A., Çelikbilek, Ç., & Saygılı, S. (2025). RAG-Driven Data Quality Governance for Enterprise ERP Systems. arXiv preprint arXiv:2511.16700.

[11] Karras, A., Theodorakopoulos, L., Karras, C., Theodoropoulou, A., Kalliampakou, I., & Kalogeratos, G. (2025). LLMs for Cybersecurity in the Big Data Era: A Comprehensive Review of Applications, Challenges, and Future Directions. Information, 16(11), 957.

[12] Parepalli, S. (2025). Architecting LLM Powered Support Bots for Data Engineering Operations Controlled Reasoning, Evidence Grounding, and Audit Ready Incident Workflows. Journal of Scientific and Engineering Research, 12(11), 211-225.

[13] Hansson, J. (2024). AIOps: How an existing Site Reliability Engineering team can leverage Artificial Intelligence in their IT-Operations.

[14] Marasani, Y. (2025). Explainable AI Frameworks for Patient-Level Claims Data Analytics. J Artif Intell Mach Learn & Data Sci, 8(1), 3382-3390.

[15] Kola, J. N. (2011). An Integrated Framework for Data Mining and Distributed Database Optimization in Resource-Constrained Network Environments. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, *2*(02), 82-86.

[16] Naidu, K. J. (2013). Performance Optimization Of ETL Pipelines In Distributed Data Warehouse Environments: A Network-Aware Scheduling Approach. *International Journal of Advance Industrial Engineering*, *1*(03), 63-67.

[17] Goel, N. Vulnerability Management in Computer Systems: Challenges and Approaches. *Educational Administration: Theory and Practice, 28 (04) 718-724 Doi: 10.53555/kuey. v28i4*, *11607*.

[18] Ravikumar, V. (2014). Fair and optimal resource allocation in wireless sensor networks.

[19] Naidu, K. J. (2014). Secure OLAP Reporting Architectures: Integrating Role-based Access Control and Query Execution Plan Optimization for Enterprise Analytical Environments. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, *5*(02), 155-159.

[20] Kola, J. N. (2017). DATA WAREHOUSING AND TEXT ANALYTICS AS INSTRUMENTS OF CULTURAL KNOWLEDGE MANAGEMENT: IMPLICATIONS FOR DIGITAL PRESERVATION AND SOCIETAL DECISION-MAKING. *Power System Protection and Control*, *45*(1), 11-15.

[21] Naidu, K. J. (2014). Secure OLAP Reporting Architectures: Integrating Role-based Access Control and Query Execution Plan Optimization for Enterprise Analytical Environments. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, *5*(02), 155-159.

[22] Takon, A. (2022). Advanced AI Techniques for Safety and Risk Evaluation in High-Hazard Engineering Systems. *International Journal of Technology, Management and Humanities*, *8*(04), 97-109.

[23] Warren, B. (2021). Transforming Enterprise Office Networks with EVPN-VXLAN: A BGP-Based Approach to Layer 2 Elimination. *International Journal of Technology, Management and Humanities*, *7*(04), 63-82.

[24] Ezeagwuna, D. (2022). Measuring Return on Investment (ROI) in Enterprise Service Management: The Role of Service Now in Enhancing Organizational Efficiency and Cost Reduction. *ADHYAYAN: A JOURNAL OF MANAGEMENT SCIENCES*, *12*(02), 59-71.

[25] Takon, A. (2020). Adaptive Pipeline Monitoring Using Unsupervised Anomaly Detection. *International Journal of Technology, Management and Humanities*, *6*(03-04), 93-106.

[26] MARASANI, Y. (2024). Enterprise Readiness for Generative AI: The Critical Role of Data Engineering. Frontiers in Computer Science and Artificial Intelligence, 3(2), 59-71.

[27] Singh, S. S. (2022). Accessibility and Universal Design in Transportation Infrastructure. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, *14*(04), 210-214.

[28] Nagraj, A. (2022). Modernizing legacy banking systems: Migration strategies and cost optimization in financial enterprises. Frontiers in Computer Science and Artificial Intelligence, 1(1), 43-52.

[29] MARASANI, Y. (2023). Machine Learning Models for Predicting Patient Treatment Switching Using Claims Data. Frontiers in Computer Science and Artificial Intelligence, 2(1), 59-66.

[30] ALAMPALLY, J. (2024). Enhancing data quality and trust in AI systems through robust data engineering. Frontiers in Computer Science and Artificial Intelligence, 3(1), 120-130.

[31] Khan, H. A. (2024). DATA-DRIVEN EPIDEMIOLOGICAL MODELING USING MACHINE LEARNING FOR DISEASE SPREAD FORECASTING AND PUBLIC HEALTH DECISION SUPPORT IN THE UNITED STATES. *International Journal of Applied Mathematics*, *37*(6s), 178-192.

[32] Ferdus, M. Z., Monsur, M. H., Akhtar, M. J., & Islam, S. (2024). Secured Auto Encryption and Authentication Process for Cloud Computing Security. *Valley International Journal Digital Library*, 1040-1044.

[33] Hossain, M. D., Kashem, M. A., Sadeq, M. J., Mustary, S., & Ferdus, M. Z. (2024, September). IoT Enabled Soil Fertilizer Monitoring and Recommendation System in the Context of Bangladeshi Agriculture. In *2024 IEEE International Conference on Power, Electrical, Electronics and Industrial Applications (PEEIACON)* (pp. 416-421). IEEE.

[34] Ferdus, M. Z., Anjum, N., Nguyen, T. N., Jisan, A. H., & Raju, M. A. H. (2024). The influence of social media on stock market: A transformer-based stock price forecasting with external factors. *Journal of Computer Science and Technology Studies*, *6*(1), 189-194.

[35] Arif, M. H. (2024). Optimizing Hospital Logistics and Healthcare Supply Chains Using Machine Learning and Artificial Intelligence Techniques. *J. Electrical Systems*, *20*(7s), 4209-4217.

[36] Abul Kashem, M., Hossain, D., Hasan Shuvo, M., Mustary, S., Ferdus, M. Z., & Uddin, J. (2025). An Explainable AI-Based Crop Recommendation Framework Leveraging IoT-Driven Environmental Data.

[37] Ferdus, M. Z., Bhuiyan, R. J., Brydie, D., Monsur, M. H., Shafi, A. H., Sani, Z. U., ... & Tabassum CN, M. (2025). AI-Driven Predictive Analytics for Early Diagnosis and Healthcare Cost Reduction.

International Journal of Medical and Health Research, *3*(4), 96-101.
[38] Das, P. K., Kashem, M. A., Ferdus, Z., & Islam, S. (2019, October). Development and application of a new computerized smell generating system. In *2019 Global Conference for Advancement in Technology (GCAT)* (pp. 1-5). IEEE.
[39] Ferdus, M. Z., Khan, M. N. I., Islam, S., & Kashem, M. A. (2019, October). VFLT: SQA Model for Cyber Physical System. In *2019 Global Conference for Advancement in Technology (GCAT)* (pp. 1-4). IEEE.
[40] Ferdus, M. Z., Islam, S., & Kashem, M. A. (2019, October). An innovative load balancing cluster composition of wireless sensor networks. In *2019 global conference for advancement in technology (GCAT)* (pp. 1-4). IEEE.
[41] Islam, S. J., Islam, S., Ferdus, M. Z., Khan, M. N. I., Kashem, M. A., & Islam, M. S. (2020, September). Load compactness and recognizing area aware cluster head selection of wireless sensor networks. In *2020 International conference on computing and information technology (ICCIT-1441)* (pp. 1-4). IEEE.
[42] Islam, S., Khan, M. N. I., Ferdus, M. Z., Islam, S. J., & Kashem, M. A. (2020, September). Improving throughput using cooperating TDMA scheduling of wireless sensor networks. In *2020 International Conference on Computing and Information Technology (ICCIT-1441)* (pp. 1-4). IEEE.
[43] Nagraj, A. (2024). GraphQL in Wealth Management Platforms: Optimizing Data Access and Performance. British Journal of Multidisciplinary Studies, 2(1), 16-24.
[44] ALAMPALLY, J. (2024). Real-Time and Near-Real-Time Analytics in Healthcare Data Ecosystems. Journal of Computer Science and Technology Studies, 6(1), 314-324.
[45] Takon, A. (2021). AI Safety Systems and Risk Analytics for High-Hazard Engineering Systems. *Multidisciplinary Innovations & Research Analysis*, *2*(2), 1-20.
[46] Kola, J. N. (2023). Quantifying Revenue Impact of Enterprise Analytics: A Revenue Attribution Framework for Business Intelligence Systems.
[47] Takon, A. (2023). Machine Learning (ML)–Based Cyber Threat Modelling for Industrial Control Systems in critical Infrastructure. *International Journal of Technology, Management and Humanities*, *9*(02), 94-108.
[48] Ezeagwuna, D. (2023). The Impact of Low-Code/No-Code Platforms on Business Innovation: A Study of Service Now's Contribution to Enterprise Agility and Digital Growth. *ADHYAYAN: A JOURNAL OF MANAGEMENT SCIENCES*, *13*(02), 90-99.
[49] Singh, S. S. (2023). Code Compliance Challenges in High-Stakes Infrastructure Projects. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, *15*(01), 213-221.
[50] Kola, J. N. (2023). Measuring the Business Value of Analytics-Driven Decisions: A Decision Impact Attribution Framework for Enterprise Environments.
[51] Singh, S. S. (2023). Architectural Identity in Transit Infrastructure: Branding vs Functionality. *Multidisciplinary Innovations & Research Analysis*, *4*(2), 1-12.
[52] Singh, S. S. (2023). Human-Centered Design in Underground Transit Environments. *Multidisciplinary Innovations & Research Analysis*, *4*(3), 1-20.
[53] Takon, A. (2024). Data-Driven Threat Intelligence for Energy and Critical Asset Management. *International Journal of Technology, Management and Humanities*, *10*(04), 253-266.
[54] Kola, J. N. Longitudinal Cohort Intelligence for Self-Insured Employer Groups: A Predictive Framework for Healthcare Cost Trajectory Modeling and Proactive Risk Intervention.
[55] Adepoju, S. A., & Adepoju, M. A. (2024). From Portals to Case Graphs: A Reference Architecture and Benchmark for Safety Investigation Operations with Agentic Orchestration.
[56] Takon, A. (2024). Data Science Approaches to Asset Integrity Management in Offshore and Onshore Oil and Gas Operations. *Multidisciplinary Innovations & Research Analysis*, *5*(2), 17-31.
[57] Ezeagwuna, D. (2024). Enterprise Workflow Automation and Workforce Productivity: Evaluating the Economic Benefits of Service Now Adoption Across Industries. *International Journal of Technology, Management and Humanities*, *10*(04), 299-313.
[58] Takon, A. (2024). Data-Driven Threat Intelligence for Energy and Critical Asset Management. *International Journal of Technology, Management and Humanities*, *10*(04), 253-266.
[59] Kola, J. N. Longitudinal Cohort Intelligence for Self-Insured Employer Groups: A Predictive Framework for Healthcare Cost Trajectory Modeling and Proactive Risk Intervention.
[60] Adepoju, S. A., & Adepoju, M. A. (2024). From Portals to Case Graphs: A Reference Architecture and Benchmark for Safety Investigation Operations with Agentic Orchestration.
[61] Takon, A. (2024). Data Science Approaches to Asset Integrity Management in Offshore and Onshore Oil and Gas Operations. *Multidisciplinary Innovations & Research Analysis*, *5*(2), 17-31.
[62] Ezeagwuna, D. (2024). Enterprise Workflow Automation and Workforce Productivity: Evaluating the Economic Benefits of Service Now Adoption Across Industries. *International Journal of Technology, Management and Humanities*, *10*(04), 299-313.
[63] Mukherjee, C. Ai-Driven Personalization of Power System Learning Modules Using Student Personas based on Behavioral Analysis of Grid Performance.
[64] Nadia, N. Y., Rabby, H. R., Arif, M. H., Tanvir, M. I. M., Ahmed, M., & Firdaus, S. (2025, October). Scalable RNN-Based Transfer Learning for Patient Sentiment Monitoring in Telehealth Platforms. In *2025 IEEE 2nd International Conference on Computing, Applications and Systems (COMPAS)* (pp. 1-6). IEEE.
[65] Takon, A. (2025). Explainable AI for Threat Modelling and Decision Support in Engineering Assets. *Journal of Cyber-Physical Security and Robotics*, *1*(02), 46-52.
[66] Mukherjee, C. (2025). Combating digital media piracy with agentic ai: Leveraging video transcription and character recognition for automated enforcement. *Authorea Preprints*.
[67] Anifowose, K. (2025). Development and Validation of AI-Assisted Analytical Methods for Biochemical Compound Detection in Pharmaceutical Chemistry. *Journal of Applied Pharmaceutical Sciences and Research*, *8*(4), 41-52.
[68] Mukherjee, C. (2025). Use of Agentic AI with OpenAI and Prompt Engineering and State-of-the Art Machine Learning Algorithm to detect the patterns in IOT Device Network Intrusion Attacks. *Authorea Preprints*.
[69] Ravikumar, V. (2025). Therapeutic Bot: Ethical Concerns in AI therapy for Neurodivergence. *J Int Scient Re Rep*.
[70] Mukherjee, C. (2025). Use of Agentic AI with LLM and Prompt Engineering and State-of-the Art Machine Learning Algorithm to detect the patterns in IOT Device Network Intrusion Attacks. *TechRxiv. August*, *6*.
[71] Takon, A. (2025). 3D Object Detection and Localization for Industrial Threat Monitoring. *Well Testing Journal*, *34*(S3), 850-880.
[72] Mukherjee, C. (2025). Harnessing large language models and ai agents for child behavior analytics in day care: a proof of concept for next-generation parental insight using simulated

data. *Machinery and Production Engineering*, *174*(2870), 26-34.
[73] Ezeagwuna, D. (2025). Artificial Intelligence for IT Service Management: Developing a Framework for ROI Optimization Using Service Now and Machine Learning Technologies. *Well Testing Journal*, *34*(S4), 394-419.
[74] Mukherjee, C. (2025). Combating digital media piracy with agentic ai: Leveraging video transcription and character recognition for automated enforcement. *Authorea Preprints*.
[75] Rajgopal, P. R. (2025). SOC Talent Multiplication: AI Copilots as Force Multipliers in Short-Staffed Teams. International Journal of Computer Applications, 187(48), 46-62.
[76] Albanese, M., Ou, X., Lybarger, K., Lende, D., Goldgof, D., Faisal, F. A., ... & Ghosh, A. (2025). Towards ai-driven human-machine co-teaming for adaptive and agile cyber security operation centers. ACM Transactions on Internet Technology.
[77] Mohammadi, D. S. A. (2022). Effects of Chronic Stress on Neuroendocrine and Immune Function: Clinical Implications for Early Intervention in Psychosomatic Disorders. Journal of Advanced Scientific Research, 13(03), 206-219.
[78] Mohammadi, D. S. A. (2022). Integrative Approaches in the Management of Anxiety and Depression: Comparing Standard Pharmacotherapy with Combined Cognitive Behavioral Therapy and Adjunct Holistic Interventions. Journal of Applied Pharmaceutical Sciences and Research, 3(3), 21-33.
[79] Mohammadi, D. S. A. (2023). The Role of Holistic Lifestyle Interventions (Mindfulness, Nutrition, Sleep Optimization and Traditional Therapies) in Improving Stress-Related Disorders and Quality of Life: A Systematic Review. INTERNATIONAL JOURNAL OF APPLIED PHARMACEUTICAL SCIENCES AND RESEARCH, 8(02), 42-54.
[80] Al Kalach, N. (2023). AI-Driven Enterprise System Integration: Improving Data Interoperability Across Complex Organizations. International Journal of Technology, Management and Humanities, 9(01), 128-149.
[81] Al Kalach, N. (2024). Enterprise Operational Intelligence Platforms: The Future of AI-Driven Business Infrastructure. Euro Vantage journals of Artificial intelligence, 1(2), 88-27.
[82] Goel, N. (2025). Federated Learning for Secure AI Models: Enhancing Privacy and Robustness in Decentralized Environments.
[83] Verma, A. THE QUANTUM LEAP FOR GRC: TRANSITIONING TO CRYPTO-AGILITY IN CLOUD INFRASTRUCTURE.
[84] Verma, A. (2025). Blockchain for Cyber Security: Enhancing Data Integrity and Trust in Digital Transactions.
[85] Verma, A. (2022). Twin Poisoning: Analyzing the Impact of False Data Injection Attacks on Digital Twin-Based Decision Support Systems. International Journal of Technology, Management and Humanities, 8(03), 39-61.
[86] Verma, A. QUANTIFYING ZERO TRUST: DEVELOPING GRC METRICS FOR MATURE CLOUD ENVIRONMENTS.
[87] Al Kalach, N. (2023). Transforming Fragmented Enterprise Data into Actionable Insights Using Artificial Intelligence. International Journal of Technology, Management and Humanities, 9(01), 150-174.