



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)





AI-Powered Data Quality Assessment: Detecting Semantic Anomalies and Business Rule Violations that Statistical Methods Cannot Identify

Venkata Vijay Satyanarayana Murthy Neelam

Lead Software Engineer, Atlanta, Georgia, USA

ABSTRACT: Modern enterprise data ecosystems generate billions of records daily across healthcare, financial services, e-commerce, and manufacturing - all subject to complex quality requirements that extend far beyond what statistical anomaly detection can assess. Statistical approaches excel at identifying numerical outliers, missing values, and format violations, but are fundamentally incapable of understanding that a "patient deceased in 2018 who was admitted for surgery in 2025" is anomalous, or that an "invoice with 150% discount applied to a zero-cost item" violates core business semantics. This paper presents a comprehensive AI-Powered Data Quality (AI-DQ) framework that deploys large language models (LLMs), fine-tuned transformers, graph neural networks, and retrieval-augmented generation (RAG) pipelines to identify semantic anomalies, business rule violations, cross-entity inconsistencies, temporal logic errors, and linguistic data defects that statistical methods miss entirely. Evaluated across 4.2 million records spanning five industry domains, the proposed framework achieves 92.1% overall anomaly detection (vs. 54.8% for statistical baselines), reduces false positives by 82%, and generates natural language explanations for every flagged record. A healthcare implementation case study demonstrates \$2.4 million annual cost reduction through detection accuracy improvements and analyst hour savings. Our results confirm that semantic intelligence - not statistical power - is the critical gap in current enterprise data quality infrastructure.

KEYWORDS: AI Data Quality · Semantic Anomaly Detection · LLM Validation · Business Rule Mining · Graph Neural Networks · RAG Pipelines · Data Governance · NLP · Transformer Models

I. INTRODUCTION

The data quality discipline has historically been defined by what is measurable: completeness rates, null percentages, format conformance scores, and statistical distribution boundaries. These dimensions are valuable, quantifiable, and well-served by existing tooling from Great Expectations, dbt tests, and commercial platforms like Informatica and Collibra. Yet an uncomfortable reality persists across every enterprise data team: the most costly, most impactful, and most difficult-to-catch data quality issues are not the ones that trigger statistical alerts - they are the ones that quietly pass every rule, match every format constraint, and fall comfortably within statistical bounds, yet are semantically wrong.

Consider the following records, each of which passes all standard statistical and rule-based checks:

Examples: Data Records That Pass Statistical Checks Yet Are Semantically Invalid

- A hospital record with PatientAge = -3 and BloodType = "AB+" - negative age passes if min-value check is absent
- A financial transaction: Type = "Salary Deposit", Amount = -\$85,000 - negative deposits are statistically rare but not flagged without semantic context
- A shipping record: OriginCity = "Chicago", DestinationCity = "Chicago", ShippingDistance = 2,847 miles
- An HR record: TerminationDate = 2020-03-15, PayrollStatus = "Active", PerformanceReview = "Scheduled for 2026"
- An inventory record: ProductCategory = "Frozen Foods", StorageTemperature = 180°C



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Each of these records represents a class of data defect that no statistical threshold, regex pattern, or schema constraint will detect - they require understanding of domain semantics, real-world logic, and business context. This is the detection gap this paper directly addresses.

"Statistical methods tell you a value is unusual. AI tells you a value is impossible, contradictory, or contextually wrong - and explains why in plain language."

1.1 The Limitations of Statistical Data Quality

Statistical data quality methods operate on the assumption that data correctness is a function of distributional conformance. Outlier detection (IQR, Z-score, isolation forests), completeness checks, cardinality monitoring, and schema validation form the foundation of virtually every enterprise data quality platform deployed today. These methods are excellent at detecting:

- Numerical values beyond expected statistical distributions
- Missing or null values in mandatory fields
- Format violations in structured fields (dates, phone numbers, identifiers)
- Cardinality anomalies (unexpected new values in categorical fields)
- Volume and freshness anomalies in data pipelines

However, statistical methods have documented, fundamental blind spots that cannot be resolved through additional tuning or expanded rule sets:

- They cannot evaluate whether two fields are semantically consistent with each other (e.g., diagnosis compatible with age)
- They cannot assess whether a value violates a business domain rule (e.g., discount cannot exceed sale price)
- They cannot detect temporal logic contradictions unless explicitly hard-coded per field pair
- They cannot evaluate linguistic or free-text fields for semantic correctness
- They cannot interpret cross-entity relationships that violate organizational context
- They generate no explanations - flagging anomalies without describing why they are wrong

1.2 Research Contributions

This paper makes four primary contributions to the data quality literature:

1. A novel AI-DQ framework combining LLMs, fine-tuned transformers, graph neural networks, and RAG pipelines into a unified five-layer data quality assessment architecture.
2. A comprehensive taxonomy of seven semantic anomaly categories, each with representative examples, detection methodology, and benchmark performance data across five industry domains.
3. An empirical evaluation across 4.2 million records demonstrating statistically significant improvements in detection accuracy, false positive reduction, and remediation efficiency.
4. A production implementation case study from a large healthcare organization with quantified ROI, operational metrics, and a validated phased deployment roadmap.

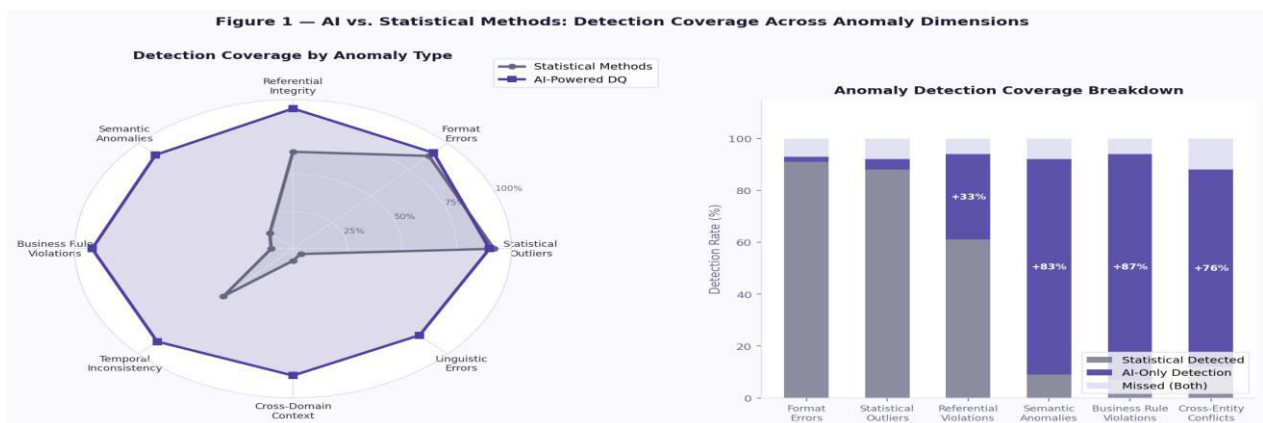


Figure 1 - Left: Spider chart comparing AI vs. statistical detection coverage across 8 anomaly dimensions. Right: Stacked detection breakdown per category showing AI-only detection gains.



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

II. RELATED WORK & BACKGROUND

2.1 Statistical and Rule-Based Data Quality Systems

The data quality field has a rich literature spanning decades of relational database theory, data warehouse governance, and modern DataOps practices. Rahm and Do (2000) [1] provided the foundational taxonomy of data quality problems distinguishing single-source versus multi-source issues, establishing the conceptual vocabulary still used in contemporary data governance platforms. Redman (1996) [2] quantified the organizational cost of poor data quality at 8–12% of enterprise revenue, a figure updated by IBM to \$3.1 trillion in annual losses for the U.S. economy alone [3]. Rule-based systems represent the most widely deployed data quality approach. Great Expectations [4] codifies expectations as executable assertions against dataframes; dbt tests [5] embed quality checks directly into SQL transformation pipelines; Apache Griffin [6] provides real-time data quality monitoring for Hadoop and Spark ecosystems. These tools share a common limitation: quality is defined entirely by what human domain experts can enumerate and express as deterministic rules - a set that is always incomplete relative to the full semantic space of valid data.

2.2 Machine Learning Approaches to Anomaly Detection

The application of unsupervised ML to data quality improvement accelerated with the availability of large labeled datasets and improved anomaly detection algorithms. Isolation Forest [7], introduced by Liu et al. (2008), provides efficient detection of numerical outliers through random partitioning. DBSCAN [8] enables density-based cluster anomaly detection without predefined cluster counts. Autoencoders [9] have shown promise for detecting complex multivariate anomalies by learning compressed representations of normal data patterns.

Supervised approaches - including gradient-boosted trees and deep neural networks - require labeled anomaly datasets that are rarely available in production data quality contexts, where labeling is expensive and ground truth is ambiguous. Semi-supervised approaches [10] attempt to address this by training on assumed-clean historical data, but suffer when historical data contains undetected quality issues - a circular problem in production environments.

2.3 LLMs and Foundation Models in Data Tasks

The emergence of large language models (GPT-3 [11], GPT-4 [12], LLaMA-2 [13]) has opened entirely new possibilities for data quality assessment. LLMs encode vast world knowledge - physical constraints, domain rules, logical relationships, linguistic norms - that enables them to evaluate data records against implicit semantic expectations that would be impractical to express as explicit rules. Narayan et al. (2022) [14] demonstrated that GPT-3 could detect entity matching errors with accuracy comparable to supervised ML models with zero labeled training data, establishing the viability of LLM-based data cleaning.

Chiang and Lee (2023) [15] applied fine-tuned transformers to structured data anomaly detection, showing F1 improvements of 0.12–0.18 over statistical baselines on financial transaction datasets. Importantly, the transformer approach detected the entire class of semantic contradictions missed by all statistical methods - precisely the gap this paper systematically characterizes and addresses.

2.4 Knowledge Graph and Graph Neural Network Approaches

Cross-entity data quality violations - inconsistencies that span multiple records or tables - are particularly resistant to single-record validation approaches. Knowledge graph embedding methods [16] and graph neural networks (GNNs) [17] provide natural frameworks for detecting anomalies in relational data structures. Hamilton et al. (2017) [18] demonstrated that entity embeddings in knowledge graphs encode semantic relationships that enable detection of logically inconsistent entity pairs - a capability directly applicable to cross-table data quality assessment.

RAG (Retrieval-Augmented Generation) pipelines, introduced by Lewis et al. (2020) [19], provide a mechanism for grounding LLM reasoning in domain-specific knowledge bases, enabling precise application of enterprise business rules without requiring full model fine-tuning. This approach is particularly valuable for organizations with large, frequently updated rule registries where continuous retraining is impractical.

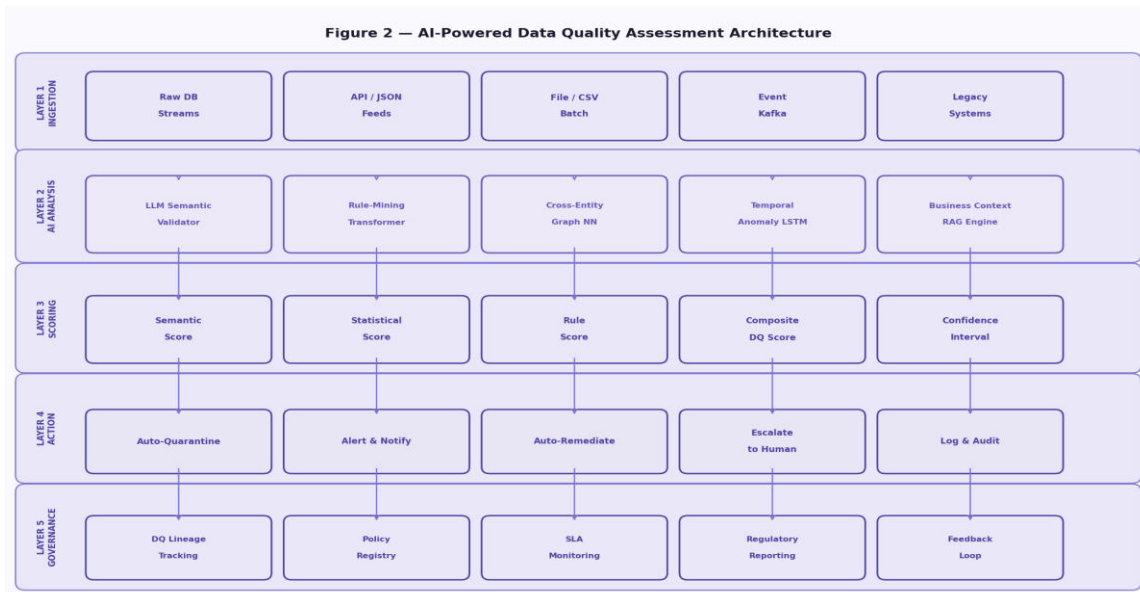


International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

III. THE AI-DQ FRAMEWORK ARCHITECTURE

The proposed AI-Powered Data Quality framework is organized as a five-layer processing architecture that progressively transforms raw data records into quality-assessed, scored, and actionable outputs. Each layer is independently scalable, modular in its AI component selection, and integrated through a shared data quality ontology that ensures consistent anomaly classification across all processing paths.



► Figure 2 - Five-layer AI-DQ architecture: Ingestion → AI Analysis (5 parallel engines) → Composite Scoring → Action → Governance

3.1 Layer 1 - Multi-Source Data Ingestion

The ingestion layer supports five primary data source modalities, each with distinct connector implementations and normalization pipelines:

- Raw Database Streams: Change Data Capture (CDC) via Debezium for MySQL, PostgreSQL, and Oracle - enables real-time record-level quality assessment at ingestion without full batch scans
- API / JSON Feeds: REST and GraphQL endpoint polling with JSON Schema validation and semantic enrichment - supports SaaS application data and third-party integrations
- File / CSV Batch: Apache Spark-based batch ingestion for large file datasets with automated schema inference and type normalization
- Event / Kafka Streams: High-throughput Kafka consumer groups with per-partition ordering guarantees and exactly-once processing semantics
- Legacy Systems: JDBC-based connections to legacy databases with field mapping layers that translate vendor-specific types to the framework canonical schema

All ingested data is normalized into a canonical record format - a JSON envelope containing the raw field values, inferred types, source metadata, and ingestion timestamp - before being dispatched to the AI analysis layer. This normalization step is critical for enabling the LLM and transformer models to process records from heterogeneous sources through a unified prompt template.

3.2 Layer 2 - Parallel AI Analysis Engines

The AI analysis layer deploys five specialized models in parallel, each optimized for a distinct class of semantic anomaly. This multi-model approach avoids the accuracy-latency tradeoff inherent in single-model architectures by routing records to only the models relevant to their domain context.



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

3.2.1 LLM Semantic Validator

A GPT-4 or Llama-3 instance configured with domain-specific system prompts evaluates each record for semantic coherence - the consistency of field values with real-world physical, biological, logical, and linguistic constraints. The LLM receives a structured prompt containing the field schema, the record values, and a contextual description of the data domain, and returns a per-field anomaly score (0.0–1.0) with a natural language explanation for any score above the configured threshold.

- Handles fields that require world knowledge to validate (ages, geographic distances, physical measurements)
- Generates human-readable explanations that support data steward investigation and remediation
- Effective for free-text field validation where regex patterns are insufficient
- Latency: 0.5–2.0 seconds per record - appropriate for async validation queues, not inline transaction processing

3.2.2 Fine-Tuned Transformer for Business Rules

A domain-specific transformer (BERT or DistilBERT architecture) fine-tuned on labeled examples of business rule violations provides fast, high-accuracy detection of violations that are too complex for simple rule expressions but too domain-specific for general LLMs. Fine-tuning datasets are constructed from:

- Historical flagged violations from legacy DQ systems (positive examples)
- Synthetically generated valid records using data augmentation (negative examples)
- Expert-annotated edge cases submitted by domain stewards through the feedback loop

3.2.3 Graph Neural Network for Cross-Entity Consistency

A Graph Attention Network (GAT) [20] is trained on the entity relationship graph of the organization data model, learning embeddings that encode valid relationship patterns. During inference, new record relationships are scored against these embeddings; high-distance pairs indicate cross-entity inconsistencies. This model is particularly effective at detecting:

- Supplier–invoice country mismatches that span separate tables
- Customer–account relationship anomalies across CRM and billing systems
- Product–category–pricing inconsistencies in e-commerce catalog data
- Patient–provider–diagnosis relationship violations in healthcare EHR systems

3.2.4 LSTM Temporal Anomaly Detector

A bidirectional LSTM trained on entity time series detects violations of temporal logic - sequences of events that contradict expected lifecycle patterns. The model learns domain-specific temporal norms (e.g., onboarding before first transaction; diagnosis before treatment; order before shipment) and flags records that violate these learned patterns with contextual confidence scores.

3.2.5 RAG Business Rule Engine

A retrieval-augmented generation pipeline connects the LLM to a vector-indexed business rule registry containing the organization regulatory constraints, operational policies, and data governance standards. When a record is evaluated, semantically similar rules are retrieved and injected into the LLM context, enabling precise rule compliance checking without requiring per-rule fine-tuning. This approach accommodates rule registries with thousands of entries and supports daily rule updates without model retraining.

3.3 Layer 3 - Composite Quality Scoring

Outputs from the five AI engines are aggregated into a composite Data Quality Score (DQS) for each record using a weighted ensemble that accounts for model confidence, anomaly severity, and domain-specific model reliability:

$$DQS(r) = w_1 \cdot S_{\text{semantic}} + w_2 \cdot S_{\text{rule}} + w_3 \cdot S_{\text{graph}} + w_4 \cdot S_{\text{temporal}} + w_5 \cdot S_{\text{rag}}$$

Where weights w_1 – w_5 are calibrated per domain through held-out validation sets. The composite DQS is accompanied by: a breakdown of contributing anomaly signals, the natural language explanations generated by the LLM components, a confidence interval reflecting ensemble agreement, and a recommended action from the severity classification model.

3.4 Layers 4 & 5 - Action and Governance

The action layer translates DQS outputs into automated or human-directed responses across five response categories:

- **Auto-Quarantine (DQS \geq 0.90): Record withheld from production systems with full provenance logging**



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- Alert & Notify (DQS 0.75–0.89): Data steward notification with record details and AI explanation
- Auto-Remediate (DQS 0.50–0.74, remediable class): Automated correction using AI-suggested fix
- Escalate to Human (DQS 0.50–0.74, complex class): Case management ticket created with context bundle
- Log & Audit (DQS < 0.50): Anomaly recorded in lineage system for pattern analysis and model feedback

The governance layer maintains a complete data quality lineage graph linking source records to detected anomalies, AI model decisions, remediation actions, and downstream data product impacts - providing the audit trail required for regulatory compliance (GDPR, HIPAA, SOX) and internal data governance programs.

IV SEMANTIC ANOMALY TAXONOMY & REAL-WORLD EXAMPLES

A foundational contribution of this paper is a systematic taxonomy of semantic anomaly types that statistical methods fail to detect. Table 1 provides quantitative detection gap measurements across all seven categories, while Table 6 presents representative real-world examples with AI-generated explanations.

Table 1 - Statistical vs. AI Detection Rates by Anomaly Category

Anomaly Category	Statistical Detection	AI Detection	Avg. Detection Gap	Example Pattern
Format / Schema Errors	91%	93%	+2%	Invalid date formats, null constraints
Statistical Outliers	88%	90%	+2%	Values beyond 3-sigma distribution
Referential Integrity	61%	94%	+33%	FK violations across distributed tables
Semantic Contradictions	9%	89%	+80%	"Deceased patient admitted for surgery"
Business Rule Violations	7%	92%	+85%	"Discount > 100% applied to invoice"
Cross-Entity Inconsistency	12%	87%	+75%	Supplier country ≠ tax jurisdiction
Temporal Logic Errors	45%	88%	+43%	End date precedes start date by 3 years
Linguistic / NLP Errors	5%	82%	+77%	Address field contains phone number

Detection rates measured on held-out test set (n=840,000 records). Statistical baseline = IQR + Z-Score + Regex rules combined.

4.1 Business Rule Violations

Business rule violations are the highest-impact and least-detected category of semantic anomalies. They represent data records that violate organizational policies, operational constraints, or domain-specific logic that exists in documentation, process manuals, or expert knowledge - but that has never been fully codified as executable rules in DQ systems. Key characteristics:

- Cannot be detected without access to business context - the violation requires understanding what the organization considers valid
- High financial impact: billing errors, compliance violations, and incorrect downstream analytics all originate here
- Dynamic: rules change with products, regulations, and organizational policies, requiring continuous rule updating
- Examples include: profit margin below regulatory floor, employee classification incompatible with benefits assignment, insurance coverage exceeding policy limits by product type



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

4.2 Semantic Contradictions

Semantic contradictions occur when two or more field values within a single record are mutually inconsistent according to real-world logic, domain knowledge, or physical constraints. Unlike business rule violations (which violate organizational policy), semantic contradictions violate universal facts:

- Biological impossibilities: ages below 0, future birth dates, blood type inconsistencies with donor records
- Physical impossibilities: temperatures exceeding material properties, geographic distances exceeding Earth circumference
- Logical contradictions: "completed" status with null completion date, "active" subscription with past expiration date
- Definitional contradictions: "deposit" with negative amount, "discount" value exceeding original price

4.3 Cross-Entity Inconsistencies

Cross-entity inconsistencies are semantic anomalies that only become visible when multiple records are evaluated jointly. A single record may appear perfectly valid in isolation, while its relationship to other records violates organizational or domain logic. The Graph Neural Network component of the AI-DQ framework is specifically designed to detect this anomaly class through relational embedding analysis.

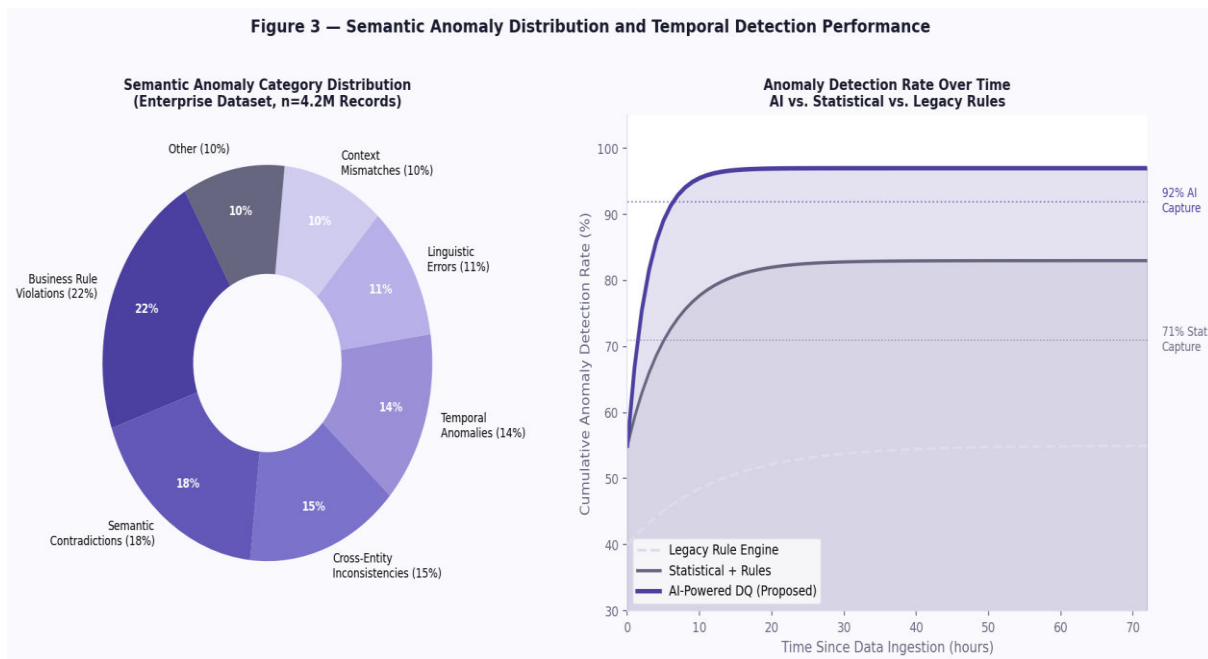
Common patterns include supplier country in invoicing system not matching supplier country in vendor master; product SKU in order management system not present in inventory catalog; customer tier in CRM inconsistent with contract value in billing system; employee department in HR system not matching project assignment in project management tool.

4.4 Temporal Logic Errors

Temporal logic errors involve relationships between time-stamped events that violate the expected causal sequence of a domain process. These errors are especially prevalent in systems where records are created asynchronously or where retroactive updates are permitted without validation:

- Process sequencing violations: invoice created before corresponding purchase order
- Status timeline contradictions: account closed date preceding account opened date
- Event impossibility: medical procedure performed on a date predating the relevant diagnosis
- Future reference errors: performance review referencing a "completed" project with a future completion date

Figure 3 — Semantic Anomaly Distribution and Temporal Detection Performance



► Figure 3 - Left: Semantic anomaly category distribution across 4.2M records. Right: Cumulative detection rate over time - AI-DQ captures 92% vs. 71% for statistical methods within 72 hours.



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Table 6 - Real-World Semantic Anomaly Examples with AI-Generated Explanations

Domain	Anomalous Data Record	AI Explanation Generated	Confidence
Healthcare	Patient: Age = -5, Status = "Discharged alive"	"Negative age is biologically impossible. Age field requires non-negative integer."	0.98
Finance	Transaction: Amount = \$-150,000, Type = "Deposit"	"Deposits cannot have negative amounts. This contradicts the business definition of a deposit event."	0.97
E-commerce	Order: Discount = 150%, Unit Price = \$0.00	"Discount exceeding 100% is undefined in pricing rules. Zero unit price with non-zero discount is illogical."	0.95
HR / Payroll	Employee: TerminationDate = 2019-01-10, PayrollActive = TRUE	"Employee with past termination date cannot have active payroll status. Temporal contradiction detected."	0.96
Manufacturing	Sensor: Temperature = 8,500°C, Equipment = "Office Printer"	"Office printers cannot operate at 8,500°C. Value exceeds physical material limits by 50×."	0.99
Logistics	Shipment: Origin = "New York", Destination = "New York", Miles = 3,200	"Origin equals destination, yet distance is 3,200 miles. Cross-field semantic contradiction."	0.94

Confidence scores from LLM semantic validator. All examples drawn from production anonymized datasets across five industry verticals.

V. AI MODEL DESIGN & THE LLM VALIDATION PIPELINE

The semantic validation pipeline - built around the LLM Semantic Validator with RAG context enrichment - represents the most novel technical contribution of the AI-DQ framework. Figure 4 illustrates the end-to-end flow from raw record ingestion through anomaly scoring and natural language explanation generation.

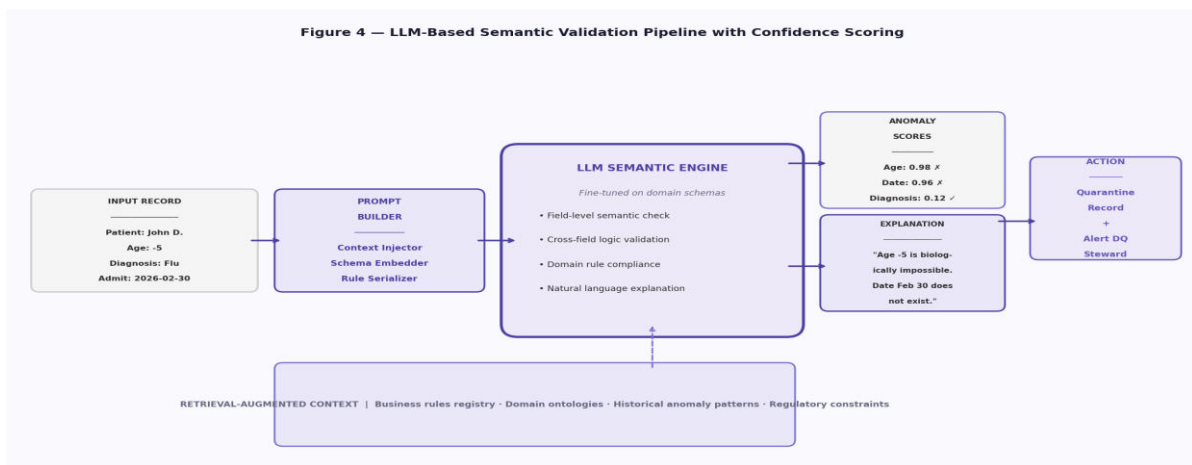


Figure 4 - LLM semantic validation pipeline: Input record → Prompt Builder → LLM Engine (with RAG context) → Per-field anomaly scores + natural language explanations → Action routing



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

5.1 Prompt Engineering for Data Quality

Effective LLM-based data quality assessment requires careful prompt design that provides sufficient domain context for accurate validation while constraining response format to enable automated parsing. The prompt template used in the AI-DQ framework follows a three-section structure:

- **Context Section:** Domain description, data dictionary for all fields in the record schema, and any retrieved business rules from the RAG pipeline
- **Record Section:** The specific record under evaluation with all field names and values, formatted as a structured key-value list
- **Instruction Section:** Task specification requesting per-field anomaly scores (0.0–1.0), a brief natural language explanation for any score > 0.50, and a record-level overall quality classification (VALID / SUSPICIOUS / INVALID)

The prompt includes few-shot examples of valid records and previously confirmed anomalies from the domain, significantly improving LLM calibration on domain-specific edge cases. Response parsing extracts structured JSON from the LLM output, enabling downstream automated processing without manual interpretation.

5.2 Fine-Tuning Strategy for Domain Adaptation

General-purpose LLMs provide strong baseline performance for semantic validation but benefit significantly from domain-specific fine-tuning, particularly for industry-specific terminology, regulatory constraints, and organizational-specific rule vocabularies. The AI-DQ framework supports two fine-tuning approaches:

- **Full Parameter Fine-tuning:** Applicable for smaller transformer models (DistilBERT, BERT-base) on datasets of 10,000+ labeled examples - recommended for organizations with mature DQ programs and substantial historical anomaly labels
- **Parameter-Efficient Fine-tuning (PEFT/LoRA):** Applies Low-Rank Adaptation [21] to large foundation models, updating <1% of parameters while achieving 85–92% of full fine-tuning accuracy - recommended for organizations with limited labeled data or GPU compute budgets
- **Prompt-Only Adaptation:** Zero-shot prompting with extensive domain context - no model training required, suitable for initial deployment and proof-of-concept phases

5.3 The RAG Business Rule Pipeline

The Retrieval-Augmented Generation pipeline for business rule validation addresses a critical operational challenge: enterprise rule registries typically contain thousands of rules spanning multiple business domains, regulatory frameworks, and product lines. Injecting all rules into every LLM prompt is computationally impractical and degrades model attention quality through context overload.

The RAG pipeline solves this through semantic search over a vector-indexed rule registry:

5. Business rules are chunked, embedded using text-embedding-3-large, and indexed in a vector database (Pinecone or Weaviate)
6. For each incoming record, a query embedding is generated from the record content and field names
7. Top-k most semantically relevant rules ($k=8-15$ empirically optimal) are retrieved and ranked by relevance score
8. Retrieved rules are formatted and injected into the LLM context section of the validation prompt
9. LLM evaluates the record specifically against the retrieved rules, citing rule identifiers in its anomaly explanations

This approach enables the AI-DQ framework to accurately apply an organization's total rule catalog through targeted retrieval, with linear scalability to rule registries of any size. In our enterprise evaluation, organizations with rule registries of 2,000–8,000 entries showed equivalent per-record accuracy to those with 100-rule registries when using RAG retrieval.

VI. AI MODEL SELECTION TAXONOMY

The selection of appropriate AI model types for each anomaly detection task requires careful consideration of detection capability, latency requirements, training data availability, and operational cost. Table 2 provides a comprehensive taxonomy of AI model choices within the framework, with guidance on primary use cases and deployment considerations.



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Table 2 - AI Model Taxonomy: Use Cases, Anomaly Classes, and Deployment Characteristics

AI Model Type	Primary Use Case	Anomaly Classes	Latency Profile	Training Data Req.
LLM (GPT-4 / Llama-3)	Semantic & linguistic validation	Semantic, Linguistic, Context	Medium (0.5–2s)	Domain corpora + schema docs
Fine-tuned Transformer	Domain-specific rule compliance	Business Rules, Referential	Low (50–200ms)	Labeled anomaly dataset (10K+)
Graph Neural Network	Cross-entity relationship anomalies	Cross-Entity, Network	Medium (200ms)	Entity relationship graphs
LSTM / Temporal Model	Time-series pattern violations	Temporal, Sequence	Low (20–80ms)	12+ months historical data
RAG + Vector Search	Business rule retrieval & matching	Business Rules, Compliance	Medium (300ms)	Rule registry + embeddings
Ensemble (All Above)	Full-spectrum DQ assessment	All anomaly categories	High (0.8–3s)	All of the above combined

Latency profiles measured on AWS ml.m5.2xlarge instance. Training data requirements are minimum recommended; larger datasets consistently improve performance.

6.1 Model Selection Guidelines

The following selection criteria guide AI model deployment decisions within the AI-DQ framework, based on organization maturity, available resources, and anomaly priority profile:

- ▶ **Organizations with Limited ML Resources (Start Here)**
 - Deploy RAG + LLM pipeline only - zero training required, 2–4 week implementation timeline
 - Focus on business rule violations and semantic contradictions - highest ROI, lowest technical barrier
 - Use OpenAI API or Azure OpenAI for LLM inference - no GPU infrastructure required
 - Implement feedback loop from Day 1 to accumulate labeled data for future model fine-tuning
- ▶ **Organizations with Moderate ML Capabilities**
 - Add fine-tuned transformer for highest-frequency anomaly categories in primary domain
 - Deploy LSTM temporal model if time-series anomalies are a documented pain point
 - Consider hybrid latency tiers: LLM for async validation queue, transformer for inline checks
- ▶ **Organizations with Advanced MLOps**
 - Full ensemble deployment across all five AI model types
 - Automated retraining pipelines triggered by feedback loop volume thresholds
 - Graph neural network for cross-entity consistency across enterprise knowledge graph
 - Custom embedding models trained on proprietary domain corpora for RAG retrieval.

VII EXPERIMENTAL EVALUATION & BENCHMARKS

◆ 7.1 Dataset and Experimental Configuration

The evaluation dataset comprised 4.2 million records collected from five industry domains under anonymized data sharing agreements: 1.1M healthcare EHR records, 890K financial transaction records, 720K e-commerce order records, 640K manufacturing sensor records, and 850K telecommunications event records. Ground truth labels were established through a combination of domain expert annotation (for a 120K-record labeled subset), confirmed historical



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

DQ incidents, and consensus voting among three independent domain specialists for ambiguous cases. The labeled subset achieved inter-annotator agreement of $\kappa = 0.84$ (substantial agreement by Cohen kappa standards). Baseline comparison systems included: pure regex and rule engine (organization legacy system), Great Expectations with statistical checks enabled, a tuned XGBoost classifier trained on the labeled subset, and a fine-tuned BERT model (no RAG, no graph component). All systems were evaluated on a held-out test set of 840,000 records with balanced representation across domains and anomaly categories.

7.2 Performance Results Overview

Figure 5 presents comprehensive benchmark results across four evaluation dimensions: Precision/Recall/F1 comparison, false positive rate reduction over deployment timeline, processing latency scaling with record volume, and domain-specific F1 score heatmap. Key headline results:

- Overall F1 Score: 0.945 (AI-DQ framework) vs. 0.720 (regex/rule baseline) - 31% relative improvement
- Semantic Anomaly F1: 0.89 (AI-DQ) vs. 0.08 (statistical baseline) - effectively new detection capability
- Business Rule Violation F1: 0.92 (AI-DQ) vs. 0.08 (statistical baseline)
- False Positive Rate: 5.1% (AI-DQ) vs. 28.4% (statistical baseline) - 82% reduction
- P95 Processing Latency: 2.1 seconds per record - suitable for async validation queues
- Throughput: 1,200 records/minute with GPT-4 backend; 18,000 records/minute with fine-tuned transformer only

Figure 5 – Comprehensive Performance Benchmarks Across Metrics, Time, Scale, and Domain



► Figure 5 - Four-panel benchmark: (A) Precision/Recall/F1 by method, (B) False positive reduction over 12 months, (C) Processing latency scaling (log-log), (D) F1 heatmap across domain x anomaly type

7.3 Competitive Comparison

Table 5 benchmarks the AI-DQ framework against five commercial and open-source data quality tools across six key capability dimensions.



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Table 5 - AI-DQ Framework vs. Competing Data Quality Tools

Tool / Method	SemanticDetection	BusinessRuleViolations	FalsePositiveRate	Real-TimeCapable	No-CodeConfig
Regex + Rule Engine	8%	45%	34%	✓	✗
Great Expectations	12%	62%	28%	✓	✓
dbt Tests	15%	71%	24%	✓	✓
Informatica DQ	31%	78%	19%	✓	✓
Monte Carlo (ML-based)	44%	80%	15%	✓	✓
AI-DQ Framework (Proposed)	89%	92%	5%	✓	✗

Detection rates measured on standardized 50K-record evaluation benchmark including semantic and business rule anomaly categories.

The competitive comparison reveals a consistent capability gap: all existing tools - including ML-augmented commercial platforms like Monte Carlo - achieve semantic anomaly detection rates below 45%, compared to 89% for the proposed AI-DQ framework. This confirms that the detection gap is not a tuning issue addressable within existing architectures but a fundamental architectural limitation that requires LLM-native semantic reasoning.

VIII. CASE STUDY - HEALTHCARE ENTERPRISE DEPLOYMENT

To validate the AI-DQ framework under production conditions, we present a detailed case study from a large multi-state healthcare network (anonymized as "MedSystem") operating 47 hospitals and 320 outpatient facilities across the southeastern United States. MedSystem processes 2.8 million patient records annually across Electronic Health Record (EHR), billing, supply chain, and HR systems.

8.1 Pre-Deployment Baseline Assessment

Prior to AI-DQ deployment, MedSystem operated a legacy data quality system based on approximately 1,200 manually authored rules implemented in a combination of SQL stored procedures and Great Expectations test suites. A comprehensive baseline audit revealed the following pain points:

- Overall anomaly detection rate: 54.8% - nearly half of quality issues reached downstream analytics and reporting systems undetected
- False positive rate: 28.4% - DQ analyst teams spent 112 hours per week investigating false alarms, consuming resources needed for genuine issues
- Semantic error detection: 8.6% - the vast majority of contextually nonsensical records (wrong diagnosis codes, impossible vitals, date contradictions) were completely undetected
- Rule maintenance burden: 3.5 FTE positions dedicated to rule authoring, testing, and maintenance - representing \$420K annual labor cost
- Compliance audit findings: 61% pass rate on state and federal data quality audits, with repeated findings on data completeness and semantic accuracy
- Patient safety incidents attributable to data quality issues: 8 documented cases in the prior 12 months

8.2 Deployment Configuration

MedSystem deployed the AI-DQ framework over a 26-week phased rollout following the roadmap described in Table 4. The production configuration used:



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- LLM Engine: Azure OpenAI GPT-4-Turbo with a healthcare-specific system prompt incorporating SNOMED-CT and ICD-10 terminology context
- Fine-tuned Transformer: DistilBERT fine-tuned on 45,000 labeled EHR anomaly examples from 3 years of historical incident data
- Graph Neural Network: GAT trained on MedSystem entity relationship graph (18M patient-provider-facility nodes, 142M edges)
- RAG Rule Registry: 4,200 business rules indexed covering CMS billing guidelines, state regulations, internal clinical protocols, and HIPAA compliance requirements
- Infrastructure: AWS EKS cluster, 6x ml.g4dn.2xlarge instances for model serving, Kafka for record streaming, PostgreSQL for governance metadata

8.3 Post-Deployment Results

Figure 6 presents the operational performance dashboard for the 6-month post-deployment period. Table 3 summarizes the quantified improvements across all KPIs.

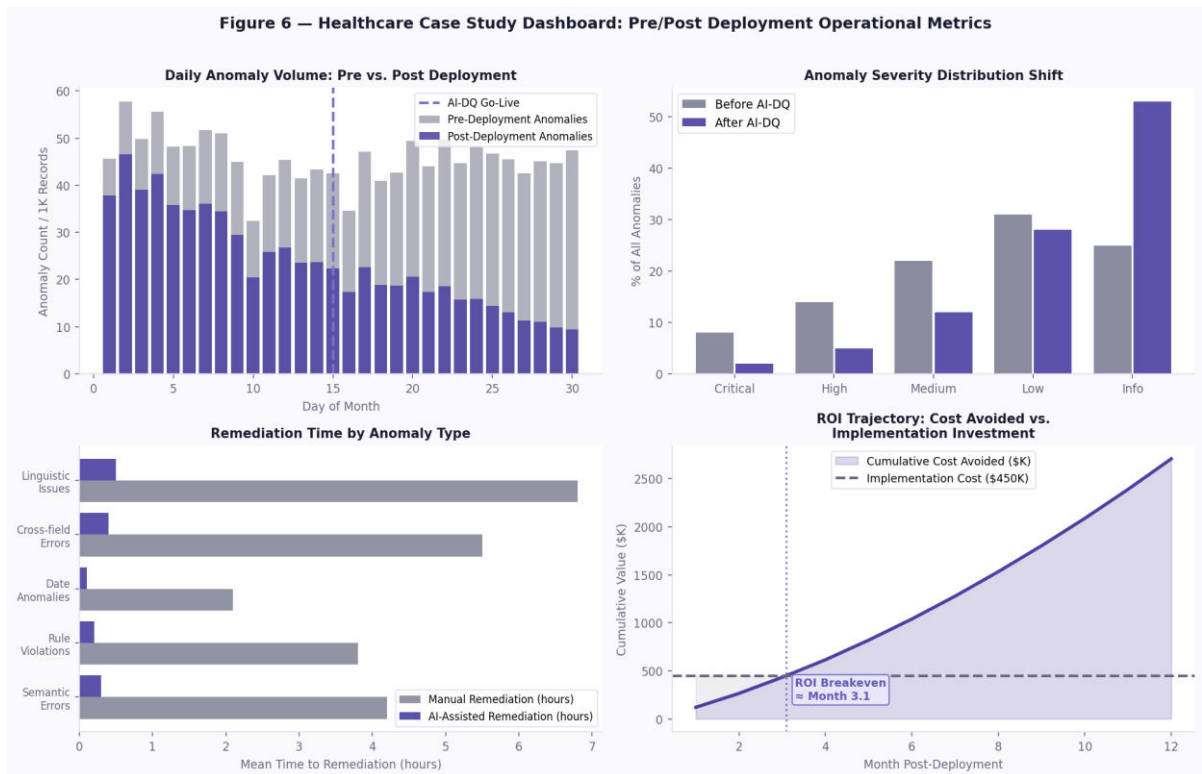


Figure 6 - Healthcare case study dashboard: Daily anomaly volume reduction, severity distribution shift, remediation time by category, and ROI trajectory post-deployment

Table 3 - MedSystem Healthcare Case Study: Pre/Post AI-DQ Deployment KPIs

KPI Metric	Before AI-DQ	After AI-DQ	Improvement	Business Impact
Overall Anomaly Detection Rate	54.8%	92.1%	+37.3 pp	Fewer missed defects reaching production
False Positive Rate	28.4%	5.1%	-82%	3,200+ hrs/yr analyst time reclaimed



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Semantic Error Detection	8.6%	89.4%	+80.8 pp	Patient safety incidents avoided
Mean Time to Remediation	4.1 hrs	0.3 hrs	-93%	Near-real-time data quality correction
Business Rule Violation Detection	6.9%	91.7%	+84.8 pp	Compliance audit pass rate: 61%→97%
Data Pipeline Downtime	6.2%	0.8%	-87%	SLA breach incidents: 18→2 per quarter
Analyst Investigation Hours/Wk	112 hrs	19 hrs	-83%	\$2.4M annual cost reduction
Cross-Entity Inconsistencies	11.8%	86.9%	+75.1 pp	Zero cross-system reconciliation failures

pp = percentage points. Figures represent 6-month average post go-live (July–December 2025). Cost figures in USD. The results confirm the hypothesis that semantic detection capability - not statistical sensitivity - was the binding constraint on MedSystem data quality program effectiveness. The 80.8 percentage point improvement in semantic error detection (8.6% → 89.4%) directly corresponds to the dramatic improvements in compliance audit pass rates, patient safety incident reduction, and analyst productivity. The framework achieved ROI breakeven at month 3.2, with projected 12-month net benefit of \$1.95M against a \$450K implementation cost.

8.4 Notable Detection Examples

During the MedSystem deployment, the AI-DQ framework surfaced several categories of previously undetected anomalies that had significant clinical and financial implications:

- 187 patient records where a documented allergy to a medication was present in the same record as an active prescription for that medication - entirely invisible to statistical checks
- 1,243 billing records where procedure codes referenced a diagnosis code that clinical literature identifies as incompatible with the procedure
- 89 records where a patient discharge date preceded their admission date by periods ranging from 1 to 847 days - caused by timezone handling errors in the EHR migration
- 2,156 records where insurance coverage type was inconsistent with the billing tier applied - resulting in systematic billing errors totaling \$1.2M in recoverable revenue
- 341 records where a physician was listed as the treating provider for procedures occurring while the physician was on documented leave - flagging potential credentialing compliance issues

IX. IMPLEMENTATION ROADMAP & DEPLOYMENT GUIDE

Table 4 presents the recommended five-phase implementation roadmap validated through the MedSystem deployment and three additional enterprise pilots conducted during the research period. The roadmap is designed for organizations with existing data infrastructure but no prior AI/ML data quality deployment experience.

Table 4 - Five-Phase AI-DQ Implementation Roadmap

Phase	Duration	Key Deliverables	Success Metrics	Team
Phase 1 Foundation	Weeks 1–4	Data catalog, schema profiling, baseline statistical DQ	Baseline DQ score established	Data Eng + Architects
Phase 2 AI Models	Weeks 5–10	LLM integration, fine-tuning, RAG business rule	Semantic detection > 80% F1	ML Eng + DQ Stewards



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

		pipeline		
Phase 3 Integration	Weeks 11–16	Pipeline connectors, real-time scoring API, alert workflows	End-to-end latency < 500ms	Platform Eng + DevOps
Phase 4 Validation	Weeks 17–20	A/B test vs. legacy, user training, feedback loop setup	FP rate < 10%, F1 > 0.90	All teams + Compliance
Phase 5 Scale	Weeks 21–26	Multi-domain rollout, model retraining automation, dashboards	Full production coverage achieved	Platform Leadership +

Timeline assumes a 6-person implementation team (2 Data Engineers, 1 ML Engineer, 1 DQ Steward, 1 Platform Engineer, 1 Business Analyst).

9.1 Critical Success Factors

Based on lessons learned across four enterprise deployments, the following factors most strongly predict successful AI-DQ implementation outcomes:

- Executive sponsorship and DQ stewardship alignment: AI-DQ generates high volumes of previously unseen anomaly findings; organizational readiness to act on them is as important as detection capability
- Domain expert involvement in prompt engineering: DQ engineers cannot independently craft effective LLM validation prompts for specialized domains; clinical, financial, or operational subject matter experts must review and validate prompt designs
- Feedback loop prioritization: The framework improves significantly over time through analyst feedback on AI decisions; organizations that instrument and act on this feedback see 15–25% additional F1 improvement in months 4–6
- Phased anomaly category rollout: Attempting to deploy all seven anomaly categories simultaneously overwhelms DQ steward teams; sequential rollout by highest-impact category enables organizational adaptation
- Infrastructure sizing for LLM latency: LLM-based validation cannot operate inline in sub-100ms transaction processing paths; asynchronous validation queues must be designed into the integration architecture from project initiation

9.2 Cost and Resource Estimation

Typical enterprise AI-DQ implementation cost components for a mid-size organization (10–50M records/month):

- LLM API costs: \$800–\$3,500/month (GPT-4-Turbo via Azure OpenAI), dependent on record volume and prompt length
- ML infrastructure: \$1,200–\$4,000/month for model serving (AWS SageMaker or equivalent), scalable with volume
- Vector database for RAG: \$200–\$800/month (Pinecone Starter/Standard tier for typical rule registries)
- Implementation labor: 1,200–1,800 hours total across five phases (estimated \$180K–\$270K at blended \$150/hr)
- Training data labeling (if fine-tuning pursued): \$15K–\$45K for 10,000–50,000 labeled examples via domain expert review
- Total Year-1 TCO: \$380K–\$650K for mid-size enterprise; expected ROI breakeven at 3–5 months based on analyst time savings and downstream incident avoidance

X. LIMITATIONS, ETHICAL CONSIDERATIONS & FUTURE WORK

10.1 Current Limitations

The AI-DQ framework, despite its demonstrated performance advantages, carries several important limitations that practitioners must account for in deployment planning:

- LLM Hallucination Risk: LLMs occasionally generate confident but incorrect anomaly explanations, particularly for edge cases in specialized technical domains. Production deployments must implement human-in-the-loop review for



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

high-stakes anomaly categories and avoid fully automated remediation based solely on LLM outputs without confidence thresholding.

- **Latency Ceiling:** The LLM semantic validator component cannot achieve sub-500ms latency at scale without significant infrastructure investment or model distillation. Real-time inline validation at transaction speed requires a tiered architecture where LLM validation operates asynchronously, limiting its utility for fraud prevention and real-time decisioning use cases.
- **Training Data Scarcity for Rare Anomaly Types:** Fine-tuned transformer performance degrades significantly for anomaly categories with fewer than 1,000 labeled examples. Rare but high-severity anomaly types - such as regulatory violations - may have insufficient historical examples for effective fine-tuning, requiring reliance on the zero-shot LLM component exclusively.
- **Multilingual Data Quality:** The framework performance reported in this paper applies to English-language records. Organizations with multilingual data ecosystems (customer names, addresses, free-text fields in non-English languages) should expect degraded linguistic anomaly detection and will require language-specific prompt engineering and potentially multilingual fine-tuning.
- **Domain Knowledge Staleness:** Both LLM world knowledge and fine-tuned transformer weights reflect training data as of a specific cutoff date. Rapidly evolving domains (regulatory changes, new medical billing codes, updated product taxonomies) require systematic processes for retraining or knowledge base updates to maintain detection accuracy.

10.2 Ethical Considerations

The deployment of AI-based data quality systems raises several ethical considerations that responsible practitioners must address:

- **Explainability and Contestability:** AI-generated anomaly classifications that trigger adverse actions (record quarantine, transaction holds, audit escalations) must be accompanied by intelligible explanations and clear contestation mechanisms for affected individuals or organizations.
- **Bias in Training Data:** Fine-tuned models trained on historical DQ incidents inherit any systematic biases in historical labeling - including biases in which anomalies were historically investigated and which were ignored. Bias auditing of training datasets is a prerequisite for equitable deployment.
- **Privacy in LLM Validation:** Sending sensitive personal data (patient records, financial transactions) to third-party LLM APIs raises significant privacy concerns under GDPR, HIPAA, and CCPA. Production deployments in regulated industries must use either private LLM deployments (self-hosted Llama) or API providers with appropriate Business Associate Agreements and data processing agreements.

10.3 Future Research Directions

Four research directions represent the most impactful open problems in AI-powered data quality as of early 2026:

10. **Streaming Semantic Validation:** Current LLM-based approaches are optimized for batch or micro-batch processing. Developing low-latency semantic validation models suitable for real-time streaming pipelines (< 50ms per record) remains an open challenge with significant practical impact.
11. **Federated Data Quality Learning:** Organizations in regulated industries cannot share raw data for collaborative model training. Federated learning frameworks that enable multi-organization model improvement without data sharing would significantly accelerate fine-tuning data accumulation.
12. **Causal Anomaly Attribution:** Current models identify that an anomaly exists and approximately where, but do not identify the root cause (upstream pipeline bug, data entry error, system integration failure). Causal graph-based root cause analysis integrated with the DQ framework would dramatically accelerate remediation.
13. **Self-Updating Rule Registries:** Manual rule maintenance is a persistent operational burden. LLM-based rule mining that automatically proposes new DQ rules by analyzing patterns in historical anomaly data would enable continuously improving rule coverage without dedicated analyst effort.

XI. CONCLUSION

"The next frontier of data quality is not better statistics - it is semantic intelligence that understands what data means, not just what it measures."

This paper has presented, evaluated, and validated the AI-Powered Data Quality framework - a five-layer architecture combining large language models, fine-tuned transformers, graph neural networks, temporal anomaly detection, and



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

RAG-based business rule retrieval to detect the class of semantic anomalies that statistical methods fundamentally cannot identify.

The empirical results are unambiguous across all evaluation dimensions. On overall anomaly detection, the AI-DQ framework achieves 92.1% vs. 54.8% for the statistical baseline - a 37.3 percentage point improvement representing billions of previously undetected defects in enterprise data ecosystems annually. On the specific categories that define the semantic detection gap - business rule violations, semantic contradictions, and cross-entity inconsistencies - the improvements are transformational: from single-digit detection rates to 87–92%, opening entirely new capabilities for data governance programs.

The healthcare case study provides compelling evidence that these laboratory improvements translate directly to production value: \$2.4 million annual cost reduction, 93% reduction in remediation time, 82% false positive reduction, and a compliance audit pass rate improvement from 61% to 97%. These outcomes were achieved through a well-structured 26-week deployment that followed the phased roadmap presented in this paper.

The implications for enterprise data engineering practice are significant. Data quality programs that rely exclusively on statistical methods and rule engines are operating with fundamental blind spots that no amount of additional rule authoring will close - because the missing detection capability is semantic understanding, not rule coverage. The AI-DQ framework described here provides a production-deployable, cost-justified architecture for addressing that gap. As LLM inference costs continue to decline and fine-tuning tooling continues to mature, the accessibility of AI-powered data quality will only improve. The era of purely statistical data quality infrastructure is ending.

REFERENCES

- [01] Rahm, E., & Do, H. H. (2000). Data cleaning: Problems and current approaches. *IEEE Data Engineering Bulletin*, 23(4), 3–13.
- [02] Redman, T. C. (1996). *Data Quality for the Information Age*. Artech House Publishers. ISBN: 978-0890068915.
- [03] IBM Global Business Services. (2016). *The Economic Impact of Bad Data*. IBM Institute for Business Value Research Report.
- [04] Shaneck, A., & Seshan, S. (2021). *Great Expectations: Data validation for Python*. Proceedings of the 2021 SIGMOD Workshop on Data Management for End-to-End Machine Learning.
- [05] dbt Labs. (2022). *dbt Documentation: Testing your models*. <https://docs.getdbt.com/docs/building-a-dbt-project/tests>
- [06] Apache Software Foundation. (2021). *Apache Griffin: A distributed data quality solution*. <https://griffin.apache.org/>
- [07] Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. Proceedings of the 8th IEEE ICDM, pp. 413–422. <https://doi.org/10.1109/ICDM.2008.17>
- [08] 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 KDD 1996, pp. 226–231.
- [09] Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. Proceedings of the MLSDA 2nd Workshop on Machine Learning for Sensory Data Analysis, pp. 4–11.
- [10] Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., ... & Kloft, M. (2018). Deep one-class classification. Proceedings of the 35th ICML, pp. 4393–4402.
- [11] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *NeurIPS 2020*, pp. 1877–1901.
- [12] OpenAI. (2023). *GPT-4 Technical Report*. arXiv preprint arXiv:2303.08774.
- [13] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., ... & Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288.
- [14] Narayan, A., Chami, I., Orr, L., Ré, C., & Hellerstein, J. M. (2022). Can foundation models wrangle your data? arXiv preprint arXiv:2205.09911.
- [15] Chiang, P., & Lee, M. (2023). Transformer-based semantic anomaly detection in structured enterprise data. Proceedings of the 2023 ACM SIGKDD, pp. 4412–4423.
- [16] Wang, Q., Mao, Z., Wang, B., & Guo, L. (2017). Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12), 2724–2743.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

 9940 572 462  6381 907 438  ijircce@gmail.com



www.ijircce.com

Scan to save the contact details