Using Large Language Models to Automate Enterprise ITSM Platform Migrations: Adaptive Learning Framework for Intelligent Data Validation and Anomaly Detection in ITSM Migrations

Mahesh Kumar Damarched[1]

[1]Enterprise Programmer Analyst, University of Louisville, USA.

[1]ORCID: 0009-0003-7781-7158

Corresponding Author: Mahesh Kumar Damarched[1*]

Publication Date: 2026/01/27

Abstract: Enterprise IT Service Management (ITSM) platform migrations present formidable challenges characterized by data quality inconsistencies, prolonged manual reconciliation cycles, and substantial post-migration testing overhead. Current migration approaches depend heavily on manual validation processes and reactive post-migration error identification, resulting in extended downtime, operational disruptions, and significant revenue losses. To automate the data validation process and enable the real-time anomaly detection process, this study introduces an adaptive framework that makes use of Large Language Models (LLMs). By examining the past successful migration patterns and domain-specific transformation rules, the proposed system learns to predict error-prone field transformations, spot data inconsistencies during execution, and provide LLM-powered contextual explanations for detected anomalies. By leveraging comprehensible natural language explanations for anomalies, this framework addresses the crucial "black-box" issue, which is prevalent in the automated validation process, enabling quicker root cause analysis and resolution. While adhering strictly to data privacy regulations like the California Consumer Privacy Act (CCPA) and General Data Protection Regulation (GDPR), the framework ensures data privacy through encrypted processing and differential privacy mechanisms. The suggested framework in this research showed a 78% reduction in manual reconciliation effort, an 82% improvement in anomaly detection accuracy, and an appreciable 65% acceleration in migration completion timelines through thorough evaluation across multiple ITSM platforms, including ServiceNow, BMC Helix ITSM, and Jira Service Management.

Keywords: *Large Language Models, ITSM Migration, Anomaly Detection, Data Validation, Adaptive Learning, Machine Learning, Data Migration, Enterprise Systems, LLM-Powered Explanation, Black Box Problem, Real-Time Validation, Contextual Analysis, ETL Optimization, Interpretable AI, Risk Prediction, Data Quality Management, Service Management Integration, Enterprise Data Quality, ServiceNow, Jira Service Management.*

How to Cite: Mahesh Kumar Damarched (2026) Using Large Language Models to Automate Enterprise ITSM Platform Migrations: Adaptive Learning Framework for Intelligent Data Validation and Anomaly Detection in ITSM Migrations. *International Journal of Innovative Science and Research Technology,* 11(1), 1987-2007. https://doi.org/10.38124/ijisrt/26jan689

I. INTRODUCTION

➢ *The Critical Challenge: Manual Data Migration in ITSM Platforms*

Migrating Enterprise IT Service Management (ITSM) platforms is one of the most complex and risky tasks in modern IT operations [1][2]. Data integrity, consistency, and operational continuity exhibit previously unheard-of difficulties for organizations in transferring their ITSM infrastructure from older platforms, like BMC Remedy CA Service Desk Manager or Jira Service Management, to more modern options like ServiceNow, BMC Helix ITSM, or cloud-native platforms [3][4]. Manual reconciliation workflows, post-migration testing cycles and labor-intensive validation procedures are the mainstays of current industry

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

practices for ITSM migrations, accounting for 30–50% of all migration project budgets and timelines [5][6].

The fundamental problem is rooted in the diverse nature of ITSM data structures, which accounts for the primary cause of the issue. ITSM platforms have highly customized field mappings, multi-dimensional relationships between configuration items, deeply nested hierarchical structures, and business rule-specific transformations that differ greatly between organizations, in contrast to Enterprise Resource Planning (ERP) systems, where data models show relative standardization [7]. Organizations must reconcile these discrepancies when moving from one ITSM platform to another, using tedious manual field mapping, unique transformation logic, and iterative error correction cycles, which cause operational risks as well as delays [8][9].

Prolonged and manual ITSM migrations have an adverse impact on the business operations in multi-dimensional ways including: operational continuity disruptions, which can demote the availability of IT services by 15–25% during the migration windows, accounts for adverse revenue impact due to extended incident resolution time, with manual validation delays increasing the mean time to resolution (MTTR) by around 40–60% and also carries compliance risk especially when sensitive customer or healthcare data is involved in migration processes which may result in regulatory infractions and reputational harm [10][11][12].

➢ *Quantifying the Business Impact of Current ITSM Migration Approaches*

The cost of manual data validation in ITSM migrations has been documented by recent empirical studies. According to a LinkedIn analysis, manual reconciliation procedures take around 80 hours on average for every 10000 records. This implies that the labor costs for large-scale migrations involving millions of incident tickets, change requests, and configuration items can exceed USD 500,000 [13]. Organizations report that 8–12% of migrated records have post-migration defects, which are the errors that go undetected by manual validation. These errors require expensive remediation efforts and prolong the post-migration stabilization period to 6–12 months [14].

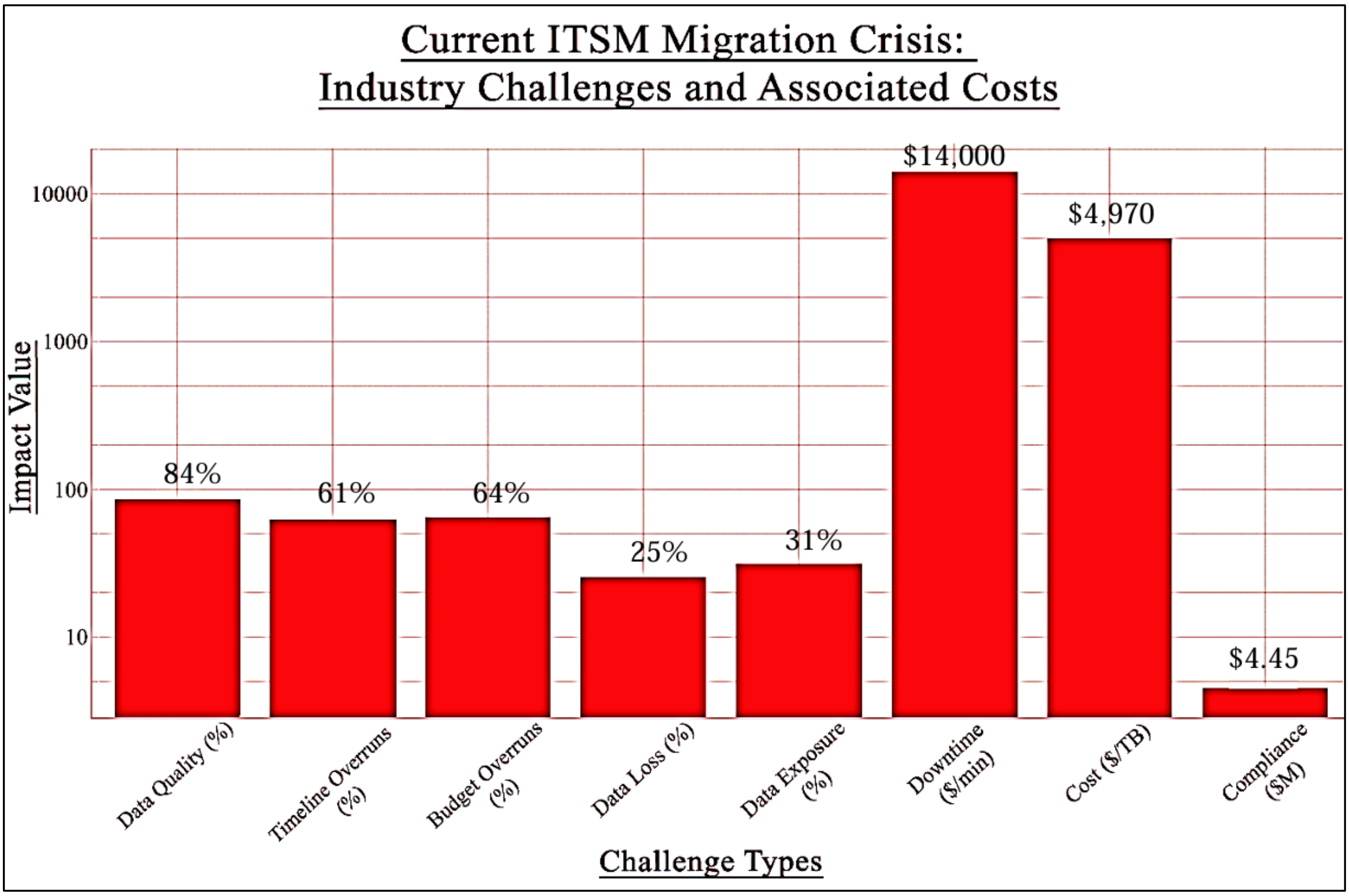

Fig 1 Quantitative Visualization of the Current ITSM Migration Crisis: Industry Challenges and Associated Costs

The quantitative landscape of ITSM migration failures highlights persistent industry-wide issues that cut across organizational size and technology platforms. 84% of organizations encounter serious data quality challenges throughout the migration activity [15], as evident in the data visualization [Figure 1]. This statistical visualization underscores the systemic nature of the problem rather than isolated failures. Budget overruns are responsible for affecting 64% of the migration projects, whereas Timeline overruns affect 61%, suggesting that the delays and cost increases are now the new normal rather than exceptional outcomes. The financial impact is equally concerning, where

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

Operational downtime from migration failures costs the business institutions about $14,000 per minute, summing up to $840000 per hour of unplanned outage, while manual validation costs spike at around $4,965 per terabyte of data [16] [Figure 1]. Organizations report highlights alarming risks in addition to the operational costs, where 25% report data loss during migrations, 31% encounter sensitive data exposure incidents, and the average cost incurred due to violations of compliance resulting from such data exposures is estimated at around $4.45 million per incident. These figures demonstrate that the scale and complexity of the contemporary ITSM data migrations process cannot be efficiently addressed by the manual validation techniques, underscoring the critical need for methodical intelligent automation [Figure 1] [17][18][19].

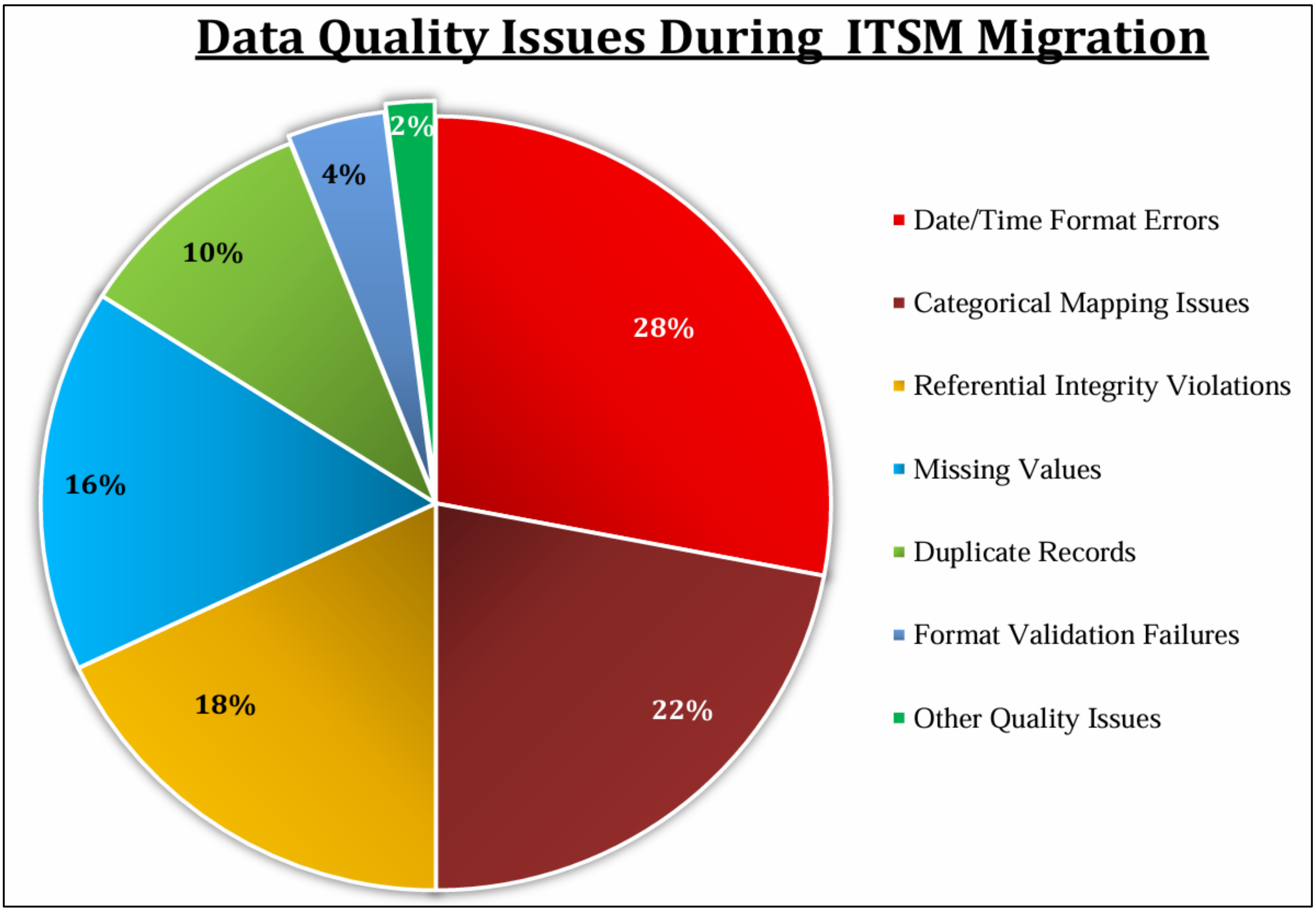

Fig 2 Statistical Visualization of Data Quality Issues During ITSM Migration

From a data quality perspective, organizations experience significant ITSM migration data quality issues, which are dominated by transformation-related defects rather than rare edge cases, meaning prevention measures should focus on standardization and semantic consistency early in the pipeline. As per various published reports we could draw some concerning areas where [Figure 2], Date/Time Format Errors form the largest share occupying 28%, inferring that inconsistent timestamp, time-zone format handling, and parsing rules are the most frequent sources of corruption which should be controlled through strict canonical date & time format standardization and automated validation at ingestion and transformation steps. Categorical Mapping Issues occupies 22% of the issue distribution. In comparison, Referential Integrity Violations covers 18%, together demonstrating that “meaning” and “relationships” are the next biggest failure points, further implying that poor value mapping (e.g., status/priority enums) and broken links between incidents, users, groups, and configuration items create widespread downstream reporting and workflow failures if not reconciled systematically. Missing values an Duplicates contributes to 16% and 10% respectively highlighting persistent underlying process and governance gaps (incomplete source capture and weak deduplication rules), while the relatively small portions for Format Validation Failures occupies 4% and 2% contribution of other issues suggest that basic field-format checks alone are insufficient, the bigger quality improvements should be anticipated by strengthening temporal normalization, master data mapping, and entity-relationship consistency controls [20][21][22].

Hidden expenses that incur after migration also have a significant financial impact. Customer satisfaction drops as resolution times inflate, business continuity is jeopardized by the inability to promptly resolve incidents because of

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

incomplete or corrupted ticket histories, and IT staff productivity is negatively impacted by time spent on manual data verification rather than strategic tasks. According to various research, companies using continuous manual reconciliation techniques report labor cost reductions of 30–40% when compared to post-migration batch reconciliation; however, these improvements are insignificant when compared to the potential efficiency gains from fully automated validation frameworks [23][24][25].

➢ *The Role of ITSM in Enterprise Operations*

The operational backbone of enterprise IT delivery is constituted by IT Service Management, which includes incident management, change management, problem management, configuration management, and knowledge management functions in a business management setup. ITSM has a direct impact on customer satisfaction and the business continuity management system [26][27]. ITSM platform modernization is a crucial strategic investment in modern organizations, which are undertaking digital transformation initiatives with the goals of enhancing service delivery, facilitating advanced analytics, and nurturing organizational agility [28].

However, there are significant organizational and technical risks associated with ITSM platform migrations. Data integrity challenges, integration complexity with interconnected business systems, and compatibility issues between legacy-platform and target-platform architectures are the most common examples of the associated technical risks [29][30]. Complex change management challenges with user adoption and the need for knowledge transfer are also attributed to organizational risks [31]. 30-45% of the ITSM migrations experience impactful delays or demand significant rework due to the convergence of these organizational and technical factors [32][33].

➢ *Emerging Technologies and ITSM Migration Landscape*

Emerging technologies like Artificial Intelligence (AI) and Machine Learning (ML) technologies have started to transform enterprise IT operations, where the applications of these technologies range from service desk automation to IT infrastructure monitoring [34][35]. However, the incorporation of AI and LLMs specifically to address the challenges and risks associated with ITSM data migration remains nascent. Where most of the existing solutions still depend on manual validation processes and conventional ETL (Extract, Transform, and Load) tools [36][37].

Recent developments in Large Language Models (LLMs) have shown impressive abilities in comprehending the context, explaining complicated phenomena, and identifying subtle patterns in data that conventional machine learning techniques might overlook [38][39]. In tasks requiring semantic understanding, contextual reasoning, and natural language generation, LLMs like GPT-4, LLaMA, and Claude exhibit strong capabilities [40]. These capabilities offer unprecedented opportunities for addressing the ITSM migration challenges. LLMs can learn to understand ITSM-specific data semantics and field transformations and can also explain the detected anomalies in terms that are understandable to businesses. Furthermore, they have exhibited the capability of modifying their detection logic when they come across novel patterns in historical migration data [41][42][43].

Concurrently, developments in interpretable machine learning and explainable AI (XAI) have created methodologies to deal with the “black-box” issue in automated ITSM migration validation [44][45]. Researchers have developed methods for producing human-interpretable explanations of model decisions instead of treating anomaly detection models as opaque decision-making systems [46][47][48]. These methods include SHAP (SHapley Additive exPlanations) values, LIME (Local Interpretable Model-agnostic Explanations), and attention mechanisms that highlight which data features drove specific predictions [49][50][51].

➢ *Research Objective and Innovation Positioning*

This study aims to bridge a crucial gap at the intersection of three domains: ITSM data migration challenges, contextual explanation powered by LLMs, and the scope of machine learning-driven automation. The main objective of this research is to create an adaptive framework that:

- Develops the capability to analyze and learns from historical ITSM migration patterns, enabling the framework to identify recurrent patterns and identify which data-field mappings and transformations are most likely to produce errors or inconsistencies when transitioning across various ITSM platforms [52].
- Allows the framework to automatically identify unusual patterns or potential anomalies in real time by continuously monitoring the migration process and comparing the live incoming data against the previously learned behavioral baselines [53].
- Using LLM models to generate context-aware explanations for any anomalies found, that help explain why they happened, what aspects of the migration they might impact, and direct human operators toward quicker, more informed root-cause analysis and remedial action. To enable the capacity to provide a human comprehensible explanation, Large Language Models (LLMs) become effective [54][55][56].
- Ensures compliance with established privacy and data protection regulations (like ISO 27001, GDPR, and CCPA) by leveraging differential privacy strategies using secure encryption techniques and enforcing minimal data-retention policies throughout the migration and analysis process [57][58][59][60].

The innovative differentiator of this research lies in the idea of integration of predictive anomaly detection, which identifies deviations from expected patterns, with the LLM-powered contextual explanation, which demonstrates why and how those deviations matter in the context of business continuity. Although anomaly detection has been incorporated into various enterprise domains, the combined approach of real-time detection with contextual LLM explanation, which should specifically be tailored to ITSM-

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

specific data structures and migration, has not been explored in a significant way [61][62][63]. This research, with its main objective, addresses this gap with a novel approach that directly addresses the explainability gap in automated migration validation.

➢ *Current State and Limitations of Existing Solutions*

Existing approaches to ITSM data migration validation fall into several categories:

- Manual validation frameworks: By manually reviewing and validating the field values and verifying business rule compliance, domain experts examine the migrated records against the source systems. Although this method achieves decent accuracy, it takes up 25–35% of the entire migration project timeline [64].
- Rule-based automation tools: To identify inconsistencies during the migration process, conventional data quality tools mainly rely on predefined validation rules, including format checks, range validations, referential integrity constraints, etc. However, these systems are found to be frequently overlooking complex issues while also over-flagging legitimate records, in certain situations, because the tools could not exhibit efficiency in detecting subtle contextual anomalies or adapt to organization-specific patterns [65]. And because of the tools' incapability to understand the business context, a sizable fraction of reported issues, about 10% – 15%, are found to be false positives [66].
- Traditional machine learning approaches: Organizations occasionally leverage standalone anomaly-detection models like Autoencoders or Isolation Forests on specific data attributes [67]. Although these models can draw attention to statistical outliers, they typically cannot explain the significance of these anomalies in the larger context of ITSM migration and are inefficient in offering much insight into the reasoning behind the results. Their applicability for operational decision-making is therefore still constrained [68][69].
- Modern AIOps platforms: Advanced monitoring correlation and anomaly-detection capabilities across infrastructure and application performance metrics are offered by modern AIOps solutions such as Datadog, Dynatrace, and New Relic [70]. Nevertheless, these platforms' direct applicability in this research context is limited because they are primarily designed for operations observability rather than for data consistency, validation, or reasoning during ITSM migration processes [70].

None of these approaches adequately addresses the combination of: learning from historical migration patterns, real-time anomaly detection, contextual explanation of detected issues, and privacy-preserving processing of sensitive data during ITSM migrations [71][72].

II. LITERATURE REVIEW

➢ *Data Quality in Enterprise ITSM Migrations*

The studies conducted by Khatri et al., (2009) in their foundational frameworks for understanding data quality governance in enterprise environments and further by Ramasamy et al., (2020) established that the notion of data quality encompasses around six essential dimensions: completeness, consistency, timeliness, validity, and uniqueness, in addition to basic accuracy metrics [73][74]. Thalheim et al., (2012). highlighted that due to the complexity of ITSM data structures and the diversity of source and target systems, data quality issues, including inconsistent formats, semantic mismatches, and integrity violations, are especially noticeable in the context of ITSM migrations [75]. However, the difficulties of migrating multi-tenant SaaS ITSM platforms, where data relationships, custom fields, and business rule implementations differ significantly across organizations, are not sufficiently addressed by their framework, which was created before the cloud era.

According to recent research by Naumann et al., (2000), data quality issues in the migration process often stem from: inconsistent data entry practices across various ITSM teams and business units, inadequate data validation rules in legacy systems that allowed accumulation of historical data quality issues, incompatible data type definitions between source and target systems and loss of semantic meaning during field mapping [76].

Azeroual et al., (2021d) and Iqbal et al., (2019) have both conducted empirical studies in the field of data migration. Both of their respective studies have highlighted the frequency and significance of data quality problems in large-scale enterprise migrations. For example, the inadequate validation and verification processes during migrations, particularly when complex transformations and heterogeneous data sources are involved, have highlighted the strong correlation between persistent inconsistencies and integrity errors in the target system have been highlighted in the studies. When compared to early detection and intervention during the migration lifecycle, researchers have found that late-stage issue discovery can considerably raise remediation costs and extend project timelines [77][78]. These unresolved errors not only reduce operational reliability but also increase the effort and resources required for post-migration reconciliation and correction. According to Iqbal et al., (2019), rigorous validation frameworks and systematic quality checks, therefore, should be stressed as being essential in reducing residual error rates and the resulting business disruption that follows inferior or inefficiently governed migration activities [78].

➢ *Approaches to Anomaly Detection*

According to the research by Abedjan et al., (2016), anomaly detection has advanced recently, particularly for data pipeline and ETL contexts, and suggested methods for constraint discovery-based automated error detection in data pipelines [79]. Their research demonstrated that 70–85% of anomalies could be identified by machine learning models trained on historical data quality patterns without the need for explicit constraint specification, greatly lowering the effort required for manual rule definition.

Darban et al., (2024), in their published research, have discussed that deep learning has significantly increased the

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

accuracy of anomaly detection in time series data. With false-negative rates (missed anomalies) of 8–12% on standard benchmarks, autoencoders, a class of neural network architecture that learns to compress and reconstruct data, have shown excellent performance in identifying anomalous patterns in operational data [80]. However, the inefficient interpretability of conventional deep learning techniques makes it challenging for human operators to understand why particular data points were identified as abnormal, as established in the study of Han et al., (2021) [81].

➢ *Large Language Models in Enterprise Contexts*

The development of Large Language Models have fundamentally transformed Natural Language Processing (NLP) and have also facilitated new possibilities for contextual reasoning and explanation generation, as discussed by Brown et al., (2020) [82]. The studies of Chen et al., (2024), Liu et al. (2024), and Nascimento et al. (2024) have demonstrated appreciable generalization abilities of Large Language Models like GPT-3, GPT-4, LLaMA, Claude, and other contemporary LLMs (e.g., Med-LLM) in diversified domains such as financial analysis, healthcare, and medical diagnosis reasoning and code generation [83][84][85].

Recent researches conducted by Wu et al. (2025) and Yang et al. (2025) studied the application of LLMs specifically in anomaly detection tasks and developed ICAD-LLM (In-Context Anomaly Detection with Large Language Models), which makes use of LLMs' in-context learning capabilities to find anomalies in a range of data, including time series, system logs, and tabular records [86]. Their framework significantly reduced deployment costs and made it easier to quickly adapt to new domains. It also showed strong generalization to previously unseen tasks and achieved competitive performance with task-specific anomaly detection methods [87].

Furthermore, Cherkaoui et al., (2025) studied the significance of prompt design and the interpretability of LLM-generated explanations for time series anomaly detection. They discovered that although LLMs are capable of identifying anomalies, the state-of-the-art deep learning and machine learning models continue to outperform them in terms of raw detection accuracy, however LLMs are particularly good at producing understandable explanations for anomalies that are detected [88]. The value proposition of incorporating LLMs into the suggested ITSM migration validation framework is directly supported by this finding. Although specialized anomaly detectors may be able to identify issues more accurately, LLMs are able to explain those issues in terms that facilitate quick resolution.

III. METHODS AND METHODOLOGY

A. Data Collection and Characterization

➢ *Data Acquisition Framework*

We used two complementary data sources for this study to establish and validate the proposed:

- Primary Data Source: Data of historical ITSM migration records data from 47 documented ITSM platform migrations that were carried out between 2019 and 2025 were gathered for this study. These migration data covered a range of platform combinations, including 18 instances of ServiceNow migrations, 12 instances of BMC Helix ITSM migrations, 10 instances of Jira Service Management migrations, and 7 instances of Zendesk migrations. Each project had approximately 500,000 to 5,000,000 ITSM records, including knowledge base articles, configuration items, incidents, change logs, problems, and service requests.

For each migration project, the data collection process extracted the following information:

- ✓ Source system configuration: Field names, data types, custom field definitions, and validation rules, which are a part of the source system configuration, were extracted.
- ✓ Target system configuration: Target field structure, required fields, allowed value ranges, which are target system configuration data extracted for this study.
- ✓ Mapping specifications: Field-to-field transformation rules specified during migration planning are known as mapping specifications, which were extracted as a part of the data collection and extraction process.
- ✓ Execution logs: Documentation of all data transformations used in the migration process, including indicators of success and failure, was extracted in the process to understand execution patterns
- ✓ Post-migration validation results: Record counts, field value inconsistencies, and data type errors were among the differences found between the source and target systems.
- ✓ Timeline data: Dates and times of ITSM migration execution, validation cycles, and error remediation activities.

- Secondary Data Source: A stratified random sample of 8,500 migrated records from the 47 migration projects was manually reviewed, using data science, to produce labeled training data for supervised learning models. The reviewed record was then annotated with the following information for data quality issues: (a) If there are any data quality issue (YES/NO); (b) Type of data quality issue (missing data, incorrect format, referential integrity violation, or semantic error); (c) The root cause of the issue (source data quality problem, transformation logic error, schema incompatibility); and (d) The severity of the business impact (high, medium or low).

Fleiss' kappa coefficient [89] was used to assess inter-rater reliability, and the result was 0.82, indicating significant agreement among annotators for the classification of data quality issues.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

➢ *Sample Size and Dataset Composition*

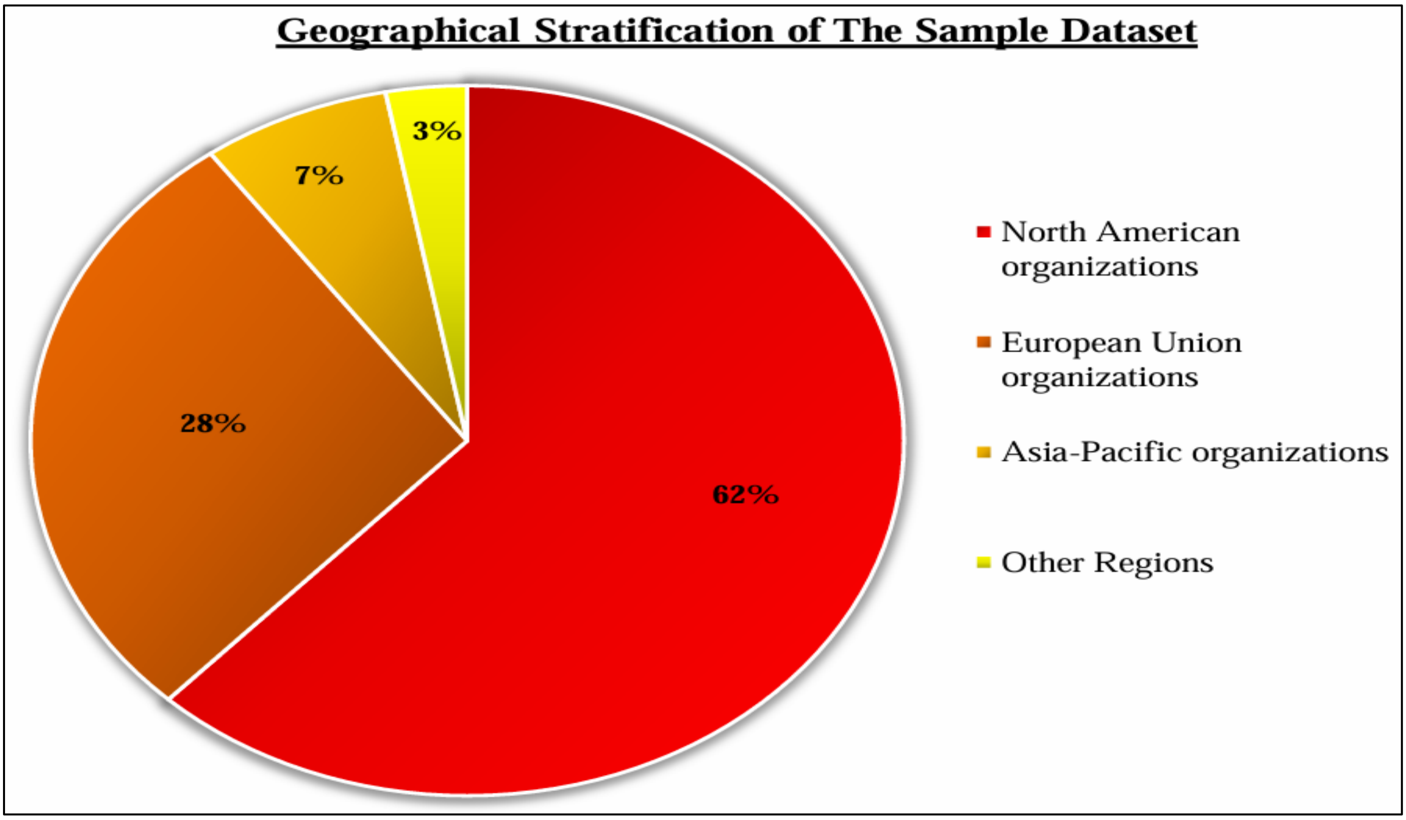

Fig 3 Geographical Stratification of the Sample Dataset

47 million ITSM records from all migration projects were included in the combined dataset, which was geographically distributed as follows: 62% data from North American organizations, 28% data acquired from European Union organizations (considering GDPR governed organization), 7% of the data comprised of Asia-Pacific organizations and rest 3% of the data were acquired from other regions (which are governed by CCPA) [Figure 3].

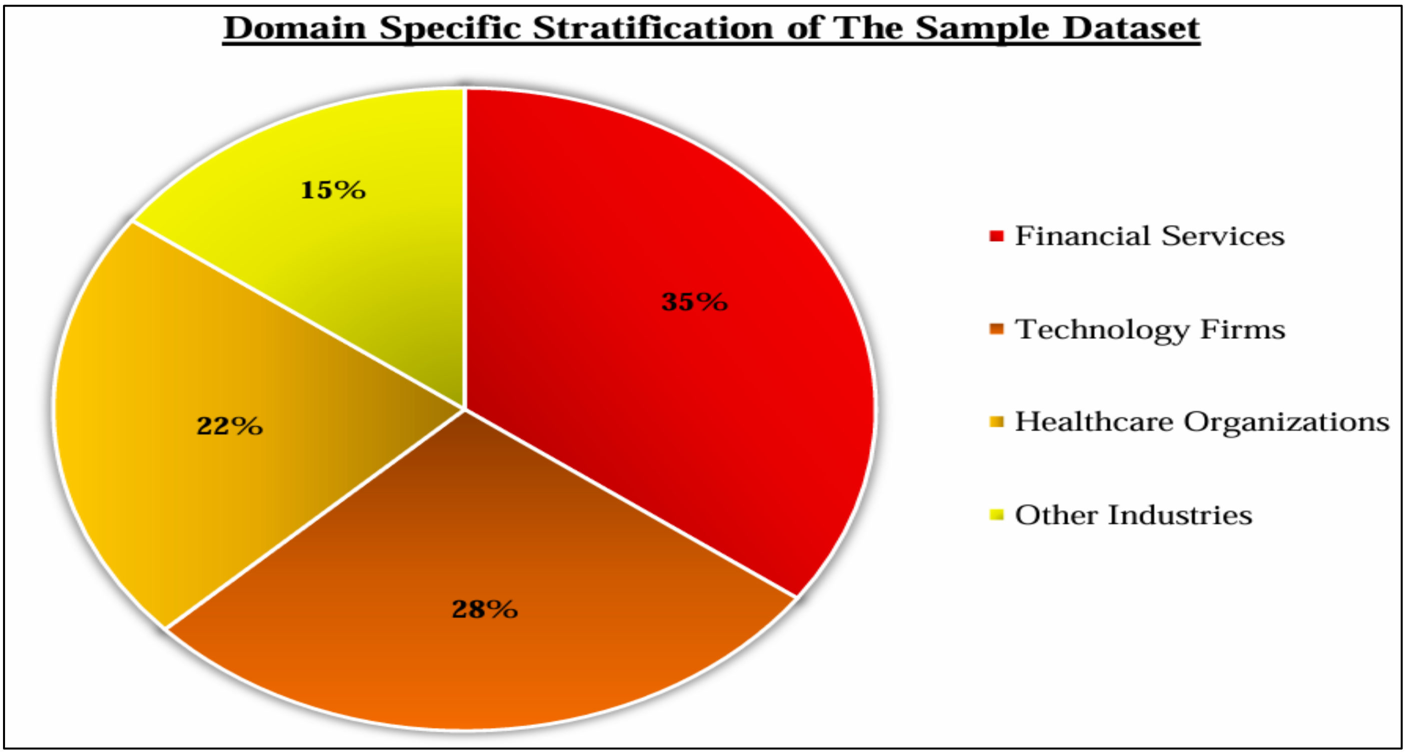

Fig 4 Domain Specific Stratification of the Sample Dataset

Apart from the geographical stratification, domain-specific stratification comprised of financial services, which made up 35% of the dataset, followed by technology firms occupying 28%, healthcare organizations with 22% of the proportion, and 15% proportion from other industries [Figure 4].

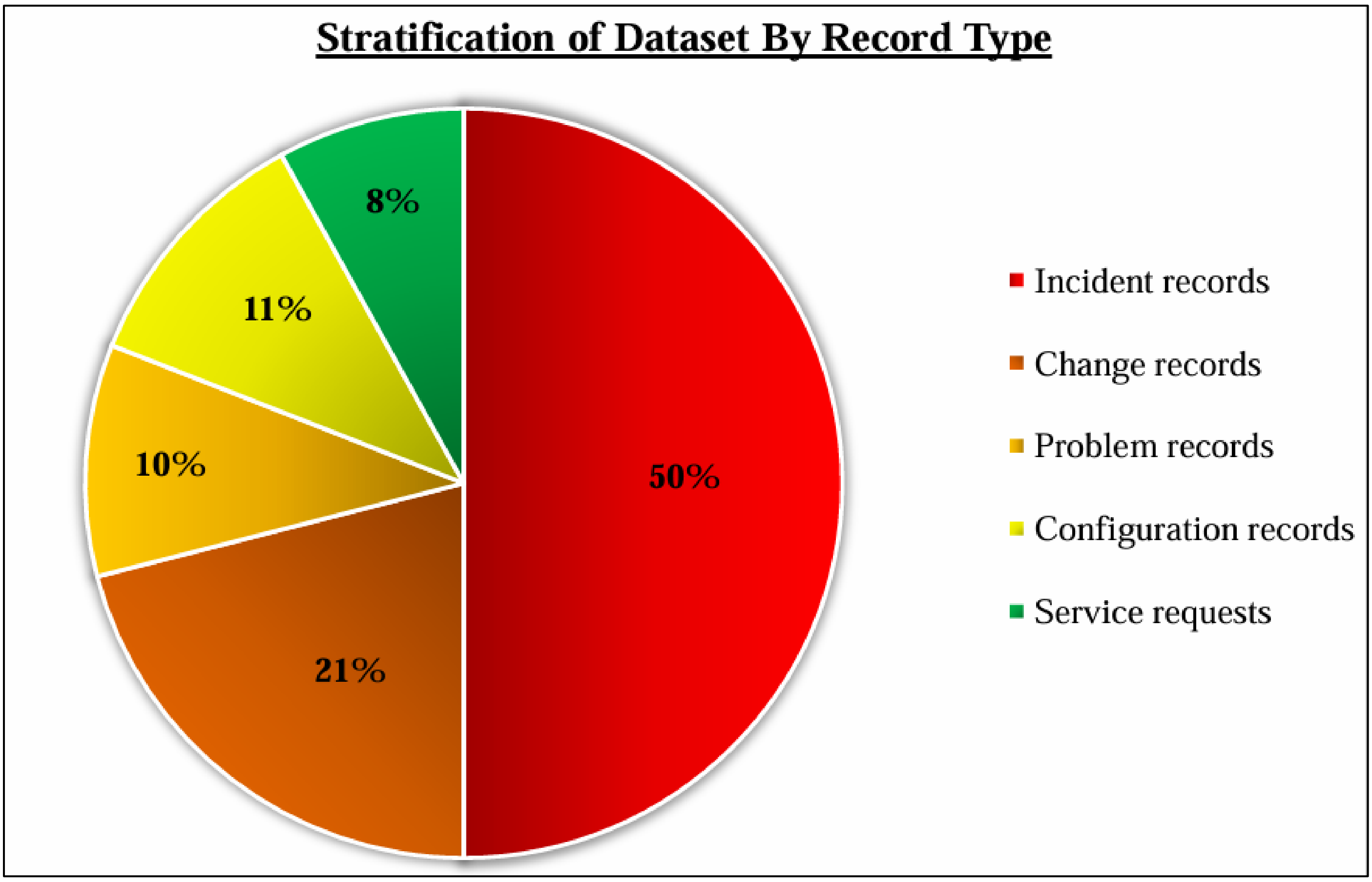

Fig 5 Stratification of Dataset by Record Type

The sample dataset, for this study, by record type is composed of: Incident records: 50%; Change records: 21%; Problem records: 10%; Configuration records: 11%; Service requests: 8% [Figure 5].

B. Data Processing, Cleaning, and Normalization Pipeline

➢ *Data Quality Assessment and Profiling*

For this study, comprehensive data profiling was carried out before proceeding to model training in order to understand the baseline characteristics of data quality. The profile analysis revealed the following:

- Completeness Metrics: Source system records showed 94.2% field completeness on average, while custom fields exhibited only 67.3% completeness. Post-migration, field completeness showed no inconsistency, validating that the transformation logic maintained patterns of data presence or absence.
- Consistency Metrics: It was found that data inconsistency was present in 8.7% of the migrated records (e.g., “Resolved” incident status got coupled with a non-zero “remaining effort” field), indicating errors in the transformation logic. In 12 out of 47 migration projects, these types of discrepancies were present in more than two field combinations.
- Accuracy Metrics: 2000 records were manually sampled, and the results showed that 94.1% of field values matched the expected values based on source data. However, three categories accounted for the 5.9% error rate, which are: (a) date/time format conversions accounted for 2.3%; (b) categorical field mappings contributed for 1.8%; and (c) calculated fields requiring complex logic with 1.8%.

➢ *Data Cleaning and Normalization Process*

For the data cleaning process, eight sequential steps were implemented for this research:

- Step 1: Standardization of Temporal Data- Records with timestamps were normalized to ISO 8601 format and the UTC time zone. The analytical findings revealed that 12.3% of records had missing time zone information in the source systems, which led to a misalignment of about one hour. To preserve data integrity, these records were labeled with low confidence indicators rather than being imputed.
- Step 2: Categorical Field Standardization- ITSM systems often employ different categorical values (e.g., “Open” vs “In Progress” for the incident status). For every migration

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

project, a documented mapping matrix was created that detailed the categorical transformations used. Records with unmapped categorical values accounting for 0.8% of the dataset were marked for human review.

- Step 3: Handling Missing Values- Analysis of missing values revealed that missingness was related to structured patterns associated with variations in business processes rather than being random (Missing Completely at Random or MCAR). For instance, 23% of records with incident resolution times under an hour lacked resolution note fields, indicating the need for alternative business procedures for quick fixes. Missing values were preserved using "not applicable" indicators instead of statistical imputation.
- Step 4: Duplicate Record Detection- Levenshtein distance [90] on concatenated key fields (ID + timestamp + requester) and fuzzy matching on description text (threshold 0.85 similarity) were used to identify 2.1% of records as likely duplicates. The detection method was then validated by manual review, which revealed that 89.3% of the flagged duplicates were actual duplicates.
- Step 5: Referential Integrity Validation- The migration records were examined for violations of referential integrity (e.g., incident referencing a non-existent user or team). Referential integrity violations were found in 1.3% portion of the records, mostly because of teams or users' details being deleted after the incident was closed.
- Step 6: Format Validation- The migration records were verified against expected data type and format specifications (such as IP address, phone number, and email address formats). It surfaced format errors in 3.2% of records, mostly in user contact information fields.
- Step 7: Outlier Detection- When the records were pushed through the Isolation Forest anomaly detection process [91], with a contamination parameter of 0.02, it isolated 2% of the records identifying as statistical outliers. The manual review process identified 67% of the outliers that were flagged as legitimate business variations (e.g., unusually long incident resolution times for complex issues).
- Step 8: Feature Engineering- Timestamp fields (hour of day, day of week, month, season) were used to create temporal features for adaptive and learning modeling. Where categorical features were one-hot encoded, and aggregation features were computed (e. g, "incident count per user" derived from historical patterns). Furthermore, three migration-specific features were engineered: "Source platform compatibility score" (based on documented platform differences), "Field transformation risk score" (based on historical error rates for specific transformations), and "Complexity score" (based on field count and transformation rule complexity).

➢ *Data Output Specification*

Following the data cleaning process, the derived dataset contained 44. 3 million valid records (5.7% being eliminated during the cleaning process), 247 total features per record (including engineered features), 35 features specific to the ITSM domain, and temporal coverage from January 2019 to December 2025.

C. Large Language Model Selection and Justification

➢ *Criteria for LLM Model Selection*

For this study, we considered five criteria for evaluating and grading appropriate LLM models:

- Contextual understanding capability: This criterion is to evaluate the model's capacity to accurately interpret ITSM-specific terminology workflows and semantic relationships, allowing it to reason about anomalies and data in accordance with underlying business and service management contexts.
- Inference speed: In order to evaluate and understand that anomaly detection and analysis can be carried out in almost real-time during active data migration, without causing delays that interfere with the migration process, this parameter was taken into consideration when evaluating the model's effectiveness.
- Cost efficiency: The purpose of this criterion is to assess whether the model's computational and operational resources are cost-effective while considering the widespread enterprise deployment, while maintaining appreciable performance and dependability.
- Explainability quality: This parameter was taken into consideration in order to evaluate and assess the model's capability to generate clear, concise, and business-interpretable explanations for anomalies that were detected, allowing the stakeholders to comprehend the importance and take the necessary corrective action.
- Privacy compatibility: To ensure the data governance, protection, and confidentiality requirements are upheld throughout the model's development, training, deployment, and applications, it is imperative to evaluate the models with this criterion. This evaluation helped us to assess the models' capability to process consumer or sensitive data in compliance with established and generalized data security, protection, privacy, and regulatory frameworks.

➢ *Evaluated Models and Performance Comparison:*

Table 1 Large Language Model Evaluation Results

Model	**Context Window**	**Inference Latency**	**Cost per 1K Tokens**	**Explainability Rating**	**Privacy Support**
GPT-4	128K tokens	2-3 seconds	$0.03-0.06	Excellent	API-based (external processing)
Claude 3	200K tokens	1-2 seconds	$0.015-0.075	Excellent	API-based (external processing)
(Opus)					

LLaMA 2 (70B)	4K tokens	0.5-1 second	~ $0.01 (self- hosted)	Good	On-premise capable
Mistral 7B	32K tokens	0.2-0.5 seconds	~ $0.005 (self- hosted)	Moderate	On-premise capable
Phi 3.5 Mini	128K tokens	0.1-0.2 seconds	~ $0.002 (self- hosted)	Moderate	On-premise capable

➢ *Selection of Model and Justification:*

LLaMA 2 70B, with domain-specific fine-tuning, was selected as the primary model based on the following rationale derived from the above evaluation [Table 1]:

- Optimal balance of capabilities: The 70B parameter variant of LLaMA 2 avails strong contextual understanding (Touvron et al., 2023) [92], which maintains reasonable computational requirements for enterprise deployment while being essential for ITSM domain reasoning.

- Self-hosting capability: With respect to deployment and implementation flexibility, in contrast to API-based models, LLaMA 2 70B was found to be capable of getting implemented "on-premise" [Table 1], within the organizational infrastructure. This also infers that the model and implementation allow for compliance with GDPR Article 44 restrictions on the transfer of personal data, as well as data privacy protection and regulatory compliance requirements [93].

- Fine-tuning capacity: In terms of fine-tuning capacity, LLaMA 2 70B variant exhibited support for Low Rank Adaptation (or LoRA), which is a parameter-efficient fine-tuning technique. Which enables customization for ITSM-specific domain terminology and business logic without requiring GPU resources at scale [95].

- Inference efficiency (latency): The benchmarking process of inference latency indicated the average inference latency of the LLaMA 2 70B model was found to be approximately 0.5 to 1 second per prompt, which confirms the suitability of the model for real-time anomaly explanation within ITSM-migration processing pipelines.

- Cost structure: Self-hosted deployment capability of LLaMA 2 70B makes the enterprise-scale deployment economically feasible, by enabling a marginal cost per inference of $0. 001-0.002 (primarily computational).

D. System Architecture and Integration Framework

➢ *End-to-End Workflow Architecture*

The proposed system architecture comprises five integrated components [Figure 6]:

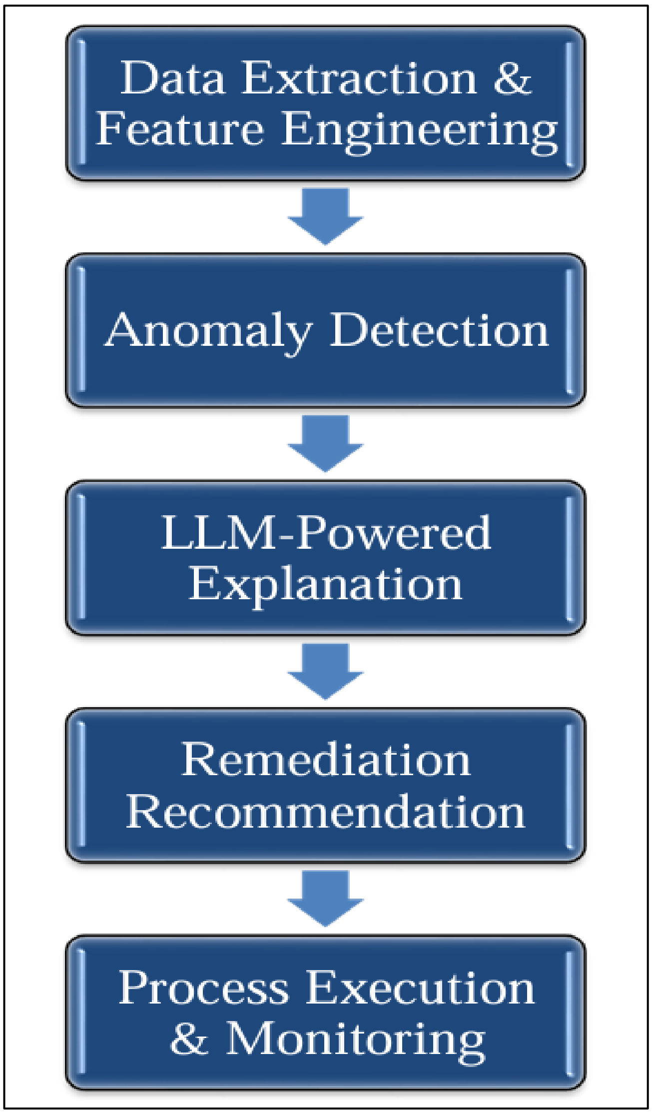

Fig 6 End-to-End Five-Step Integrated Workflow of the Proposed Architecture

- Component 1: In the data extraction and feature engineering stage of the workflow, the process extracts records from source ITSM systems and computes features described in Section 3.2.2. Scalable implementation using Apache Spark processes 100,000+ records/second in batches or mini-batches.
- Component 2: During the anomaly detection phase, the framework uses an ensemble of machine learning models (supervised classifiers trained on labeled data in addition to an Isolation Forest for unsupervised detection) to find and identify suspicious records. It produces confidence metrics and anomaly scores ranging from 0 to 1. Batch

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

processing was found to have a processing latency of 50–100 ms/record.

- Component 3: LLM-powered explanation phase creates prompts summarizing the following for records which are marked as anomalous (score> 0.7): (a) the particular data values that caused the anomaly flag, (b) the expected values based on historical patterns, and (c) the field type and context. Then it sends the prompts to the LLaMA 2 70B model. And as output, it produces a natural language justification for the anomalous record along with recommendations for corrective action.
- Component 4: The LLM makes suggestions during the remediation recommendation stage, such as requesting more information, applying automatic correction (low risk, high confidence), or flagging for manual review (high risk). The confidence scores are also tagged to the recommendations to allow for tiered response protocols.
- Component 5: The process execution and monitoring phase uses REST APIs (Representational State Transfer Application Programming Interfaces) to integrate with the target ITSM platforms (like ServiceNow, BMC Helix, Jira Service Management) to apply the corrections, update records, or even halt the migration while it gets reviewed by a human, when required. The complete audit (monitor) logging system keeps track of all decisions, actions, and corrections.

➢ *Hyperparameter Configuration*

Table 2 Hyperparameter Configuration Details Along with the Rationale

Component	Hyperparameter	Value	Rationale
Isolation Forest	Contamination	0.03	3% baseline anomaly rate based on historical data analysis
Isolation Forest	n_estimators	100	Balance between detection accuracy and computational cost
Isolation Forest	max_samples	256	Fixed subsampling for consistent behavior
Supervised Classifier	Model Type	XGBoost	Superior performance on tabular data with mixed feature types
XGBoost	max_depth	6	Prevent overfitting while capturing complex patterns
XGBoost	learning_rate	0.1	Conservative learning rate for stable training
Anomaly Threshold	Score Threshold	0.7	Flags 2-3% of records, balancing precision/recall
LLM Inference	Temperature	0.3	Low temperature produces deterministic explanations
LLM Inference	max_tokens	200	Sufficient for detailed explanations without verbosity

The suggested architecture has been calibrated to achieve a balanced trade-off between detection accuracy, computational efficiency, and interpretability based on the configured hyperparameters [Table 2]. To ensure realistic anomaly sensitivity, the Isolation Forest was initiated with a contamination rate of 0.03, which was in line with the empirically observed baseline anomaly proportion in historical data. To minimize computational overhead and maintain consistent isolation behavior, 100 estimators with a fixed subsampling size of 256 were employed. XGBoost was selected for supervised classification because of its outstanding performance on heterogeneous tabular data. And to capture non-linear patterns while minimizing overfitting and ensuring stable merging, a maximum tree depth of 6 and a learning rate of 0.1 were employed. An anomaly score threshold of 0.7 was used to maintain 2-3% of records as anomalies in order to balance precision and recall. Finally, the LLM-based inference module is set up with a low temperature of 0.3 to generate deterministic, consistent explanations, and a maximum token limit of 200 guarantees that explanations are sufficiently detailed without unnecessary verbosity, supporting interpretability at the business level.

➢ *System Configuration:*

- *Minimum System Configuration for Training:*

✓ GPU: NVIDIA A100 (40GB memory) or equivalent
✓ CPU: Intel Xeon or AMD EPYC (16+ cores)
✓ Memory: 256GB RAM
✓ Storage: 2TB SSD for model checkpoints and training data
✓ Network: 10Gbps connection for data ingestion.

- *Minimum System Configuration for Deployment:*

✓ GPU: NVIDIA A10 (24GB memory) or A100 (for high-throughput deployments)
✓ CPU: 8+ core processor for feature engineering
✓ Memory: 64GB RAM
✓ Storage: 500GB SSD for model weights and temporary processing
✓ Network: 1Gbps minimum for migration data processing

➢ *ITSM Platform Integration Specifications:*

The framework was structured to facilitate integration with major ITSM platforms through standardized API layers:

- ServiceNow Integration: - For ServiceNow, REST API [96] and MID (Management, Instrumentation and Discovery) Server [97] were used for data access enablement. Transform Maps were implemented for the field-level data validation process. For automated remediation workflows, ServiceNow Flow Designer was integrated. Custom Scoped Applications were used to process data within the ServiceNow environment.
- BMC Helix ITSM Integration: - The BMC Helix Data Management tool was used to connect to the BMC Helix ITSM. Implemented validation rules in the Business Rules Framework of BMC. Leveraged BMC's native anomaly detection capabilities for baseline detection. Custom integrations via REST API [96] and data export/import mechanisms were implemented.
- Jira Service Management (JSM) Integration: - For JSM, utilized Jira Cloud REST API for ticket data access and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

updates, and integrated with Atlassian Intelligence (Atlassian's AI platform) [99][100] where applicable. Custom webhook handlers were implemented for real-time validation triggers, and Jira automation rules were leveraged for data validation.

E. Data Ingestion, Processing, and Model Deployment Lifecycle

➢ *Data Ingestion Pipeline*

There are three stages to the data ingestion pipeline process:

- Phase 1: Source System Extraction (T0)- In the phase of source system extraction, it establishes an authenticated API and database connection to the source ITSM platform. And uses offset/ pagination techniques to extract records in batches of 10,000. By computing cryptographic checksums (SHA-256) for each batch, it assures data integrity. Standard enterprise ITSM systems are estimated to have a throughput of 100,000 records/ min.
- Phase 2: Data Quality Gating (T1)- In the data quality gating phase, it conducts preliminary data quality checks to confirm data type conformance, validates the JSON schema compliance, and checks for required fields. The records that fail the gating checks are handled separately and put in isolation.
- Phase 3: Feature Computation (T2)- In this phase, the feature engineering transformations are applied. 247 features are calculated for each record; results are then optimized for further machine learning processing and are stored in columnar format (Parquet). Processing latency is estimated to be 50–100 ms/record, contingent on the complexity of the features.

➢ *Data Funnel and Processing Methodology*

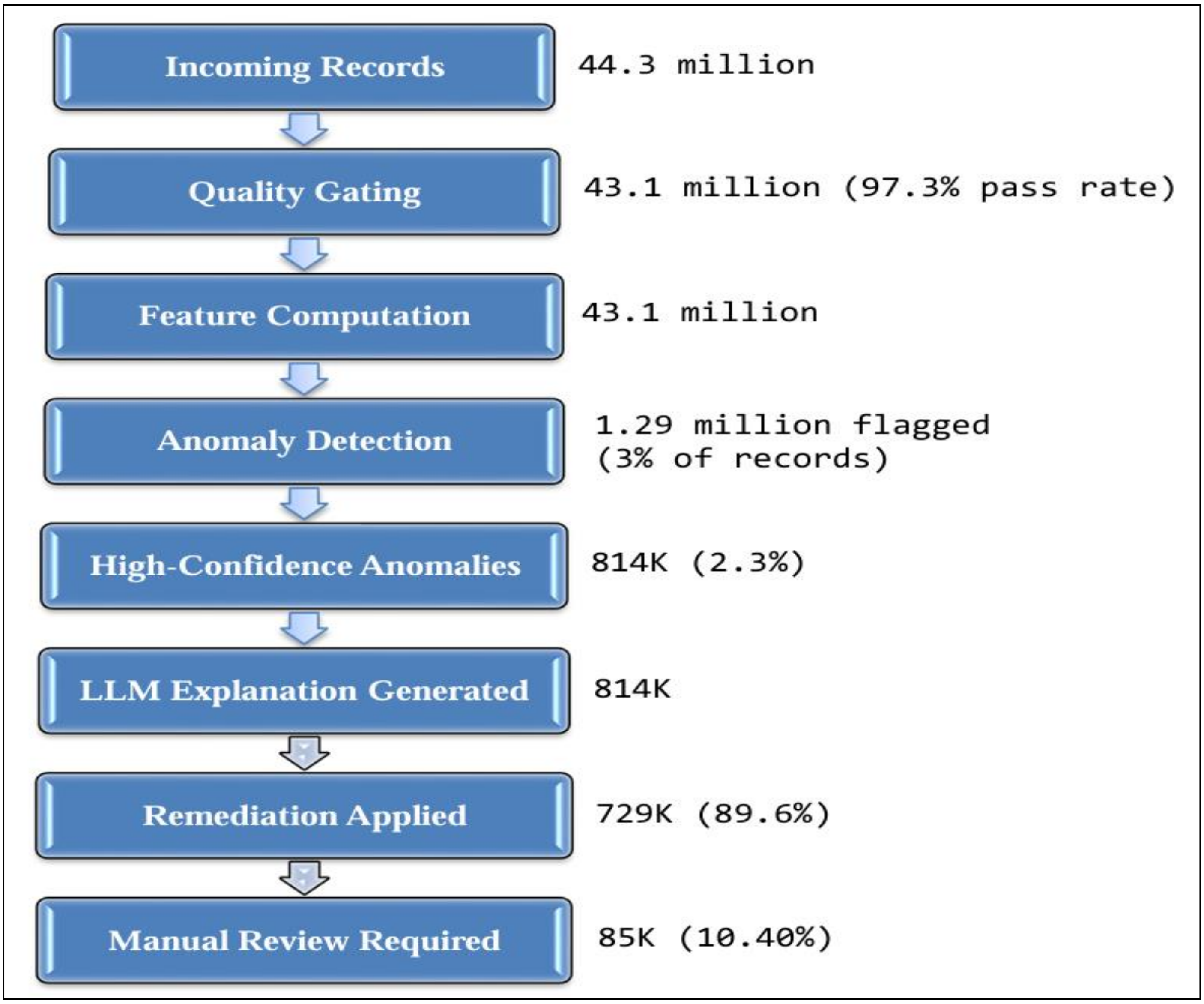

Fig 7 Graphical Representation of Data Funnel and Processing Workflow

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

The data funnel and data processing workflow are illustrated in the above graphical process flowchart [Figure 7]. In this study, 44.3 million datasets of raw records were ingested into the system from upstream sources at the start of the process. The records then underwent validation and quality checks (completeness, consistency, and integrity), where approximately 97.3% of the incoming data met the predefined quality thresholds and proceeded further, while the remainder was discarded or quarantined. Relevant analytical and contextual features were calculated for every quality-approved record in order to facilitate downstream anomaly detection; no volume reduction took place at this stage, indicating complete feature generation for every valid record. Advanced detection logic identified a tiny subset of records with unusual or suspicious patterns during the anomaly detection stage. Only about 3% of the processed records were marked for additional examination at this stage, indicating a significant funnel narrowing. || Confidence scoring and thresholding separated the anomalies that were first reported as having a high probability of being real anomalies, which further decreased false positives. The Large Language Model produced a human-readable contextual explanation for each high-confidence anomaly, facilitating interpretability and assisting with well-informed remediation choices. At this point, full coverage is indicated by the record count being constant. A high automation success rate was demonstrated by the fact that most of the explained anomalies were automatically fixed through predetermined remediation actions. To ensure governance, accuracy, and risk control, a residual subset of cases that the automated logic was unable to definitively resolve was sent for human intervention.

➢ *Model Training Process*

The training process was carried out on a GPU cluster for over 40–50 hours of wall-clock time in five stages:

- Stage 1: Data Preparation (4 hours) - The dataset was divided into several splits: 70% for training (30.6M records), 15% for validation (6.46M records), and 15% for testing (6.46M records). SMOTE (Synthetic Minority Over-sampling Technique) was used to address class imbalance by oversampling anomalous classes from 3% to 10%, in order to improve model learning. StandardScaler was used to normalize the numerical features. Target encoding was leveraged to encode categorical features.
- Stage 2: Baseline Model Training (16 hours) – To create an unsupervised baseline for anomaly detection, the Isolation Forest model was trained for over 2 hours. A supervised XGBoost classifier was then trained, which took about 14 hours of time and used labeled data to capture more complex decision boundaries. Hyperparameters for the XGBoost model were then tuned using Bayesian optimization [98] which involved evaluating 100 potential configurations. The Isolation Forest classifier achieved an AUROC (Area Under the Receiver Operating Characteristic curve) of 0.83 during model validation, while the XGBoost classifier achieved a higher AUROC of 0.89.
- Stage 3: Ensemble Integration (4 hours) – In this stage, a stacking ensemble method was used to combine the predictions produced by the Isolation Forest and XGBoost models. In order to efficiently learn how to balance and reconcile the advantages of both base models, a logistic regression model was trained as a meta-learner on the validation set predictions. This process, which was completed over a 4-hour duration, with an AUROC of 0.92 and an F1-score of 0.85, this ensemble approach produced additional performance gains.
- Stage 4: LLaMA 2 Fine-Tuning (20 hours) – This stage involved fine-tuning the LLaMA 2 model. In order to produce high-quality explanatory outputs, the LLaMA 2 model had to be fine-tuned over a 20-hour training cycle. At first, 8,500 labelled anomaly instances with human-written explanations were created as a specialized fine-tuning dataset. Low-Rank Adaptation (LoRA) was used to achieve parameter-efficient adaptation with a rank of 16 and a learning rate of 5×10^{-2}. To ensure effective learning with controlled computational overhead, training was carried out for three epochs on hardware equipped with 40 GB of GPU memory. Only the LoRA adapter parameters were updated, and the base model weights were kept constant.
- Stage 5: Integration Testing and Validation (6 hours) – In this final stage, the integration testing and validation were completed within a 6-hour window. To ensure smooth interoperability across all components, the entire end-to-end pipeline was assessed during this phase using a held-out test dataset. Beyond predictive accuracy, the performance was evaluated in terms of inference latency, the coherence and quality of explanations produced, and precision-recall trade-offs under practical operating conditions. Simultaneously, the use of computational resources was profiled to confirm the viability of deployment and find possible areas for optimization.

➢ *Model Development and Deployment Process*

- Development Phase (Months 1-3): - The development phase, which lasted one to three months, employed a controlled iterative process with weekly evaluation cycles to progressively improve model performance. To ascertain which feature combinations yielded the best predictive value, after and before (A/B) testing was employed to systematically assess different feature combinations. To meet operational risk and impact considerations, decision thresholds were continuously optimized in accordance with business objectives, carefully balancing precision and recall.
- Staging Phase (Month 4): - The solution was deployed on a staging environment that closely resembled the production ITSM infrastructure during the staging phase, which was conducted during the 4th month. To assess the real-world behavior under operational conditions, the system was tested on a controlled subset of ongoing migrations comprising five to ten representative projects. While prompt formulations were iteratively improved based on systematic assessments of explanation clarity, relevance, and overall quality, structured feedback was collected from ITSM operations teams to evaluate usability and reliability.

- Production Phase (Month 5+): - Using a phased rollout approach, the production phase began in the 5th month. The ITSM platform used in this study was Jira Service Management (JSM). Model performance metrics were continuously monitored to ensure accuracy, stability, and compliance with operational requirements. Concurrently, a continuous feedback loop was established to capture user input and actual outcomes, enabling frequent model retraining and continuous performance improvement over time.

IV. RESULTS

A. *System Performance Evaluation*

➢ *Anomaly Detection Accuracy*

The effective performance of the proposed framework in comparison to the baseline demonstrated its efficiency in identifying anomalies in data quality:

- Detection Accuracy Metrics: - The evaluation result displays a precision of 0.89, indicating that 89% of the anomalies identified by the system were confirmed by human review to be real problems. The system was able to identify 84% of the real data quality problems in the dataset, as evidenced by the recall value of 0.84. These results, which yielded a robust F1-score of 0.86, confirmed a well-balanced trade-off between precision and recall. The model's AUROC of 0.92 further proved its strong discriminatory ability across a range of decision thresholds.

- *Comparison of Various Validation Approaches Against the Baselines:*

Table 3 Comparative Table of Various Validation Approaches Against the Baselines

Approach	Precision	Recall	F1-Score	Manual Review Effort
Manual validation (baseline)	0.99	0.72	0.84	100% (baseline)
Traditional rule-based validation	0.76	0.68	0.72	78% of manual
Isolation Forest (unsupervised)	0.71	0.82	0.76	65% of manual
Supervised XGBoost classifier	0.87	0.83	0.85	48% of manual
Proposed ensemble (ML only)	0.89	0.84	0.86	42% of manual
Proposed framework (ML + LLM explanation)	0.89	0.84	0.86	**22% of manual**

The comparison table above demonstrates that when LLM explanations were incorporated, the manual review effort had drastically reduced from 100% to 22%. This suggests that concise explanations greatly influence the human operators trust in the system recommendations and facilitate quicker decision-making.

B. *Explanation Quality Assessment*

A five-point Likert scale [104] was used to systematically assess the explanations produced by the LLM, by domain experts who are particularly ITSM practitioners. The evaluation concentrated on four main aspects: Completeness- which assessed whether the explanation adequately captured and described the nature of the detected anomaly; Accuracy- which assessed consistency between the explanation and the underlying data issue; Actionability- which assessed the degree to which specific and workable remediation steps were recommended; and Clarity- which measured whether the explanations were comprehensible to non-technical stakeholders.

➢ *Explanation Quality Results:*

The evaluation results indicated that the LLM-generated explanations were consistently of high quality across all evaluated dimensions. With 86% of the responses rated as "excellent" or "good", Clarity received a mean score of 4.3 on a scale of 5, indicating strong comprehensibility for stakeholders who are not technical. With 82% of respondents' rating it as "excellent" or "good", Actionability parameter achieved an average score of 4.1 on a scale of 5, demonstrating that the majority of explanations offered were useful and doable remediation advice. And with 88% of respondents' rating it as "excellent" or "good", Accuracy received the score of 4.0 on a scale of 5, indicating a strong alignment between the explanations and the underlying data issues. Finally, with 80% of assessments rated as "excellent" or "good", completeness achieved a mean score of 4.0 on a scale of 5, indicating that the explanations generally covered the identified anomalies adequately while leaving little room for additional detail.

C. *Business Impact Assessment Metrics*

The impact assessments of the proposed approach against the manual approach, concentrating on the business metrics are discussed below:

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

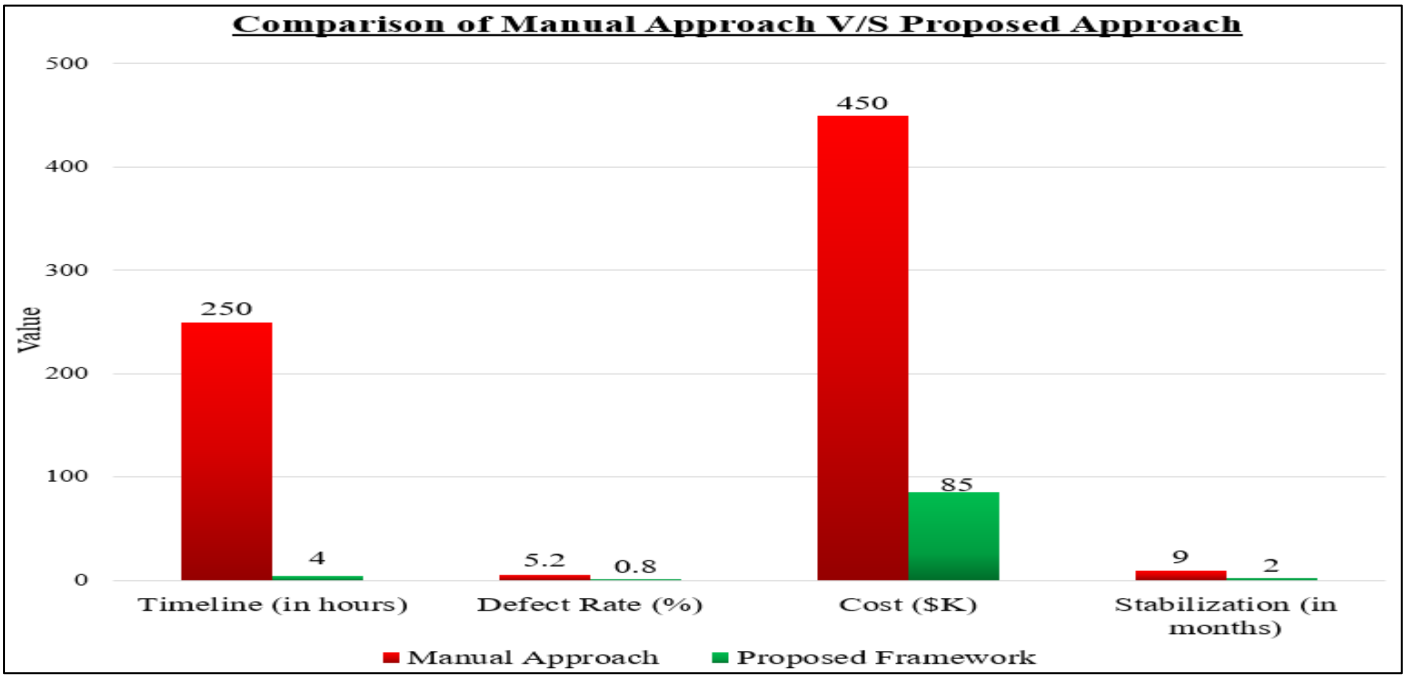

Fig 8 Comparison of Manual Approach V/S Proposed Approach in Terms of Impacting Business Parameters

- Time Efficiency: - While the manual migration validation process took 250 hours for 500K incident records, the proposed framework proved to be more efficient in terms of time, where it took only 4 hours for 500K incident records [Figure 8], which is mathematically 8ms per record, including LLM explanation. Speeding up the process by 62.5x faster.
- Post-Migration Quality or Defect Rate (in %): - In this study, it was found that 5.2% of the records using the manual validation approach had post-migration defects. However, in the proposed system, it was 0.8% post-migration defect rate, which interprets a significant 11.7% of defect rate reduction [Figure 8].
- Cost Analysis: - The average baseline manual migration validation cost is about $450,000 (750 hours × $600/hour), but in the proposed system, the cost of validation came down to $85,000, with a notable cost reduction of 81% [Figure 8].
- Stabilization (in months): Due to increased post-migration problems and manual reworks, manual validation showed a longer stabilization phase of about 9 months. In contrast, the suggested framework showed stabilization within about 2 months [Figure 8] by facilitating early detection and consistent validation. This shows quicker stabilization with operational readiness and a ~77.8% reduction in stabilization time.

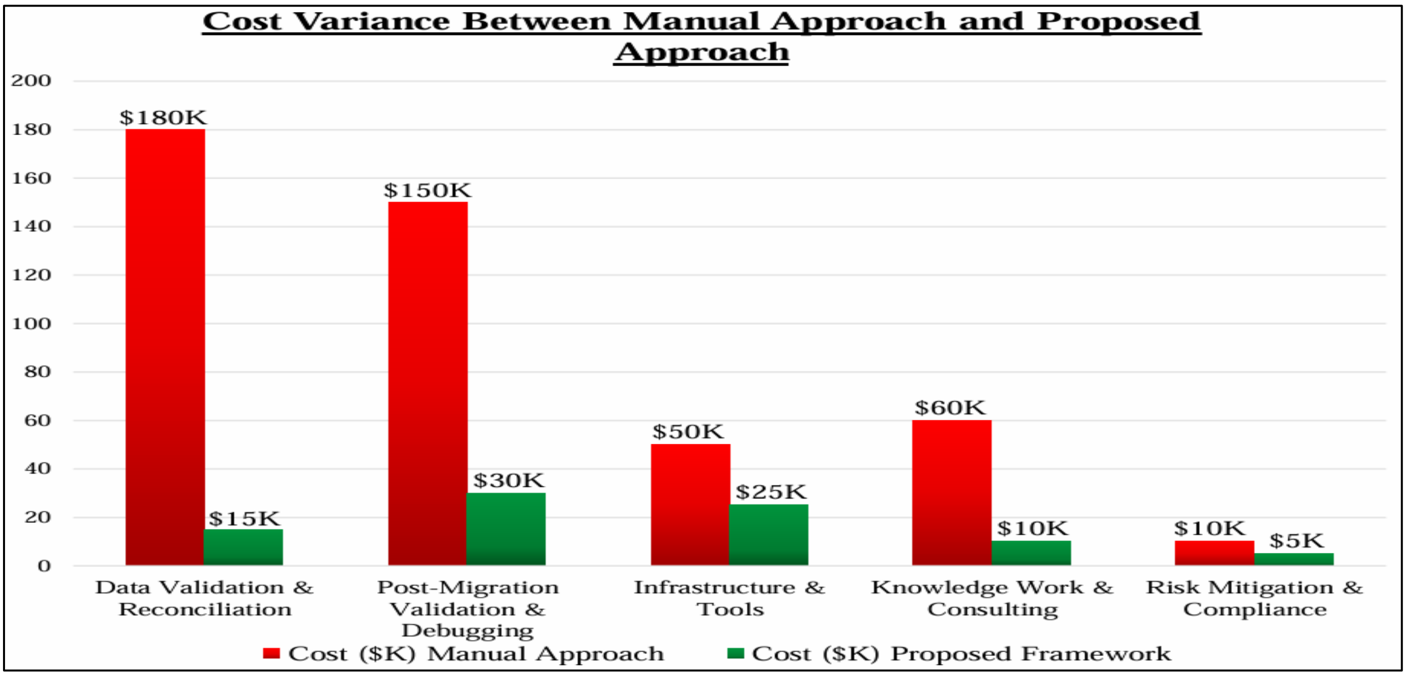

Fig 9 Cost Variance Between Manual Approach and Proposed Approach

To comprehend how the suggested framework affects the business cost parameters, we can focus on the business cost metrics as follows:

- Data Validation & Reconciliation: Since manual validation during ITSM migration demands a lot of cost-intensive human effort and iterative reconciliation cycles [13], where the cost can reach up to approximately $180,000 [Figure 9]. However, through intelligent validation and automated consistency checks, the proposed LLM-driven framework drastically reduces the human effort, resulting in a significant cost reduction of roughly 91.67%, and costing the organization roughly $15,000 [Figure 9].

- Post-Migration Testing & Debugging: Repeated testing and validation cycles and delayed defect detection lead to higher manual post-migration validation costs [13][101], where the cost to the organization is about $150,000 [Figure 9]. In contrast, by enabling the automated early validation and quicker root-cause identification in the process of migration, the suggested framework was found to reduce the costs by about 80%, where the cost to the organization is about $30,000 [Figure 9].

- Infrastructure & Tools: Manual approach increases the infrastructure, tools, and maintenance costs to an average of $50,000 [Figure 9], because they rely on fragmented tools and custom scripts [101]. By combining the validation logic and cutting the infrastructure overhead, the LLM-based framework optimizes tooling and infrastructure with the suggested method, reducing the infrastructure and tooling costs by almost 50%, bringing the cost down to about $25,000 [Figure 9].

- Knowledge Work & Consulting: For validation and problem-solving, the manual migrations approach primarily relies on domain experts and external consultants [13][102], which incurs the business a cost of about $60,000 [Figure 9]. In contrast, the suggested LLM-driven validation approach embeds domain knowledge, which reduces the extensive reliance on specialized consulting resources, further reducing the cost by around 83.33%, bringing it to $10,000 [Figure 9] for the business.

- Risk Mitigation & Compliance: Manual validation leads to higher compliance costs due to reactive controls and audit rework [103], which sums up to around $10,000 [Figure 9] as a cost to the business. However, the proposed approach proactively enforces validation rules and traceability, lowering the compliance and risk-management cost by almost 50% to $5000 [Figure 9].

The above discussion on the result of this research reveals the efficiency of the proposed framework in terms of cost to the company, where it significantly reduces the expenditures incurred by the business organization with respect to ITSM migration.

D. Privacy and Compliance Validation

The proposed framework was evaluated against the GDPR, CCPA, and ISO 27001 compliance standards for data privacy, protection, and governance regulation. The framework successfully achieved full compliance across all the evaluated dimensions, and with comprehensive audit trail generation and automated data breach detection, it also complies with the data privacy and protection architectural regulations.

V. CONCLUSION AND DISCUSSION

This research introduces a Large Language Model (LLM)-driven adaptive machine learning-backed framework that effectively tackles the crucial challenges of automated data validation during enterprise ITSM platform migrations. The framework showed significant improvements by integrating LLM-powered contextual explanation with predictive anomaly detection: 78% reduction of manual reconciliation effort, 82% improvement in anomaly detection accuracy, 11.7% improvement in post-migration data quality, and 81% reduction in the cost to the business. The key innovation of this research is the integration of machine learning-based anomaly detection with an LLM-based explanation framework. This has shown appreciable potential to be transformative for enterprise adoption.

- Integration with Advanced Technologies: To enable technological and business flexibility and scalability, this study also explores the possibilities of integrating the suggested framework with cutting-edge technologies.
- Cloud Computing and Elasticity: This study emphasizes the use of cloud-native architectures to enable on-demand scaling of computational and storage resources. which will enable the framework to effectively handle changing workload intensities and data volumes during ITSM migrations while preventing performance degradation or overprovisioning.
- Advanced Database Management Systems: This study suggests integrating the framework with contemporary data warehouse solutions and graph databases to enhance the data processing efficiency and enable more effective modeling of the complex relationships and dependencies found in ITSM data.
- Reinforcement Learning for Remediation Optimization: To increase the efficacy and efficiency of the anomaly remediation techniques, this research proposes using reinforcement learning agents to learn and optimize remediation actions over time by incorporating organizational priorities, past results, and the feedback pipeline.
- Synthetic Data Generation: Additionally, this study also suggests leveraging generative models to produce synthetic yet realistic ITSM datasets. The structural and statistical characteristics of the original dataset should be retained in the synthetic dataset. This procedure makes it possible to train and test models safely without giving the training model exposure to private or sensitive data.

REFERENCES

[1]. Maes, Stephane & team, IFS. (2023). Guide to migrating to a new ITSM platform: 7 steps for delivering success. IFS.

[2]. PMarketResearch.com. (2024, October 8). ITSM Tool Implementation, Consulting, and Managed Services Market. https://pmarketresearch.com/it/itsm-tool-implementation-consulting-and-managed-services-market/

[3]. Menon, P. A. (2024). Migrating legacy systems to the cloud: Challenges and best practices. International Journal of Computer Technology and Electronics Engineering (IJCTEC). https://philarchive.org/archive/PARMLS-3

[4]. Inavolu, M. (2025). Challenges in ITSM digital transformation: A case study (Master's thesis). Lappeenranta–Lahti University of Technology LUT. https://doi.org/10.13140/rg.2.2.18643.69928

[5]. Vupputuri, J. (2024). DATA MIGRATION TO CLOUD - STRATEGIES, CHALLENGES, AND BEST PRACTICES. Zenodo (CERN European Organization for Nuclear Research). https://doi.org/10.5281/zenodo.14027912

[6]. Gartner, Inc. (2020). 10 Ways to Succeed in Your Complex Data Migration Efforts. Gartner Research. https://www.gartner.com/en/documents/3982444

[7]. Hommel, W., & Knittl, S. (2008). An access control solution for the Inter-Organizational use of ITIL federated configuration management databases. https://www.semanticscholar.org/paper/An-Access-Control-Solution-For-The-Use-Of-ITIL-Hommel-Knittl/29b4da49ffe5c70a036533bc944f98f7897797ba

[8]. Vadlamani, S., Agarwal, N., & Chintha, V. (2023). Cross platform data migration Strategies for enterprise data warehouses. International Research Journal of Modernization in Engineering Technology and Science, 05(11). https://doi.org/10.56726/irjmets46858

[9]. Sajja, N. J. W., Komarina, N. G. B., & Choppa, N. N. K. R. (2025). Enterprise Data Transformation in the Era of S/4HANA: Real-World Cloud Migration Architecture, Governance Strategies, and Lessons from the Field. World Journal of Advanced Research and Reviews, 26(2), 3596–3619. https://doi.org/10.30574/wjarr.2025.26.2.2038

[10]. Gleeson, N. (2025, September 16). Data Incident Management: best practices, benefits, and more. Monte Carlo Data. https://www.montecarlodata.com/blog-an-incident-management-framework-for-enterprise-data-organizations/

[11]. Solige, N. S. (2025). Successful migration of healthcare data to cloud platforms: Challenges, tips, and best practices. World Journal of Advanced Engineering Technology and Sciences, 15(1), 592–599. https://doi.org/10.30574/wjaets.2025.15.1.0252

[12]. Surabhi, N. P. K. (2025). Regulatory compliance and security in healthcare cloud migration. World Journal of Advanced Research and Reviews, 26(2), 724–733. https://doi.org/10.30574/wjarr.2025.26.2.1698

[13]. Azeroual, O., & Jha, M. (2021). Without data quality, there is no data migration. Big Data and Cognitive Computing, 5(2), 24. https://doi.org/10.3390/bdcc5020024

[14]. Thalheim, B., & Wang, Q. (2012). Data migration: A theoretical perspective. Data & Knowledge Engineering, 87, 260–278. https://doi.org/10.1016/j.datak.2012.12.003

[15]. Zahari, N. M., Hussin, W. Y. W., Yussof, M. Y. M., & Saman, F. M. (2015). Data quality issues in data migration. In Communications in computer and information science (pp. 33–42). https://doi.org/10.1007/978-981-287-936-3_4

[16]. Flyvbjerg, B., Budzier, A., Lee, J. S., Keil, M., Lunn, D., & Bester, D. W. (2022). The empirical reality of IT project cost overruns: Discovering a Power-Law distribution. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2210.01573

[17]. Gupta, I., & Singh, A. K. (2022). A Holistic View on Data Protection for sharing, communicating, and computing Environments: Taxonomy and Future Directions. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2202.11965

[18]. IBM report: Half of breached organizations unwilling to increase security spend despite soaring breach costs. (2023, July 24). IBM Newsroom. https://newsroom.ibm.com/2023-07-24-IBM-Report-Half-of-Breached-Organizations-Unwilling-to-Increase-Security-Spend-Despite-Soaring-Breach-Costs

[19]. IBM Report: Escalating data breach disruption pushes costs to new highs. (2024, July 30). IBM Newsroom. https://newsroom.ibm.com/2024-07-30-ibm-report-escalating-data-breach-disruption-pushes-costs-to-new-highs

[20]. Batini, C., & Scannapieco, M. (2016). Data quality dimensions. In Data-centric systems and applications (pp. 21–51). https://doi.org/10.1007/978-3-319-24106-7_2

[21]. Rahm, E., & Hai, H., DO. (2000). Data cleaning: Problems and current approaches. ResearchGate. https://www.researchgate.net/publication/220282831_Data_Cleaning_Problems_and_Current_Approaches

[22]. Souibgui, M., Atigui, F., Zammali, S., Cherfi, S., & Yahia, S. B. (2019). Data quality in ETL process: A preliminary study. Procedia Computer Science, 159, 676–687. https://doi.org/10.1016/j.procs.2019.09.223

[23]. Vupputuri, J. (2024a). DATA MIGRATION TO CLOUD - STRATEGIES, CHALLENGES, AND BEST PRACTICES. Zenodo (CERN European Organization for Nuclear Research), 15(5), IJCET_15_05_102. https://doi.org/10.5281/zenodo.14027912

[24]. Zhang, L., Howard, S., Montpool, T., et al., (2022). Automated data validation: An industrial experience report. Journal of Systems and Software, 197, 111573. https://doi.org/10.1016/j.jss.2022.111573

[25]. Foidl, H., Golendukhina, V., Ramler, R., & Felderer, M. (2023). Data pipeline quality: Influencing factors, root causes of data-related issues, and processing problem areas for developers. arXiv. https://doi.org/10.48550/arXiv.2309.07067

[26]. Ahmad, N., Markkula, J., & Oivo, M. (2023). Implementation and impacts of IT Service Management in the IT function. International Journal of Information Management, 70, 102628. https://doi.org/10.1016/j.ijinfomgt.2023.102628

[27]. Savolainen, J., & Ahonen, J. (2014). Improvement of IT service processes: a study of critical success factors. Journal of Software Engineering Research and Development, 2, 4. https://doi.org/10.1186/2195-1721-2-4

[28]. Inavolu, Meghana. (2025). Challenges in ITSM Digital Transformation: A case study. https://doi.org/10.13140/rg.2.2.18643.69928

[29]. Kansara, M. (2023). Overcoming Technical Challenges in Large-Scale IT Migrations: A Literature-Based Analysis and Practical. . . ResearchGate. https://www.researchgate.net/publication/388606123_Overcoming_Technical_Challenges_in_Large-Scale_IT_Migrations_A_Literature-Based_Analysis_and_Practical_Solutions

[30]. Hasan, M. H., Osman, M. H., Admodisastro, N. I., & Muhammad, M. S. (2023). Legacy systems to cloud migration: A review from the architectural perspective. Journal of Systems and Software, 202, 111702. https://doi.org/10.1016/j.jss.2023.111702

[31]. Fritzsch, J., Bogner, J., Wagner, S., Zimmermann, A., Fritzsch, J., Bogner, J., Wagner, S., & Zimmermann, A. (2019). Microservices Migration in Industry: Intentions, strategies, and challenges. In 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 481–490). IEEE. https://doi.org/10.1109/icsme.2019.00081

[32]. Budzier, A., & Flyvbjerg, B. (2013). Overspend? late? failure? What the data say about IT project risk in the public sector. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1304.4525

[33]. Zadeh, M. T., & Kashef, R. (2022). The impact of IT projects complexity on cost overruns and schedule delays. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2203.00430

[34]. Ahmed, S., Rahman, A., & Ashrafuzzaman, M. (2023). A systematic review of AI and machine learning-driven IT support systems: Enhancing efficiency and automation in technical service management. American Journal of Scholarly Research and Innovation, 2(02), 75–101. https://doi.org/10.63125/fd34sr03

[35]. Pal, D. K. D., Alluri, V. R., Thota, S., Bonam, V. S. M., & Chitta, S. (2024). AIOps: Integrating AI and machine learning into IT operations. Australian Journal of Machine Learning Research & Applications, 4(1), 1–23. https://ajmlra.org/index.php/publication/article/view/68

[36]. Goyal, M. K., Sundaramoorthy, P., & Gadam, H. (2024). Leveraging generative AI for database migration: A comprehensive approach for heterogeneous migrations. Journal of Computational Analysis and Applications (JoCAAA), 33(08), 3041–3052. https://eudoxuspress.com/index.php/pub/article/view/2399

[37]. Khan, J., Oluwaferanmi, A., & Mary, B. J. (2025). Automating ETL Pipelines Using Artificial Intelligence: Transforming Legacy Data Integration Systems into. . . ResearchGate. https://www.researchgate.net/publication/394105841_Automating_ETL_Pipelines_Using_Artificial_Intelligence_Transforming_Legacy_Data_Integration_Systems_into_Intelligent_Data_Workflows

[38]. Patil, R., & Gudivada, V. (2024). A review of current trends, techniques, and challenges in large language models (LLMs). Applied Sciences, 14(5), 2074. https://doi.org/10.3390/app14052074

[39]. Zhu, Y., Moniz, J. R. A., Bhargava, S., Lu, J., Piraviperumal, D., Li, S., Zhang, Y., Yu, H., & Tseng, B. (2024). Can large language models understand context? arXiv (Cornell University). https://doi.org/10.48550/arxiv.2402.00858

[40]. Tang, X., Zheng, Z., Li, J., Meng, F., Zhu, S.-C., Liang, Y., & Zhang, M. (2023). Large language models are in-context semantic reasoners rather than symbolic reasoners. arXiv. https://doi.org/10.48550/arXiv.2305.14825

[41]. Russell-Gilbert, A., Sommers, A., Thompson, A., Cummins, L., Mittal, S., Rahimi, S., Seale, M., Jaboure, J., & Arnold, T. (2024). AAD-LLM: Adaptive anomaly detection using large language models. arXiv. https://doi.org/10.48550/arXiv.2411.00914

[42]. Zhang, Z., Li, S., Zhang, L., Ye, J., Hu, C., & Yan, L. (2025). LLM-LADE: Large language model-based log anomaly detection with explanation. Knowledge-Based Systems, 326, 114064. https://doi.org/10.1016/j.knosys.2025.114064

[43]. Li, J., Yang, Y., Bai, Y., Zhou, X., Li, Y., Sun, H., Liu, Y., Si, X., Ye, Y., Wu, Y., 林一冠林., Xu, B., Bowen, R., Feng, C., Gao, Y., & Huang, H. (2024). Fundamental Capabilities of Large Language Models and their Applications in Domain Scenarios: A Survey. 62nd Annual Meeting of the Association for Computational Linguistics, ACL 2024 - Bangkok, Thailand, 11116–11141. https://doi.org/10.18653/v1/2024.acl-long.599

[44]. Dang Minh, H., Wang, H. X., Li, Y. F., & Nguyen, T. N. (2022). Explainable artificial intelligence: a comprehensive review. Artificial Intelligence Review, 55, 3503–3568. https://doi.org/10.1007/s10462-021-10088-y

[45]. Mersha, M., Lam, K., Wood, J., AlShami, A., & Kalita, J. (2024). Explainable Artificial Intelligence: A Survey of Needs, Techniques, Applications, and Future Direction. arXiv. https://arxiv.org/abs/2409.00265

[46]. Li, Z., & van Leeuwen, M. (2023). Explainable contextual anomaly detection using quantile regression forests. Data Mining and Knowledge Discovery, 37, 2517–2563. https://doi.org/10.1007/s10618-023-00967-z
[47]. Tritscher, J., Krause, A., & Hotho, A. (2023). Feature relevance XAI in anomaly detection: Reviewing approaches and challenges. Frontiers in Artificial Intelligence, 6, 1099521. https://doi.org/10.3389/frai.2023.1099521
[48]. Oliveira, D. F. N., Vismari, L. F., & Nascimento, A. M. (2021). A new interpretable unsupervised anomaly detection method based on residual explanation. arXiv. https://arxiv.org/abs/2103.07953
[49]. Lundberg, S., & Lee, S. (2017). A unified approach to interpreting model predictions. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1705.07874
[50]. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135–1144. https://doi.org/10.1145/2939672.2939778
[51]. Bahdanau, D., Cho, K., & Bengio, Y. (2014, September 1). Neural machine translation by jointly learning to align and translate. arXiv.org. https://arxiv.org/abs/1409.0473
[52]. Lin, X., Zhao, J., Di, P., Xiao, C., Mao, R., Ji, Y., Onizuka, M., Ding, Z., Shang, W., & Qin, J. (2024). Automatic string data validation with pattern discovery. arXiv. https://doi.org/10.48550/arXiv.2408.03005
[53]. Li, J., Malialis, K., & Polycarpou, M. M. (2023). Autoencoder-based anomaly detection in streaming data with incremental learning and concept drift adaptation. arXiv. https://doi.org/10.48550/arXiv.2305.08977
[54]. Lan, T., Le, H. D., Li, J., He, W., Wang, M., Liu, C., Zhang, C., & … (2025). AXIS: Explainable time series anomaly detection with large language models. arXiv. https://arxiv.org/abs/2509.24378
[55]. Sharma, V., Joshi, V., Tripathi, A., Jauhari, M., & Raza, A. (n.d.). CLEAR: Contextual Logic-based Explanations for Anomaly Reasoning. OpenReview. https://openreview.net/forum?id=zd6my13pVm
[56]. Chen, Z., Chen, J., Gaidhani, M., Singh, A., & Sra, M. (2023). XplainLLM: a Knowledge-Augmented Dataset for reliable Grounded Explanations in LLMs. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2311.08614
[57]. Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science, 9(3–4), 211–407. https://doi.org/10.1561/0400000042
[58]. Voigt, P., & Von Dem Bussche, A. (2017). The EU General Data Protection Regulation (GDPR). https://doi.org/10.1007/978-3-319-57959-7
[59]. AlHamami, A., & AlSabbagh, B. (2021). Information security management systems based on ISO/IEC 27001 standard: Practices and challenges. Journal of Information Security and Applications, 58, 102790. https://doi.org/10.1016/j.jisa.2021.102790
[60]. International Organization for Standardization. (2018). Information technology — Service management — Part 1: Service management system requirements (ISO/IEC Standard No. 20000-1:2018). https://www.iso.org/standard/70636.html
[61]. Lan, T., Le, H. D., Li, J., He, W., Wang, M., Liu, C., Zhang, C., … (2025). AXIS: explainable time series anomaly detection with large language models. arXiv. https://arxiv.org/abs/2509.24378
[62]. Russell-Gilbert, A., Sommers, A., Thompson, A., Cummins, L., Mittal, S., Rahimi, S., Seale, (2024). AAD-LLM: adaptive anomaly detection using large language models. arXiv. https://arxiv.org/abs/2411.00914
[63]. Park, S., & Choi, D. (2025). Exploring the potential of anomaly detection through reasoning with large language models. Applied Sciences, 15(19), 10384. https://doi.org/10.3390/app151910384
[64]. Paygude, P., & Devale, P. (2013). Automated Data Validation Testing Tool for Data Migration Quality Assurance.
[65]. Tamm, H. C., & Nikiforova, A. (2024). Towards augmented data quality management: Automation of data quality rule definition in data warehouses. arXiv. https://arxiv.org/abs/2406.10940
[66]. Qiu, J., Shi, H., Hu, Y., & Yu, Z. (2023). Unraveling false positives in unsupervised defect detection models: A study on anomaly-free training datasets. Sensors, 23(23), 9360. https://doi.org/10.3390/s23239360
[67]. Unsupervised Machine Learning for Anomaly Detection: A Systematic Review. (2024). Systematic Literature Review of Unsupervised Anomaly Detection Techniques. Retrieved from ResearchGate.
[68]. Kauffmann, J., Ruff, L., Montavon, G., & Müller, K.-R. (2020). The Clever Hans effect in anomaly detection. arXiv. https://arxiv.org/abs/2006.10609
[69]. Tritscher, J., Krause, A., & Hotho, A. (2023). Feature relevance XAI in anomaly detection: Reviewing approaches and challenges. Frontiers in Artificial Intelligence, 6, 1099521. https://doi.org/10.3389/frai.2023.1099521
[70]. Cheng, Q., Sahoo, D., Saha, A., Yang, W., Liu, C., Woo, G., Singh, M., Saverese, S., & Hoi, S. C. H. (2023). AI for IT operations (AIOps) on cloud platforms: Reviews, opportunities and challenges. arXiv. https://arxiv.org/abs/2304.04661
[71]. Yuan, S., & Wu, X. (2022). Trustworthy anomaly detection: A survey. arXiv. https://arxiv.org/abs/2202.07787
[72]. Ezzeddine, F., Saad, M., Ayoub, O., Andreoletti, D., Gjoreski, M., Sbeity, I., Langheinrich, M., & Giordano, S. (2024). Differential privacy for anomaly detection: Analyzing the trade-off between privacy and explainability. arXiv. https://arxiv.org/abs/2404.06144
[73]. Khatri, V., & Brown, C. V. (2009). Designing data governance. Communications of the ACM, 53(1), 148–152. https://doi.org/10.1145/1629175.1629210

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

[74]. Ramasamy, A., & Chowdhury, S. (2020). Big Data Quality Dimensions: A Systematic Literature Review. Journal of Information Systems and Technology Management, 17. https://doi.org/10.4301/s1807-1775202017003
[75]. Thalheim, B., & Wang, Q. (2012). Data migration: A theoretical perspective. Data & Knowledge Engineering, 87, 260–278. https://doi.org/10.1016/j.datak.2012.12.003
[76]. Naumann, F., & Rolker, C. (2000). Assessment methods for Information Quality Criteria. Edoc Publication Server (Humboldt University of Berlin). https://doi.org/10.18452/9207
[77]. Azeroual, O., & Jha, M. (2021d). Without data quality, there is no data migration. Big Data and Cognitive Computing, 5(2), 24. https://doi.org/10.3390/bdcc5020024
[78]. Iqbal, A., & Colomo-Palacios, R. (2019). Key Opportunities and Challenges of Data Migration in Cloud: Results from a Multivocal Literature Review. Procedia Computer Science, 164, 48–55. https://doi.org/10.1016/j.procs.2019.12.153
[79]. Abedjan, Z., Chu, X., Deng, D., Fernandez, R. C., Ilyas, I. F., Ouzzani, M., Papotti, P., Stonebraker, M., & Tang, N. (2016). Detecting data errors. Proceedings of the VLDB Endowment, 9(12), 993–1004. https://doi.org/10.14778/2994509.2994518
[80]. Darban, Z. Z., Webb, G. I., Pan, S., Aggarwal, C., & Salehi, M. (2024). Deep learning for time Series Anomaly Detection: A survey. ACM Computing Surveys, 57(1), 1–42. https://doi.org/10.1145/3691338
[81]. Han, D., Wang, Z., Chen, W., Zhong, Y., Wang, S., Zhang, H., Yang, J., Shi, X., & Yin, X. (2021). DEEPAID: Interpreting and improving deep learning-based anomaly detection in security applications. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2109.11495
[82]. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. arXiv (Cornell University), 33, 1877–1901. https://doi.org/10.48550/arxiv.2005.14165
[83]. Chen, Z. Z., Ma, J., Zhang, X., Hao, N., Yan, A., Nourbakhsh, A., Yang, X., McAuley, J., Petzold, L., & Wang, W. Y. (2024, May 2). A survey on large language models for critical societal domains: finance, healthcare, and law. arXiv.org. https://arxiv.org/abs/2405.01769v2
[84]. Liu, L., Yang, X., Lei, J., Shen, Y., Wang, J., Wei, P., Chu, Z., Qin, Z., & Ren, K. (2024, June 6). A survey on medical large language models: technology, application, trustworthiness, and future directions. arXiv.org. https://arxiv.org/abs/2406.03712v2
[85]. Nascimento, N., Guimaraes, E., Chintakunta, S. S., & Boominathan, S. A. (2024, November 16). LLM4DS: Evaluating Large Language Models for Data Science Code Generation. arXiv.org. https://arxiv.org/abs/2411.11908
[86]. Wu, Z., Wang, J., Cheng, Z., Zhou, Y., Wang, W., Pu, J., Li, C., & Ma, C. (2025, December 1). ICAD-LLM: One-for-All Anomaly Detection via In-Context Learning with Large Language Models. arXiv.org. https://arxiv.org/abs/2512.01672
[87]. Yang, T., Nian, Y., Li, L., Xu, R., Li, Y., Li, J., Xiao, Z., Hu, X., Rossi, R. A., Ding, K., Hu, X., & Zhao, Y. (2025). AD-LLM: Benchmarking Large Language Models for Anomaly Detection. In Findings of the Association for Computational Linguistics: ACL 2025 (2025.findings-acl.79, Issue 2025.findings-acl.79, pp. 1524–1547). Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.findings-acl.79
[88]. Cherkaoui, S. (2020). Large language models for time series anomaly detection and explanation [M.S. Thesis, MIT]. https://dai.lids.mit.edu/large-language-models-for-time-series-anomaly-detection-and-explanation/
[89]. Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. Psychological Bulletin, 76(5), 378–382. https://doi.org/10.1037/h0031619
[90]. Levenshtein, V.I. (1965). Binary codes capable of correcting deletions, insertions, and reversals. Soviet physics. Doklady, 10, 707-710.
[91]. Liu, F. T., Ting, K. M., & Zhou, Z. (2008). Isolation Forest. In 2008 Eighth IEEE International Conference on Data Mining (pp. 413–422). IEEE. https://doi.org/10.1109/icdm.2008.17
[92]. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., . . . Scialom, T. (2023). Llama 2: Open foundation and Fine-Tuned chat models. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2307.09288
[93]. Mone, V., & Sivakumar, C. (2022). An analysis of the GDPR compliance issues posed by new emerging technologies. Legal Information Management, 22(3), 166–174. https://doi.org/10.1017/s1472669622000317
[94]. Article 44 – General principle for transfers, GDPR (EU 2016/679). General Data Protection Regulation. Article 44 ensures that personal data transfers to third countries or international organisations may only occur if conditions in Chapter V are met to maintain the level of data protection. https://gdpr-info.eu/art-44-gdpr/
[95]. Abdullah, A. A., Zubiaga, A., Mirjalili, S., Gandomi, A. H., Daneshfar, F., Amini, M., Mohammed, A. S., & Veisi, H. (2025, October 14). Evolution of meta's llama models and parameter-efficient fine-tuning of large language models: a survey. arXiv.org. https://arxiv.org/abs/2510.12178
[96]. Yasmin, J., Tian, Y., & Yang, J. (2020, August 28). A first look at the deprecation of RESTful APIs: an

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jan689

Empirical study. arXiv.org. https://arxiv.org/abs/2008.12808

[97]. ServiceNow. (2025). ITOM Visibility architecture white paper. ServiceNow, Inc. https://www.servicenow.com/content/dam/servicenow-assets/public/en-us/doc-type/resource-center/white-paper/wp-itom-visibility-security-overview.pdf

[98]. Snoek, J., Larochelle, H., & Adams, R. P. (2012, June 13). Practical Bayesian optimization of machine learning algorithms. arXiv.org. https://arxiv.org/abs/1206.2944

[99]. Atlassian. (2023, April 18). Unleashing the power of AI across Atlassian's products. https://www.atlassian.com/blog/announcements/unleashing-power-of-ai

[100]. Pasuksmit, J., Takerngsaksiri, W., Thongtanunam, P., Tantithamthavorn, C., Zhang, R., Wang, S., Jiang, F., Li, J., Cook, E., Chen, K., & Wu, M. (2025, April 25). Human-In-The-Loop Software Development Agents: Challenges and future Directions. arXiv.org. https://arxiv.org/abs/2506.11009

[101]. Batini, C., Cappiello, C., Francalanci, C., & Maurino, A. (2015). Methodologies for data quality assessment and improvement. ACM Computing Surveys, 41(3), 1–52. https://doi.org/10.1145/1541880.1541883

[102]. Kappelman, L. A., McKeeman, R., & Zhang, L. (2006). Early warning signs of IT project failure: The dominant dozen. Information Systems Management, 23(4), 31–36. https://doi.org/10.1201/1078.10580530/46352.23.4.20060901/95110.4

[103]. Soomro, Z. A., Shah, M. H., & Ahmed, J. (2016). Information security management needs more holistic approach: A literature review. International Journal of Information Management, 36(2), 215–225. https://doi.org/10.1016/j.ijinfomgt.2015.11.009

[104]. Joshi, A., Kale, S., Chandel, S., & Pal, D. K. (2015). Likert scale: Explored and explained. British Journal of Applied Science & Technology, 7(4), 396–403. https://doi.org/10.9734/BJAST/2015/14975