Street Art & Urban Creativity | Vol. 10, No. 2, 2024 | ISSN 2183-9956
Street Art & Urban Creativity. Scientific Journal

AI-Enabled Clinical Data Management

From Manual Data Cleaning to Intelligent Data Operations

KEERTHI MANNE[1]
[1]502 PIN OAK CT, SPRINGCITY, PA-19475.

Abstract

Clinical data management (CDM) ensures that trial data are accurate, complete, and traceable, yet much of it still depends on manual cleaning: programmed edit checks, source data verification, query tracking, hand coding of medical terms, and a laborious path to database lock. Revised good clinical practice guidance and new regulatory principles for artificial intelligence (AI) encourage risk-proportionate, technology-enabled quality management. This paper proposes a framework for intelligent data operations that combines a maturity model, an architecture for AI-assisted detection, prioritization, and coding, a CRISP-DM-based lifecycle, and a credibility-based governance process, and it tests three components in a simulation. The simulation uses a synthetic trial of about 1,500 participants at 40 sites with about 84,000 data fields, 2% of which were corrupted, and a synthetic adverse-event dictionary; it demonstrates the method and does not report real trial data. At the same review budget as a set of diligent rule-based edit checks (about 1,213 queries per trial, precision 75.4%, recall 54.8%), a residual-based anomaly model reached precision 92.1% and recall 66.9%, and it matched the rules' recall with 24% fewer queries. The model was weakest for digit-level transcription errors (recall 57%) and blind to copy-forward errors, which rules also missed. Statistical site monitoring detected all three anomalous sites in every trial, with about eight additional flags per trial. Fuzzy dictionary matching auto-coded 69.9% of terms at 98.7% accuracy, against 29.4% for exact matching, but it performed poorly on novel phrasing (12.6% accuracy). The results support a hybrid design in which deterministic checks remain, AI ranks and suggests, and people decide.

Keywords: Clinical Data Management, Artificial Intelligence, Data Cleaning, Anomaly Detection, Medical Coding, Central Statistical Monitoring, Risk-Based Quality Management, ICH E6(R3).

1. Introduction

Clinical data management (CDM) turns what is recorded at the bedside into a dataset that can support a regulatory decision. Figure 1 summarizes its main components: a clinical data management system, a data management plan, case report forms, a data validation manual, data coding, database design, and database lock. Each of these steps exists to protect data quality, and each has traditionally relied on people: data managers write edit checks, review listings, issue and track queries, code adverse events and medications against standard dictionaries, and reconcile open issues before the database is locked [1].

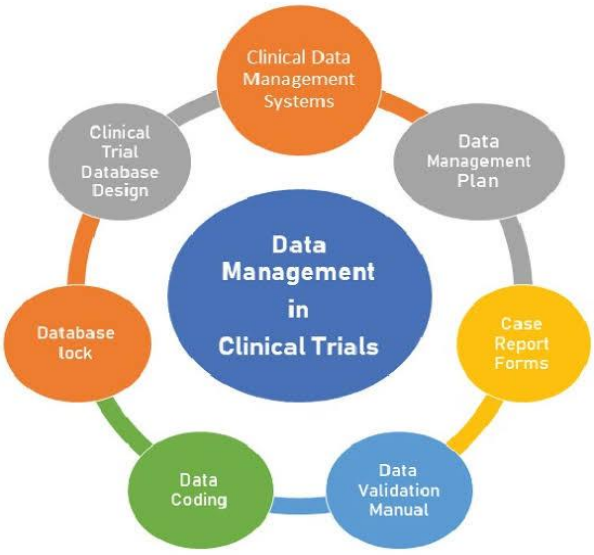


Figure 1: Data management in clinical trials, showing clinical data management systems, the data management plan, case report forms, the data validation manual, data coding, database lock, and clinical trial database design as the connected components of the CDM process.

Much of this work is manual data cleaning, and its burden is growing as trials draw on electronic data capture, central laboratories, patient-reported outcomes, wearable sensors, and electronic health records. Programmed edit checks are fast but crude: a range check finds a systolic blood pressure of 1,280 mmHg but cannot find a plausible value of 182 that should have been 128. Exhaustive source data verification is costly, and studies suggest that it changes only a small fraction of data [2], [3]. Regulators have responded by endorsing risk-proportionate approaches. The revised ICH guideline for good clinical practice, E6(R3), adds a data governance section and stresses proportionate, risk-based quality management [4], following earlier guidance that accepted centralized monitoring in place of routine full on-site verification [5], [6].

Artificial intelligence (AI) offers a way to move from reviewing everything to reviewing what matters. Models can learn what a plausible record looks like, rank suspicious values, flag atypical sites, and suggest dictionary terms for adverse events. Regulators have begun to set expectations for such tools: the U.S. Food and Drug Administration (FDA) issued draft guidance in 2023 on AI used to support regulatory decision-making for drugs and biologics, built on a risk-based credibility framework [7], and in January 2023 the FDA and the European Medicines Agency (EMA) jointly published ten guiding principles of good AI practice in drug development [8]. The draft guidance does not cover uses limited to operational efficiency that do not affect patient safety, drug quality, or the reliability of study results, so whether a data-cleaning model is in scope depends on its context of use [7].

This paper asks three questions. First, how should AI be organized within CDM so that it supports, and does not undermine, data integrity and regulatory expectations? Second, how much better than conventional edit checks can AI-based error detection be at a fixed review effort, and for which types of error? Third, how far can AI go in coding and site monitoring before human review is needed? The contributions are: (i) a framework for intelligent data operations with a maturity model, architecture, lifecycle, and governance process; (ii) a reproducible simulation that compares edit checks, statistical outlier detection, residual-model anomaly detection, and a hybrid score at equal review budgets; (iii) an evaluation of site-level statistical monitoring and AI-assisted medical term coding; and (iv) an effort model that translates detection and coding performance into review hours under stated assumptions. All experiments use synthetic data.

2. LITERATURE REVIEW

A. The Traditional CDM Lifecycle and Its Standards

The CDM lifecycle is governed by a mature body of practice. The Society for Clinical Data Management publishes good clinical data management practices that describe database design, validation, discrepancy management, coding, and lock [1]. The Clinical Data Interchange Standards Consortium defines standards for data collection and tabulation (CDASH and SDTM) that make datasets interoperable and reviewable [9]. Regulation requires validated

computerized systems with secure, time-stamped audit trails [10], [11], and ICH E6(R3) makes data governance across the data life cycle an explicit sponsor responsibility [4]. These foundations matter for AI: any intelligent component must operate inside this controlled environment and leave a trace that an inspector can follow. Table 1 maps the main CDM activities to their manual practice and to the AI-enabled alternatives considered in this paper.

TABLE 1. Mapping of CDM activities from manual practice to AI-enabled approaches

CDM activity	Manual practice	AI-enabled approach	Control retained
Edit-check specification	Checks written by hand from the protocol	Drafting of check specifications from protocol text	Human approval; traceability to protocol
Data validation	Fixed range and consistency checks; listing review	Deterministic checks plus ML anomaly scores for ranking	Deterministic checks retained; documented thresholds
Query management	Queries written and tracked by hand	Prioritized queue; drafted queries with reason codes	Data manager approves queries
Monitoring and source verification	Extensive on-site verification	Central statistical monitoring to target visits and sites	Risk-based plan; documented triggers
Medical coding	Coder assigns each term	AI-suggested terms with confidence; auto-coding above threshold	Coder approval; sampled quality control
Database lock	Checklists and manual reconciliation	Lock-readiness dashboard with open-issue tracking	Sign-off by accountable roles

B. Data Quality and Error in Clinical Trials

Data quality is multidimensional. A harmonized terminology distinguishes conformance (values follow specified formats and constraints), completeness, and plausibility (values are believable given other data), and each calls for different checks [12], [13]. Error rates depend on how data are captured and transcribed: electronic data capture changes where errors occur, and studies of data abstracted from medical records show that accuracy depends on the complexity of the item and on the abstractor [14], [15]. Importantly, many errors are plausible values that lie within the physiological range and are therefore invisible to range checks, which is the central limitation of rule-based cleaning. We quantify errors as the number per 10,000 fields inspected (Eq. 9) so that results can be compared across studies of different size.

C. Risk-Based Quality Management and Central Monitoring

The argument for risk-based quality management is that effort should be concentrated on the data and processes critical to participant safety and result reliability [16], [17]. Regulators accept centralized monitoring as a way to detect problems that are not visible at a single site [5], [6]. Statistical central monitoring uses the data themselves to find atypical centers, for example sites with unusually low variability, preference for certain terminal digits, or implausible duplication, which can indicate poor measurement or data fabrication [18], [19]. These methods are the precursors of AI-based data operations and fit naturally into the architecture proposed here.

D. AI Methods for Data Cleaning and Coding

Data cleaning has long been studied in the database community, which classifies errors and surveys detection and repair techniques [20], [21]. Anomaly detection supplies the statistical tools, from robust z-scores to tree-based methods such as isolation forests [22], [23]. A practical approach for tabular clinical data is to predict each value from the others and to treat a large prediction residual as evidence of error; gradient-boosted trees are a strong learner for this purpose [24]. Interactive cleaning methods learn from human corrections [25]. For coding, adverse event terms are mapped to the Medical Dictionary for Regulatory Activities (MedDRA) [26], and natural language processing algorithms for mapping clinical text to ontology concepts range from string matching to pre-trained language models [27], [28]. Large language models have also been evaluated for other trial operations such as matching patients to trials [29]. Their use in cleaning and coding raises the same questions of reliability and oversight addressed below.

E. Regulatory Expectations for AI

Three developments define the regulatory environment. ICH E6(R3), adopted in January 2023 and published by the FDA as final guidance in September 2023, asks that systems be fit for purpose and that quality be built into trial design [4]. The FDA draft guidance of January 2023

proposes a seven-step credibility process: define the question, define the context of use, assess the risk of the model, plan, execute, and document credibility activities, and determine whether the model is adequate for its purpose [7]. The EMA reflection paper of 2023 addresses AI across the medicinal product life cycle [30], and the joint FDA and EMA principles of 2023 stress human-centric design, a risk-based approach, adherence to standards, a clear context of use, data governance and documentation, performance assessment, and life cycle management [8]. Human factors also matter: users of decision support tend to over-rely on automated advice [31], and AI tools often disappoint when deployment context is ignored [32].

3. RELATED WORK

Related work falls into three groups. Statistical central monitoring has been developed and used for years to find atypical sites [18], [19], but it focuses on sites and not on individual values. Machine learning for data cleaning is well established in databases and in general anomaly detection [21], [22], but it is rarely evaluated in clinical trial settings against the edit checks that data managers actually use. AI-assisted coding and text normalization have progressed with biomedical language models [27], [28], but evaluations seldom report how performance varies between terms that resemble the dictionary and terms that do not. Regulatory documents describe what is expected of AI in drug development [7], [8] but do not tell a data manager how much review effort a model saves.

Four gaps follow. Comparisons with rule-based checks at equal review effort are rare. Performance is seldom broken down by error type, although that breakdown determines where a model can safely replace or supplement rules. Detection, site monitoring, and coding are usually studied separately, so there is no view of a CDM process as a whole. And effort is seldom expressed in hours under explicit assumptions. We address these gaps with an integrated framework and a simulation that reports equal-budget comparisons, error-type breakdowns, and an effort model.

4. METHODOLOGY

A. Intelligent Data Operations and a Maturity Model

We use the term intelligent data operations for CDM in which AI helps detect, prioritize, draft, and in limited cases resolve data issues, within a validated and governed system. Because autonomy should increase only as evidence accumulates, we define a five-level maturity model (Table 2). Level 0 is manual review; level 1 adds programmed rules; level 2 uses statistical or machine learning models to rank candidate errors and sites; level 3 lets AI draft queries and suggest codes with human approval; and level 4 lets AI act on low-risk items within pre-agreed limits, with people auditing samples and exceptions. The experiments in this paper cover levels 1 to 3.

TABLE 2. Maturity model for intelligent data operations

Level	Description	Typical CDM tasks	Human role
0 Manual	Review of listings and ad hoc checks	Spreadsheet review	Performs all review
1 Rule-based	Programmed, validated edit checks	Range, consistency, date checks	Writes rules; reviews every flag
2 ML-prioritized	Models rank candidate errors and sites	Anomaly scores; central monitoring	Reviews the ranked queue
3 AI-assisted	AI drafts queries and suggests codes	Query drafting; coding suggestions	Approves actions or samples
4 Supervised autonomy	AI resolves low-risk items within limits	Auto-closing trivial queries; auto-coding	Audits samples and exceptions

B. A CRISP-DM Lifecycle for AI in CDM

AI components need a development lifecycle as much as database builds do. We use the cross-industry standard process for data mining (CRISP-DM) [33], [34], shown in Figure 2, because it makes explicit the iterative relationship among business understanding, data understanding, data preparation, modeling, evaluation, and deployment. Applied to CDM, business understanding means defining the data quality objective and the critical-to-quality data, which corresponds to the question of interest and context of use in the FDA credibility process [7]. Data understanding is profiling of the trial data and their error patterns. Data preparation covers mapping to standards, handling of missing values, and construction of features such as the

median of other visits. Modeling is the selection and training of detectors and coders. Evaluation compares models with the incumbent rules and decides whether the model is adequate. Deployment includes planning for monitoring and maintenance, which in CDM means drift surveillance and change control. The arrows in Figure 2 return from evaluation and deployment to business understanding, reflecting the life cycle management expected of AI in drug development [8].

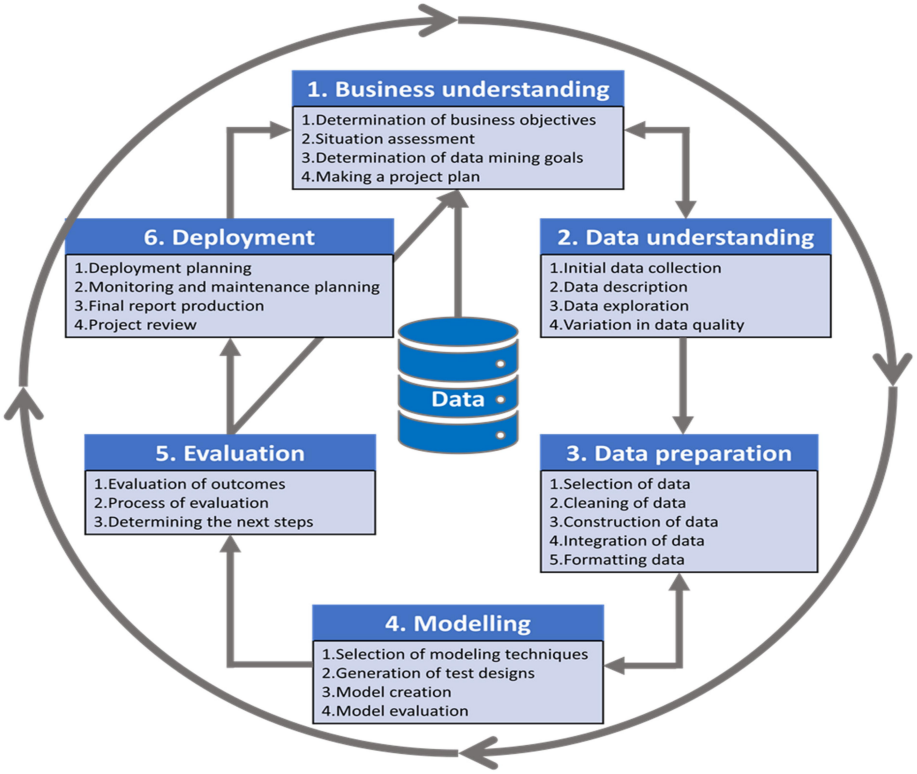


Figure 2: CRISP-DM process model with six phases: business understanding (business objectives, situation assessment, data mining goals, project plan), data understanding (initial data collection, description, exploration, data quality), data preparation (selection, cleaning, construction, integration, formatting), modelling (technique selection, test design, model creation, model evaluation), evaluation (evaluation of outcomes, process review, next steps), and deployment (planning, monitoring and maintenance, final report, project review), all centred on data.

C. System Architecture

Figure 3 shows the proposed architecture. Data from electronic data capture, central laboratories, patient-reported outcomes and wearables, and health records are ingested and mapped to standards, with metadata and an audit trail. A detection layer combines deterministic edit checks, machine learning anomaly scores, and site-level statistical monitoring. Outputs feed a prioritized query queue with reason codes, which data managers and sites review. A coding assistant suggests dictionary terms with a confidence score, and coders approve them. A lock-readiness dashboard tracks open queries and uncoded terms. Governance spans the whole lifecycle: context of use, credibility assessment, validation, audit trail, drift monitoring, human oversight, and change control. Deterministic checks are deliberately retained because their behavior is transparent, reproducible, and easy to validate, and because some errors (for example, a physiologically impossible value) should always generate a query.

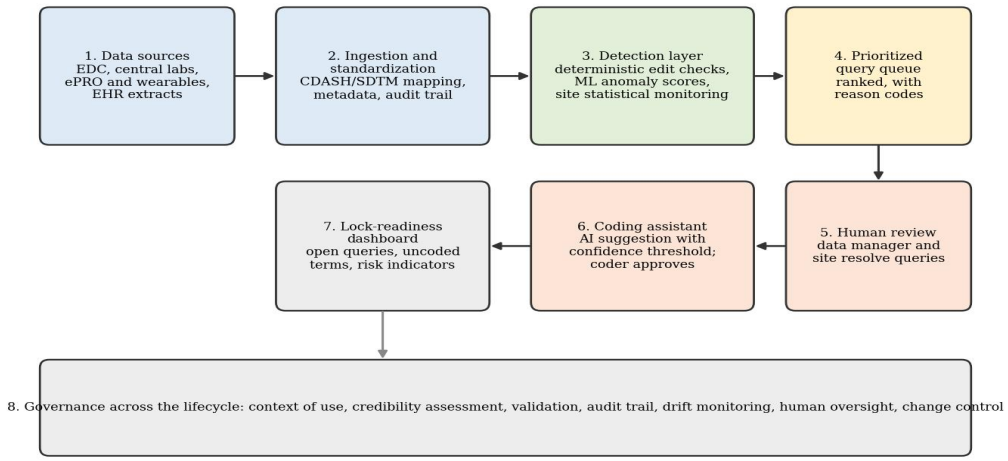


Figure 3: Proposed architecture for intelligent data operations: data sources, ingestion and standardization, a detection layer combining edit checks, ML anomaly scoring and site monitoring, a prioritized query queue, human review, an AI coding assistant, a lock-readiness dashboard, and lifecycle governance.

D. Error Detection Methods

Four detectors were compared. (1) Rule-based edit checks represent a diligent manual configuration: wide physiological range checks, two-sided cross-visit jump checks, which flag a value only if it differs from the previous and the next visit by more than a variable-specific percentage, and visit-date logic. Thresholds were set from general clinical conventions and not tuned. (2) A univariate robust z-score flags values far from the global median. (3) A residual model predicts each value from the other measurements at the same visit, the visit number, the median of the participant's other visits, and the previous and next values, using gradient-boosted trees [24], [35] with five-fold cross-fitting by participant so that a record is never scored by a model trained on it. The anomaly score is the robust standardized residual,

$$z_{ij} = \frac{|x_{ij} - \hat{x}_{ij}|}{\hat{\sigma}_j}, \qquad \hat{\sigma}_j = 1.4826 \cdot median\,|r - median(r)| \quad (1)$$

where x_ij is the observed value of variable j for record i, x̂_ij the cross-fitted prediction, and r the vector of residuals for variable j. (4) A hybrid priority score adds the rule signals to the model score, so that hard rule violations (H) and cross-visit jumps (D) rise to the top of the queue:

$$S_{ij} = z_{ij} + 3\,H_{ij} + D_{ij} \qquad (2)$$

The weights were fixed in advance and not tuned. All detectors are compared at the same review budget, equal to the number of queries generated by the rule set, because in practice the scarce resource is reviewer time.

E. Site-Level Statistical Monitoring

Three site-level tests target patterns associated with poor data quality [18], [19]. The variance test compares each site's pooled within-participant variance s_s^2, with ν_s degrees of freedom, with the pooled variance of all sites s_p^2, flagging implausibly low variability:

$$\chi_s^2 = \frac{\nu_s s_s^2}{s_p^2} \quad (3)$$

Tests for the three variables (systolic and diastolic pressure and heart rate) were combined by Fisher's method. The terminal-digit test compares the observed counts O_d of last digits of systolic pressure with the uniform expectation E_d,

$$\chi^2 = \sum_{d=0}^{9} \frac{(O_d - E_d)^2}{E_d} \quad (4)$$

and a one-sided binomial test compares each site's frequency of repeated consecutive weight and temperature values with the pooled frequency. Across all 120 site-test combinations, the Benjamini–Hochberg procedure controls the false discovery rate at level q [36]; the i-th smallest of m p-values is rejected if

$$p_{(i)} \le \frac{i}{m}\,q \quad (5)$$

F. AI-Assisted Medical Term Coding

The coding assistant maps each verbatim adverse event term to the closest dictionary entry. Terms are represented by TF-IDF vectors of character n-grams (lengths 2 to 4), which tolerate typographical variation, and similarity is the cosine [37]:

$$sim(a, b) = \frac{a \cdot b}{\|a\| \, \|b\|} \quad (6)$$

A suggestion is accepted automatically if the best similarity s_i is at least a threshold τ; otherwise the term goes to a human coder. The auto-coding rate and the accuracy among auto-coded terms are

$$AC(\tau) = \frac{1}{N}\sum_{i=1}^{N} 1[s_i \geq \tau] \quad (7)$$

$$Acc(\tau) = \frac{\sum_i 1[\hat{y}_i = y_i, s_i \geq \tau]}{\sum_i 1[s_i \geq \tau]} \quad (8)$$

G. Credibility Assessment and Governance

Each AI component is described by its context of use and risk, following the FDA credibility process [7]. For the detector, the context of use is prioritizing candidate errors for human review, which carries low to moderate risk because a missed error is possible but a model never changes data on its own. For the coding assistant the context of use is suggesting codes, with moderate risk if auto-coding is enabled, because an error would propagate to safety tables unless sampled review catches it. The credibility plan comprises performance against the incumbent method at equal effort, breakdown by error type and critical-to-quality field, calibration of confidence, and a monitoring plan for drift. In line with joint regulatory principles [8], people remain accountable for decisions, and design features such as ranked queues with reason codes, forced review of exceptions, and sampling of auto-coded terms limit over-reliance [31].

5. EXPERIMENTAL SETUP

A. Synthetic Trial and Error Injection

We simulated a multi-site trial of about 1,500 participants at 40 sites (20 to 55 participants per site), each with six scheduled visits (days 0, 14, 28, 56, 84, and 112). Eight clinical variables (systolic and diastolic blood pressure, heart rate, weight, temperature, ALT, creatinine, and hemoglobin) and the visit day gave about 83,905 fields per trial. Participant-level baselines were drawn from a correlated multivariate normal distribution, visit-to-visit variation was added, half of the participants received a small downward trend in blood pressure, and sites had small random shifts. Errors were injected into 2% of fields (about 1,669 per trial, an initial rate of about 199 per 10,000 fields) according to the types in Table 3. Three sites were additionally given data-generating behaviors that are not field-level errors but are targets for central monitoring: site 8 had within-participant variability reduced to 30% for blood pressure and heart rate, site 20 had strong preference for terminal digits 0 in blood pressure, and site 32 had frequent copy-forward of weight and temperature.

TABLE 3. Injected error types in the synthetic trial (approximate shares of injected errors)

Error type	Share	Example	Visible to range checks?
Digit or magnitude perturbation	51%	SBP 128 entered as 182; 10–30% change	Only if the value leaves the range
Gross magnitude	20%	SBP 128 entered as 1280	Yes
Visit-date error	12%	Day 28 entered as day 82	Yes, through date logic
Copy-forward	10%	Value repeated from previous visit	No
Unit error	6%	Weight in pounds; temperature in °F; creatinine in µmol/L	Yes, when the value leaves the range

B. Coding Data

A synthetic adverse-event dictionary was built from 50 generic preferred terms (for example headache, nausea, and peripheral oedema), each with three synonyms, giving about 200 dictionary entries. It is not MedDRA, which is licensed. For each trial, 5,000 verbatim terms were generated with a skewed distribution of terms: 50% were the dictionary term itself, 30% a dictionary synonym, and 20% a novel phrasing not in the dictionary (for example, “room spinning” for dizziness). Noise was added: a character-level typographical error (25% of terms), a severity prefix (35%), a trailing phrase (25%), and upper case (20%). Exact matching was the baseline.

C. Metrics, Assumptions, and Replication

The error rate is the number of errors per 10,000 fields,

$$ER = \frac{N_{err}}{N_{fields}} \times 10^4 \quad (9)$$

and detection is measured by precision (query yield) and recall,

$$P = \frac{TP}{TP + FP}, \qquad R = \frac{TP}{TP + FN} \quad (10)$$

with the area under the precision–recall curve (AUPRC) used for continuous scores because errors are rare [38], [39]. At a budget of B queries, recall is

$$Recall@B = \frac{|Top_B \cap E|}{|E|} \quad (11)$$

where E is the set of true errors. Review effort is converted to hours with assumed times of t_q = 10 minutes per query and t_c = 1.5 minutes per manually coded term, with N_q queries and N_c manually coded terms,

$$H = \frac{N_q t_q + N_c t_c}{60} \quad (12)$$

and the residual error rate after cleaning assumes that a query resolves an error correctly with probability ρ = 0.9,

$$ER_{res} = \frac{N_{missed} + (1 - \rho)\, N_{caught}}{N_{fields}} \times 10^4 \quad (13)$$

The times and ρ are assumptions, not measurements, and the effort results should be read as illustrations. Each experiment was repeated for five independently generated trials, and means are reported. Code was written in Python with NumPy and scikit-learn [35] and is provided with this paper.

6. RESULTS

A. *Error Detection at Equal Review Effort*

The rule-based edit checks generated 1,213 queries per trial, about 14.5 per 1,000 fields. Their precision was 75.4% and their recall 54.8% (Table 4). At the same number of queries, the residual model reached precision 92.1% and recall 66.9%, and the hybrid score 91.3% and 66.4%, a difference of about 12 recall points. A univariate z-score was worse than the rules (precision 62.7%, recall 45.5%) and, because it cannot see visit dates, missed all date errors. The AUPRC of the residual model was 0.747 against 0.496 for the univariate score. The hybrid score gave no further gain in accuracy over the residual model, which is expected because the rules mostly find errors that the model also finds.

TABLE 4. Error detection at the rule-based query budget, and at twice that budget (mean of five trials; initial error rate about 199 per 10,000 fields; ρ = 0.9)

Method	Queries	Precision (%)	Recall (%)	F1	Critical-lab recall (%)	Residual errors per 10,000
Rule-based	1,213	75.4	54.8	0.634	42.7	100.9
Univariate z	1,213	62.7	45.5	0.527	48.7	117.5
Residual model	1,213	92.1	66.9	0.775	61.1	79.1
Hybrid	1,213	91.3	66.4	0.768	60.3	80.1
Residual model (2× budget)	2,425	51.1	74.3	–	69.8	66.0
Hybrid (2× budget)	2,425	51.0	74.1	–	69.5	66.3

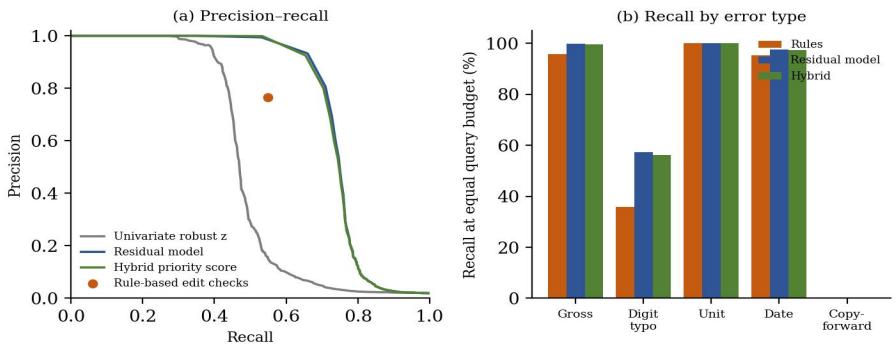


Figure 4: (a) Precision–recall curves for the univariate robust z-score, the residual model, and the hybrid priority score, with the rule-based edit checks shown as a single operating point. (b) Recall by error type at the rule-based query budget. Synthetic data; first of five trials in (a).

B. Effort at Equal Recall and Residual Error

Measured the other way, the residual model needed 922 queries and the hybrid 919 to reach the recall of the rules, compared with 1,213 for the rules, a reduction of about 24% (Figure 6b). The univariate z-score needed 5,486, about 4.5 times the rule-based effort. At twice the rule-based budget, the rules still reached only 55.5% recall, because rules cannot raise a flag for values that satisfy every rule, while the residual model reached 74.3% and the hybrid 74.1%. Under the assumed resolution probability, the residual error rate fell from about 199 per 10,000 fields at the start to 101 with rules, 79 with the residual model at the same budget, and 66 at twice the budget (Table 4).

C. Where Detection Fails

Recall depended strongly on the type of error (Figure 4b). Gross magnitude errors (about 96–100%), unit errors (100%), and date errors (95–98%) were found by both approaches. Digit or magnitude perturbations within the plausible range were the hardest class: rules found 35.8%, the residual model 57.2%, and the hybrid 56.2%, so over 40% of these errors escaped even the best detector at this budget. Copy-forward errors were almost never found by any method (below 1%) because a repeated value is, by construction, plausible and consistent with the participant's history. Detection in the critical laboratory fields (ALT, creatinine, and hemoglobin) was lower than overall: 61.1% for the residual model and 42.7% for the rules, which is unwelcome because these fields matter most for safety. These blind spots call for dedicated checks, such as repeated-value detection, and for retention of deterministic rules.

D. Site-Level Monitoring

The three site-level tests detected all three anomalous sites in every one of the five trials (Figure 6a): the variance test identified the low-variability site, the terminal-digit test the site with digit preference, and the repeated-value test the site with copy-forward. They also raised between 4 and 14 additional flags per trial (mean 8.4 out of 117 other site-test combinations), so about two thirds of flags were not among the three planted anomalies. Some of these flags may reflect chance excesses of the injected field-level errors at particular sites, which would still be a legitimate reason for a data manager to look, but they show that statistical monitoring output is a triage signal and not a verdict.

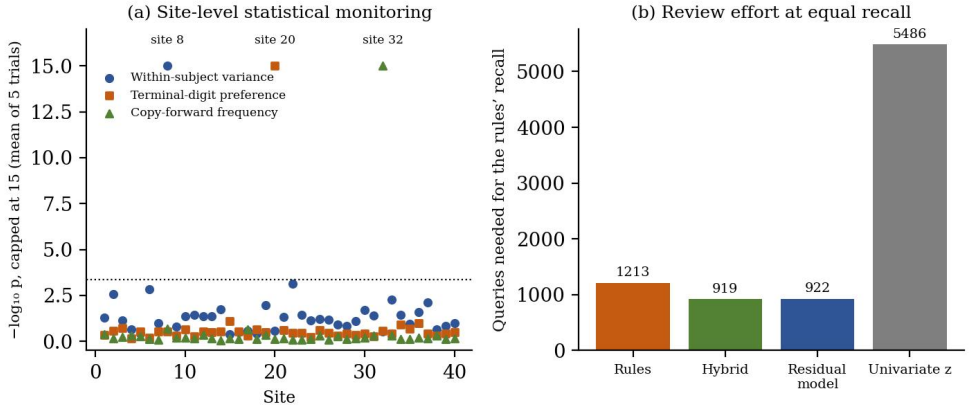


Figure 5: (a) Site-level statistical monitoring: $-\log_{10}$ p-values (capped at 15; mean over five trials) for the within-participant variance, terminal-digit, and repeated-value tests at each of 40 sites; sites 8, 20, and 32 carried planted anomalies and the dotted line is the Bonferroni level. (b) Queries needed to reach the recall of the rule-based edit checks. Synthetic data.

E. AI-Assisted Coding

Exact dictionary matching coded only 29.4% of the synthetic verbatim terms, because most terms carried typographical errors, qualifiers, or novel phrasing, although every term it coded was correct (Table 5). Character n-gram matching coded far more. At $\tau = 0.7$ it auto-coded 69.9% of terms with accuracy 98.7% among those, and at $\tau = 0.5$ it auto-coded 86.0% with accuracy 96.0% (Figure 5a). Raising the threshold traded coverage for accuracy: 38.1% coverage at 100% accuracy for $\tau = 0.9$.

TABLE 5. Auto-coding performance of exact and character n-gram matching by threshold τ (mean of five trials; 5,000 synthetic terms per trial)

Method	Auto-coded (%)	Accuracy of auto-coded (%)	Novel phrasing auto-coded (%)	Novel phrasing accuracy (%)
Exact dictionary match	29.4	100.0	–	–
Fuzzy match, τ = 0.5	86.0	96.0	39.8	61.0
Fuzzy match, τ = 0.6	77.2	97.8	11.0	28.8
Fuzzy match, τ = 0.7	69.9	98.7	5.1	12.6
Fuzzy match, τ = 0.8	56.6	99.2	2.1	0.0
Fuzzy match, τ = 0.9	38.1	100.0	0.0	–

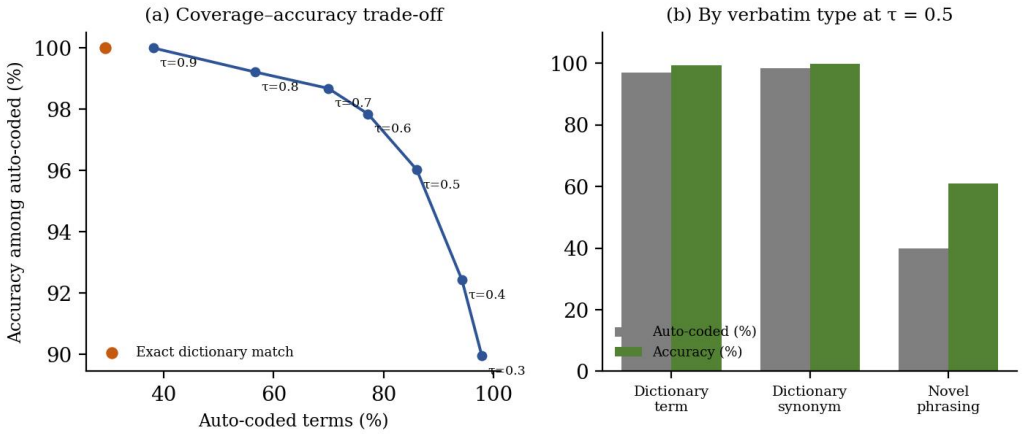


Figure 6: (a) Coverage–accuracy trade-off of fuzzy auto-coding at thresholds τ from 0.3 to 0.9, compared with exact dictionary matching. (b) Auto-coded share and accuracy by type of verbatim term at τ = 0.5. Synthetic data.

The aggregate hides a sharp contrast (Figure 6b). Dictionary terms and synonyms with noise were coded at 99.5% to 99.8% accuracy at τ = 0.5, but novel phrasings, which do not resemble any dictionary string, were auto-coded 39.8% of the time with only 61.0% accuracy, and at τ = 0.7 the 5.1% of novel terms that passed the threshold were coded correctly only 12.6% of the time. In other words, surface similarity handles typographical variation but not paraphrase, and the confidence score is not a reliable indicator of correctness for paraphrased terms. Semantic models are needed for these cases, together with coder review.

F. Effort Model

TABLE 6. Illustrative review effort under assumed times (10 minutes per query; 1.5 minutes per manually coded term; 0.5 minutes per sampled quality-control check)

Scenario (per simulated trial)	Items reviewed	Hours (assumed times)	Result
Query review: rule-based	1,213 queries	202	914 true errors found
Query review: residual model, same budget	1,213 queries	202	1,117 true errors found
Query review: residual model, rules' recall	922 queries	154	Same errors found as rules
Coding: all terms by hand	5,000 terms	125	No machine coding errors
Coding: auto at τ = 0.7, 10% sampled QC	1,504 by hand	41	About 41 undetected machine-coded errors
Coding: auto at τ = 0.8, 10% sampled QC	2,168 by hand	57	About 20 undetected machine-coded errors

Table 6 translates the results into hours under the stated assumptions. At the same query budget, the residual model finds about 203 more true errors than the rules; alternatively, it matches the rules' yield with about 48 fewer review hours (about 24%). For coding, auto-coding at τ = 0.7 reduces manual effort from 125 to about 41 hours, at the cost of residual machine-coding errors that sampled review only partly detects. These numbers depend on assumed times and error-resolution probabilities and should be replaced with a sponsor's own measurements.

7. DISCUSSION

The simulation supports four messages. First, AI adds value mainly by prioritization. At the same review effort, a model that predicts values from context found more errors with fewer

false queries than edit checks, particularly plausible-but-wrong values that satisfy every rule, and it did so without changing any data. This is the level 2 capability in the maturity model and the one with the clearest regulatory path, since a human still decides. The gain depends on the baseline, however: a cruder rule set would have looked worse, and a better tuned one better, and any sponsor should test models against its own edit checks.

Second, AI does not replace deterministic rules. The hybrid score performed as well as the residual model but kept hard rule violations at the top of the queue, which is useful because those queries are expected, auditable, and inexpensive to explain. Both methods were nearly blind to copy-forward errors and found only about 57% of digit-level transcription errors. A deployment should therefore keep deterministic checks, add dedicated checks for known blind spots, and report recall by error type and for critical-to-quality fields, as the credibility process asks [7].

Third, coding automation is useful but conditional. Confidence-thresholded matching handled the large share of terms that were dictionary entries with noise and cut manual effort substantially, but it failed for paraphrases, and its confidence score was a poor guide for those terms. A sensible design is to auto-code only above a conservative threshold, route everything else to a coder, show suggestions with confidence, and sample auto-coded terms for quality control. Semantic models of the kind developed for biomedical text [27], [28] could extend coverage, but their errors must be measured in the same way.

Fourth, central monitoring should be treated as triage. All planted anomalies were found, but about two thirds of site-level flags were not planted anomalies. A risk-based plan should state in advance which signals trigger a review, which trigger a visit, and which only get logged [16], [19]. More generally, the roles of data managers change: less time on routine review and more on specifying checks, interpreting signals, validating models, and overseeing exceptions. Automation bias is a real risk for reviewers who see a ranked list [31], which is why the architecture keeps reason codes, exception review, and sampling.

The study has important limitations. The data are synthetic, so effect sizes depend on our assumptions about error types, error rates, and noise, and they would differ in real trials, where the true error status of a field is rarely known and must be approximated by audit samples. The rule-based baseline was set by general conventions and not tuned. The models cover structured numeric data and not free text, images, or device streams. The effort estimates rest on assumed times and resolution probabilities. We did not evaluate large language models, query drafting, or level 4 autonomy, and the coding experiment used a small synthetic dictionary and not MedDRA. Future work should repeat the evaluation on retrospective data from completed trials with audited ground truth, extend detection to free text and sensor data, test semantic coding models, evaluate query drafting with human reviewers, and study drift and change control over the life of a study [8].

8. CONCLUSION

This paper presented a framework for moving clinical data management from manual cleaning to intelligent data operations and tested three of its components in a synthetic trial. The framework combines a five-level maturity model, an architecture that keeps deterministic checks alongside AI-based detection, prioritization, and coding, a CRISP-DM lifecycle for AI components, and a credibility-based governance process aligned with ICH E6(R3) and the 2023 and 2023 regulatory documents on AI. At the review budget of a diligent rule set, a residual-based model raised recall from 54.8% to 66.9% with precision of 92.1%, or alternatively matched the rules with 24% fewer queries. It remained weak for plausible digit-level errors and blind to copy-forward errors. Site monitoring found every planted anomaly together with additional flags that need triage, and character n-gram auto-coding covered 69.9% of terms at 98.7% accuracy but failed on novel phrasing.

The central implication is that AI in CDM should be introduced as supervised decision support whose value is demonstrated against the incumbent method, error type by error type. Deterministic rules, human review of exceptions, sampling of automated coding, and lifecycle monitoring remain essential. The authors recommend that sponsors pilot these components on retrospective trial data with audited ground truth, measure review effort directly, and expand autonomy only as the evidence from each level supports it.

References

1. A. Mahendra Vardhan and S. Sridhar, "Determining False Positive Analysis of Software Vulnerabilities with Predefined Scan Rules using Random Forest Classifier and Decision Tree Technique," 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2022, pp. 622-625, DOI: https://doi.org/10.1109/ICAC3N56670.2022.10074458
2. Dip Bharatbhai Pate, Dynamic vs. Static visualizations: Understanding their use cases in big data analysis, International Journal of Science and Research Archive, 2022, 05(02), 392-395., DOI https://doi.org/10.30574/ijsra.2022.5.2.0077
3. Dip Bharatbhai Patel, Building scalable business intelligence systems in the cloud: A technical approach, International Journal of Science and Research Archive, 2021, 02(01), 212-215., Received on 03 February 2021; revised on 17 April 2021; accepted on 19 April 2021,DOI: https://doi.org/10.30574/ijsra.2021.2.1.0047
4. Dip Bharatbhai Patel, Understanding the Impact of Text Analytics on Social Media Sentiment Analysis, Vol. 5 No. 5 (2019) ,September-October, DOI: https://doi.org/10.32628/CSEIT1936413
5. Dip Bharatbhai Patel, Real-Time Data Streaming in BI Dashboards: Opportunities and Challenges, Archives / Vol. 5 No. 4 (2019) : July-August, DOI: https://doi.org/10.32628/CSEIT192142
6. Dip Bharatbhai Patel, the Role of Data Visualization in Modern Intelligence, Archives / Vol. 2 No. 2 / Research Article, Published, 2020-11-28 DOI: https://doi.org/10.32996/jcsts.2020.2.2.4
7. A. Srivastava, "AI-assisted cloud migration of healthcare claims data from legacy systems: A metadata-driven framework," Int. J. Commun. Networks Inf. Secur., vol. 14, no. 3, Nov. 2022. [Online]. Available: https://ijcnis.org/index.php/ijcnis/article/view/8682
8. A. Srivastava, "Enhancing provider and claims data accuracy using artificial intelligence on cloud-native data platforms," J. Comput. Anal. Appl., vol. 31, no. 4, Nov. 2023. [Online]. Available: https://eudoxuspress.com/index.php/pub/article/view/4347/3189
9. A. Srivastava, "Optimizing large-scale data migration for cloud-native architectures," J. Comput. Anal. Appl., vol. 31, no. 2, pp. 652–662, May 2023. [Online]. Available: https://eudoxuspress.com/index.php/pub/article/view/4603
10. Backup Integrity and Recovery Readiness Assessment for High-Availability Databases Raghu Gollapudi, Computer Fraud & Security, Vol. 2023, Issue 8, pp. 35-48, 10.52710/cfs.1031
11. Risk-Controlled Near-Zero-Downtime Oracle Database Migration Using GoldenGate Raghu Gollapudi, International Journal of Computational and Experimental Science and Engineering, 8(3), 2022 ,10.22399/ijcesen.5382
12. Manoj Kumar Kagitha, Automated Security Compliance in Ci/Cd: A Natural Language Processing (NLP) Approach to Detecting Vulnerabilities in Infrastructure-As-Code (Terraform/Bicep),Publication year 2021, Volume 2021, Issue 12, https://computerfraudsecurity.com/index.php/journal/article/view/970
13. Manoj Kumar Kagitha, Self-Healing Cloud Infrastructure: A Hybrid Reinforcement Learning Framework for Predicting Microservice Failures in Azure Kubernetes Service (AKS),Publication year 2023, Volume 2023, Issue 12,https://computerfraudsecurity.com/index.php/journal/article/view/971
14. Manoj Kumar Kagitha, LOAD BALANCING TECHNIQUES IN DISTRIBUTED CLOUD ENVIRONMENTS, (IJESMR - 2017), D.O.I:10.53555/jcr.v6:i1.13380, https://jcreview.com/archives/volume-6/issue-1/13380
15. Manoj Kumar Kagitha, AI-DRIVEN OBSERVABILITY FOR HYPERSCALE COLOCATION DATA CENTERS: A CASE STUDY OF LOG ANALYTICS AND ANOMALY DETECTION PIPELINES, Archives / Vol. 27 No. 7 (2019): JOCAAA,https://www.eudoxuspress.com/index.php/pub/article/view/5053
16. Manoj Kumar Kagitha, GREENOPS IN THE CLOUD: AN AI-DRIVEN TELEMETRY ANALYSIS MODEL FOR REDUCING CARBON FOOTPRINT IN HIGH-AVAILABILITY CLUSTERS, Vol. 28 No. 6 (2020): JOCAAA, https://www.eudoxuspress.com/index.php/pub/article/view/5054
17. Naresh Lokiny. (2022). Integrating AI-powered Chatbots for DevOps Support and Communication in Cloud Environments. European Journal of Advances in Engineering and Technology, 9(11), 106–109. https://doi.org/10.5281/zenodo.13325989
18. Naresh Lokiny, (2021), "Disaster Recovery and Business Continuity Planning in DevOps Cloud

with AI", International Journal of Science and Research (IJSR), 10(3), 2023-2027. https://dx.doi.org/10.21275/SR24724151733, https://www.ijsr.net/getabstract.php?paperid=SR24724151733
19. Naresh Lokiny, & Ranganath Nandanampati. (2020). DevSecOps: Integrating Security into DevOps with AI in Cloud. Journal of Scientific and Engineering Research, 7(10), 239–242. https://doi.org/10.5281/zenodo.13348695
20. Naresh Lokiny, & Pradip Reddy. (2021). Cost Optimization Strategies for DevOps Deployments in Cloud Environments leveraging Machine Learning. European Journal of Advances in Engineering and Technology, 8(3), 69–72. https://doi.org/10.5281/zenodo.13325845
21. Jayanth Para, AI Based Cloud Computation Observational Method & Process, Vol. 51 No. 4 (2023): October-December 2023, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/171 , DOI: https://doi.org/10.46121/pspc.51.4.3
22. Jobayar Alom, Ahsan Ahmed. (2023). Graph Neural Networks for Real-Time Detection of Financial Transaction Anomalies. Acta Scientiae, 24(5), 82–90. https://doi.org/10.22178/acta.24.5.6
23. Kaleshwar Aryasomayajula, PREVENTING BIAS IN AI-BASED DECISION-MAKING: ANALYZING TECHNIQUES TO REMOVE UNFAIR PREJUDICE IN ALGORITHMIC OUTCOMES, Vol. 50 No. 2 (2022): April-June 2022, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/294
24. Kaleshwar Aryasomayajula, MODERN DEVELOPMENT PRACTICES: INVESTIGATIONS INTO SERVERLESS COMPUTING, LOW-CODE/NO-CODE PLATFORMS, AND DEVELOPER PRODUCTIVITY IN REMOTE ENVIRONMENTS, Vol. 50 No. 4 (2022): October-December 2022, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/364
25. Vikas Suresh Bagora, KERNEL-LEVEL PERFORMANCE OPTIMIZATION FOR CARRIER-GRADE BROADBAND AND HIGH-SPEED NETWORKING SYSTEMS, Vol. 47 No. 1 (2019), Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/359
26. Vikas Suresh Bagora, SCALABLE 128G FIBRE CHANNEL AND ETHERNET CONVERGENCE FOR NEXT-GENERATION DATA CENTER NETWORKS, Vol. 50 No. 4 (2022): October-December 2022, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/362
27. Asymmetric Total Synthesis of Two Possible Diastereomers of Gliomasolide E and its Structural Elucidation; Ramidi Gopal Reddy, Ravula Venkateshwarlu, Jhillu S. Yadav and Debendra K. Mohapatra. (J. Org. Chem. 2017, 82(2), 1053-1063). https://doi.org/10.1021/acs.joc.6b02611 , https://pubs.acs.org/doi/abs/10.1021/acs.joc.6b02611
28. Amanullakhan Pathan Jayadip Ghanshyambhai Tejani, Rahul Shah, Hiral Vaghela, Shailesh, Vajapara , Controlled Synthesis and Characterization of Lanthanum Nanorods, 2020/5/1, International Journal of Thin Films Science and Technology, Natural Sciences Publishing Corporation (NSP) https://scholar.google.com/citations?view_op=view_citation&hl=en&user=efbNMkwAAAAJ&citation_for_view=efbNMkwAAAAJ:M3NEmzRMIkIC
29. Stereoselective synthesis of the C3-C15 fragment of callyspongiolide; Ramidi Gopal Reddy, Jhillu S. Yadav and Debendra K. Mohapatra. (Tetrahedron Letters. 2018, 59, 3579-3582). https://doi.org/10.1016/j.tetlet.2018.08.041, https://www.sciencedirect.com/science/article/pii/S0040403918310426
30. Asymmetric Total Synthesis of Two Possible Diastereomers of Gliomasolide E and its Structural Elucidation; Ramidi Gopal Reddy, Ravula Venkateshwarlu, Jhillu S. Yadav and Debendra K. Mohapatra. (J. Org. Chem. 2017, 82(2), 1053-1063). https://doi.org/10.1021/acs.joc.6b02611, , https://pubs.acs.org/doi/abs/10.1021/acs.joc.6b02611
31. Kaleshwar Aryasomayajula, PREVENTING BIAS IN AI-BASED DECISION-MAKING: ANALYZING TECHNIQUES TO REMOVE UNFAIR PREJUDICE IN ALGORITHMIC OUTCOMES, Vol. 50 No. 2 (2022): April-June 2022, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/294
32. Kaleshwar Aryasomayajula, MODERN DEVELOPMENT PRACTICES: INVESTIGATIONS INTO SERVERLESS COMPUTING, LOW-CODE/NO-CODE PLATFORMS, AND DEVELOPER PRODUCTIVITY IN REMOTE ENVIRONMENTS, Vol. 50 No. 4 (2022): October-December 2022, Power System Protection and Control, ISSN-1674-3415,

https://pspac.info/index.php/dlbh/article/view/364
33. Kaleshwar Aryasomayajula, Micro services: "A Comparative Analysis of Monolithic vs. Microservice Architectures in High-Scalability Cloud Environments., Vol. 51 No. 2 (2023): April-June 2023 , Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/374
34. Controlled Synthesis and Characterization of Lanthanum Nanorods, Author(s), Jayadeep Tejani, Rahul Shah, Hiral Vaghela, Shailesh Vajapara, Amanullakhan Pathan, doi:10.18576/ijtfst/090205, URL: https://www.naturalspublishing.com/Article.asp?ArtcID=20705, May-2020
35. Sudipkumar Ghanvat , Aishwarya Badlani, Shreya Sandeep Joshi, Marketing Effectiveness in the Post-Cookie Era: A Comparative Analysis of First Party and Zero-Party Data Strategies on Campaign Performance and Customer Equity, Vol. 31 No. 4 (2023): JOCAAA , https://www.eudoxuspress.com/index.php/pub/article/view/3940
36. Chander Vijay S Sanbhi, BAYESIAN UPDATING OF RAM MODELS FOR DYNAMIC AVAILABILITY FORECASTING UNDER REALISTIC DOWNTIME CONSTRAINTS, Vol. 51 No. 3 (2023): July-September 2023, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/381
37. Chander Vijay S Sanbhi, HYBRID RELIABILITY MODELLING FOR ROTATING EQUIPMENT: PHYSICS- INFORMED LEARNING WITH SPARSE FAILURE DATA, Vol. 51 No. 4 (2023): October-December 2023, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/382
38. A. Srivastava, "AI-assisted cloud migration of healthcare claims data from legacy systems: A metadata-driven framework," Int. J. Commun. Networks Inf. Secur., vol. 14, no. 3, Nov. 2022. [Online]. Available: https://ijcnis.org/index.php/ijcnis/article/view/8682
39. A. Srivastava, "Enhancing provider and claims data accuracy using artificial intelligence on cloud-native data platforms," J. Comput. Anal. Appl., vol. 31, no. 4, Nov. 2023. [Online]. Available: https://eudoxuspress.com/index.php/pub/article/view/4347/3189
40. A. Srivastava, "Optimizing large-scale data migration for cloud-native architectures," J. Comput. Anal. Appl., vol. 31, no. 2, pp. 652–662, May 2023. [Online]. Available: https://eudoxuspress.com/index.php/pub/article/view/4603
41. Santhosh Chitraju Gopal Varma, The Evolution of Cloud-Native Architectures: Exploring the Synergy between Kubernetes and Microservices, DOI: https://doi.org/10.63282/3050-9246.IJETCSIT-V1I4P104, Vol. 1 No. 4 (2020), https://ijetcsit.org/index.php/ijetcsit/article/view/108
42. Santhosh Chitraju Gopal Varma, Artificial Intelligence in Cloud Computing: Building Intelligent, Distributed, and Fault-Tolerant Systems, Vol. 3 No. 1 (2022), DOI: https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I1P105, https://ijaibdcms.org/index.php/ijaibdcms/article/view/78
43. Sunil Karmakar, AI driven Identity & Access Management with Oracle Cloud security, Vol. 30 No. 2 (2022): JOCAAA, https://eudoxuspress.com/index.php/pub/article/view/4118
44. Bandari, V. V., & Lekkala, H. B. (2023). Physics-informed reinforcement learning for real-time control of complex manufacturing processes. Power System Protection and Control, 52(2), 164–170. https://pspac.info/index.php/dlbh/article/view/343
45. Bandari, V. V., & Lekkala, H. B. (2023). AI-based operator behavior monitoring and cost optimization using digital traceability in manufacturing systems. Power System Protection and Control, 52(1), 38–45. https://pspac.info/index.php/dlbh/article/view/342
46. Kancharla Vijayalakshmi, Bethapudi Samuel Anand Andrews, Estimation of Cyproheptadine Hydrochloride and Tricholine Citrate Simultaneously in Syrup Dose by Applying Stability Indicating HPLC Methodology, December 31, 2020, Vol. 11, Issue 4, Pages 1813–1819, DOI: 10.26452/ijrps.v11iSPL4.4383, https://ijrps.com/v11si4/3683
47. Vijayalakshmi Kancharla ,Kumar Abbaraju V D N, Andrews B S A, Nagaraju Dasari, Nagu Korupolu, Melphan as Anticancer and New Insights of Drug as Parenteral Dosage form against SARS-CoV-2 by Using Development and Validation , November 18, 2020, Vol. 11, Issue 1, Pages 1437–1446
DOI: 10.26452/ijrps.v11iSPL1.3686, https://ijrps.com/v11si1/2911
48. Ramesh, D., Singh, A., & Channaiah, L. H. (2023). The importance of validated kill steps and thermal death kinetics of Salmonella in food safety management. Global Journal of Nutrition & Food Science, 5(1), Article 000601. https://doi.org/10.33552/GJNFS.2023.05.000601

49. Bharati Reddi, Chandan Kishor,Aruna Jangam, Sandeepchowdary Bala, Uma Rajeswari Batchu ,Rambabu Gundla , Anthony Addlagatta, Regioselectivity in inhibition of peptide deformylase from Haemophilus influenzae by 4- vs 5-azaindole hydroxamic acid derivatives: Biochemical, structural and antimicrobial studies, Date: November 2022, Article: 106095, Volume: Volume 128, 10.1016/j.bioorg.2022.106095, https://www.sciencedirect.com/science/article/abs/pii/S0045206822005016
50. Bhavita Kattula, Bharati Reddi, Aruna Jangam, Lekhika Naik , Bala Manikanta Adimoolam , Suresh Vavilapalli , Sayanna Are , Jagadeshwar Reddy Thota, Surender Singh Jadav , Mohammed Arifuddin Anthony Addlagatta, Development of 2-chloroquinoline based heterocyclic frameworks as dual inhibitors of SARS-CoV-2 MPro and PLPro, Date: 1 July 2023 Article: 124772, Volume: Volume 242, Part 2, 10.1016/j.ijbiomac.2023.124772 https://www.sciencedirect.com/science/article/pii/S0141813023016665
51. Iram Irfan, Asghar Ali, Bharati Reddi, Mohd. Abrar Khan, Phool Hasan, Sarfraz Ahmed Amad Uddin, Magdalena Piatek, Kevin Kavanagh, Mohammad Abid, Design, Synthesis and Mechanistic Studies of Novel Isatin-Pyrazole Hydrazone Conjugates as Selective and Potent Bacterial MetAP Inhibitors, 19 August 2022, https://doi.org/10.3390/antibiotics11081126 https://www.mdpi.com/2079-6382/11/8/1126
52. GAURAV KUMAR SINHA, Efficient Test-Time Compute Scaling for Resource-Constrained Large Language Models Vol. 32 No. 1 (2023): JOCAAA, DOI: https://doi.org/10.1109/ICAC3N56670.2022.10074458 https://eudoxuspress.com/index.php/pub/article/view/5918
53. GAURAV KUMAR SINHA , The Impact of Artificial Intelligence on Predictive Quality Assurance in Modern Manufacturing, Vol. 33 No. 05 (2023): JOCAAA, https://doi.org/10.1109/ICAC3N56670.2022.10074458 , https://eudoxuspress.com/index.php/pub/article/view/5919
54. GAURAV KUMAR SINHA , Hybrid Quantum-Classical Neural Networks for Combinatorial Optimization and Machine Learning, Vol. 33 No. 08 (2023): JOCAAA, DOI: https://doi.org/10.1109/ICAC3N56670.2022.10074458, https://eudoxuspress.com/index.php/pub/article/view/5921
55. Lalith Chandra Bandaru , Mohammed Shakeer Bandrevu, GRAPHSEC: Graph-Based Supply Chain Attack Detection and Risk Propagation Analysis for Enterprise Salesforce Deployment Pipelines, Paper Id: 233135,Published On: 2023-11-03, Published In: Volume 11, Issue 6, November-December 2023, https://www.ijirmps.org/research-paper.php?id=233135 , DOI: https://doi.org/10.37082/IJIRMPS.v11.i6.233135