Math. Appl. 14(1)2025, 198–214

AI-Based Prediction of Clinical Trial Database Lock Readiness

Keerthi Manne[1]

Abstract: *Database lock is the point at which a clinical trial's data are declared final and statistical analysis can begin, and the weeks between the last patient's last visit and lock are among the most closely watched in drug development. Lock dates are planned months ahead, are frequently missed, and are usually forecast by hand from status reports of open queries, missing pages, coding and reconciliation backlogs. This paper asks whether machine learning applied to weekly snapshots of those backlogs can forecast the remaining time to lock, flag that a study will miss its plan, and declare readiness for lock with calibrated confidence. We build a reproducible simulation of a portfolio of 150 studies run by five contract research organisations, in which eight backlog categories are worked down at organisation-specific rates, interrupted by staffing dips, vendor data transfers and late safety findings, followed by a variable sign-off period, and in which the planned lock date uses only information available at the last patient visit. We compare the planned date, a burn-down extrapolation, a ridge regression, gradient-boosted trees on current counts and gradient-boosted trees with trajectory features (changes, burn rates and projected clearance times), with leave-one-organisation-out evaluation over five seeds. Across all snapshots the trajectory model had a mean absolute error of 3.8 weeks against 9.3 weeks for the plan and 5.9 for burn-down extrapolation; eight weeks before lock its error was 1.8 weeks against 4.5 and 3.9, and four weeks before lock 1.3 against 2.9 and 1.8. Two weeks before lock it was no better than the plan (2.5 against 2.0 weeks). The probability that lock is within four weeks had an area under the curve of 0.93 and an expected calibration error of 0.04; a gate that declared readiness at probability 0.8 was reached before lock in only 29% of studies and was wrong by more than two weeks in 10.6% of those. Four weeks after the last visit, a burn-down extrapolation and the learned model flagged studies that would slip with similar skill (area under the curve 0.90 and 0.89), and the plan itself has none. Prediction intervals were too narrow (70% coverage at a nominal 80% eight weeks before lock, and none two weeks before). Open queries dominated the explanations. The data are synthetic, and the study shows what such forecasts can and cannot do, not how a real portfolio would perform.*

Introduction

A clinical trial ends twice. Its last participant completes the last visit on a date that is known months in advance, and its data are declared final on a later date, the database lock, after which the data are frozen, the randomisation code is broken for the analysis and the statisticians begin.

[1]502 Pin Oak Ct, Springcity, PA-19475.

Keywords: Database lock, Clinical data management, Time-to-event forecasting, Gradient boosting, Quantile regression, Calibration, Study conduct.

Between the two dates lies a stretch of work that sponsors regard as the most time-critical of the trial: the remaining data queries are resolved, the missing forms are chased, medical terms are coded, serious adverse events are reconciled between the clinical and the safety databases, external data from laboratories and devices are reconciled, investigators sign their case report forms, protocol deviations are classified and the medical review is finished. Only when all of these are done, and a final quality review has passed, can the lock be authorised. Every week of delay postpones the analysis, the regulatory submission and, ultimately, patient access to a treatment, and delay has a cost that is large relative to the cost of the data management work itself [8].

Lock dates are nevertheless planned early and often missed. The planned interval is typically set from standard timelines and the size of the study, and the actual interval depends on the backlog that accumulated during the trial and on the capacity of the team to clear it, neither of which is known when the plan is made. During the lock period, data managers report status in weekly counts, and the forecast of the lock date is an informed judgement. Several well-known failures follow: optimism about queries that will not be answered by the sites, surprise at late vendor data transfers, and last-minute discoveries during the final review. Data management guidance describes the activities and the lock process but does not offer a quantitative forecast [3].

Prediction is attractive because the information needed is already recorded. The counts of open items in each category, and their changes from week to week, are exactly the quantities that experienced managers look at, and past studies provide examples of how those quantities relate to the time that remained. A learned model can combine them systematically, give a probability that the study will be ready within a given number of weeks, and warn early that a study is on course to miss its plan. But a forecasting model for database lock faces three difficulties that are specific to this setting. Data are scarce: a sponsor completes a handful of studies a year, so there are few examples of the end state, although each study contributes many weekly snapshots that are strongly correlated. Studies differ in ways that are not observable in the counts, notably in the productivity of the team and the organisation that runs them, so that a model must generalise to a team that it has not seen. And the decision based on the forecast is asymmetric: declaring a study ready too early commits resources and may lead to a failed lock, while declaring it late wastes time.

This paper makes four contributions. (i) It formulates database lock readiness as a forecasting problem on weekly snapshots, with the remaining time, the probability of readiness within a horizon and the early warning of slippage as targets. (ii) It describes a feature set based on burn-down trajectories that can be computed from the status counts that data management teams already report. (iii) It provides a reproducible simulation of a multi-organisation portfolio, with a plan that is independent of the outcome, and evaluates models by leave-one-organisation-out validation, together with decision-oriented measures: errors at fixed distances from lock, calibration, interval coverage, the reliability of a readiness gate and the early detection of slippage. (iv) It reports where learning does not help: a simple burn-down extrapolation

matches the learned model for early warning of slippage and for the final weeks before lock, the learned intervals are too narrow, and a readiness gate at a high probability is reached in a minority of studies.

The rest of the paper is organised as follows. Section II reviews the background on data management and forecasting, Section III the related work, Section IV the methods, Section V the experimental setup, Section VI the results, Section VII a discussion with limitations, and Section VIII concludes.

LITERATURE REVIEW

A. The Database Lock Process

Good Clinical Practice requires that trial data be accurate, complete and verifiable, and that systems that hold them be validated and controlled [1], [2]. Data management practice describes database lock as the culmination of a sequence of cleaning activities and a set of pre-lock checks, which include that all expected data have been received, discrepancies have been resolved or accepted, coding has been completed and reviewed, safety data have been reconciled, and the required approvals have been recorded [3]. After lock, changes require a documented unlock procedure, which is costly and is avoided. The practice is therefore conservative: teams commonly add a final period of quality review before they authorise the lock, and the duration of this period is itself variable.

B. Why Timelines Slip

Delays in lock arise from a few recurring mechanisms. Query resolution depends on the responsiveness of sites, which do not work to the sponsor's timeline. Late or incomplete data transfers from vendors add reconciliation work after the point at which the backlog was expected to fall. Safety reconciliation depends on two independent systems and a small number of late events can extend it. Staffing changes in the data management team reduce throughput abruptly. And the work is sequential in part: items that are cleared may be re-opened by later findings. The result is that the backlog does not fall smoothly, and the time to clear it is a heavy-tailed quantity. A forecast that extrapolates the current rate of clearance assumes a smoothness that the process does not have.

C. Forecasting Remaining Time

Forecasting the remaining time to an event from the current state of a process is a standard problem in several fields, including survival analysis and predictive process monitoring. Survival models relate covariates to the distribution of the time to an event and accommodate censoring [10]; here all studies are eventually locked, so that censoring arises only in monitoring and a direct regression of the remaining time is natural. Gradient-boosted decision trees [5] are widely used for tabular regression and classification, and they combine heterogeneous features and interactions without much preparation. To express uncertainty, quantile regression [6] estimates conditional quantiles by minimising an asymmetric loss and yields prediction intervals without a distributional assumption. For decisions that depend on the probability of an event, probabilistic forecasts should be evaluated by proper scoring rules [7] and by calibration [8].

D. Evaluation Across Organisations

Studies are run by different teams and organisations with different working practices, and a model that is evaluated with random splits of snapshots would be evaluated on studies from the same organisations that it was trained on, and would overstate its performance on a new one. A more demanding protocol holds out each organisation in turn, which tests transfer to an unseen team. Because snapshots of one study are strongly correlated, any split has to be by study at least [9]. Models that are used for a long time are also subject to drift as processes change [12], [16].

RELATED WORK

Plan-based forecasting. The usual forecast is the planned lock date, updated by the study team's judgement. It encodes experience about the size and complexity of a study but ignores the state of the backlog. We include it as the baseline, using a plan constructed only from information available at the last patient visit.

Burn-down extrapolation. Project management uses the rate at which work is being completed to extrapolate the date at which it will be done. It uses the current state, and its weakness is its smooth-rate assumption. We include it as a second baseline and find that it is competitive.

Statistical and machine-learning forecasting. Regression of remaining time on state features is the direct approach, with linear models as a standard baseline and boosted trees as a flexible alternative [5]. We use both, and add quantile models [6] for intervals.

Readiness decisions. A decision to start the lock process is a decision under uncertainty, and a calibrated probability of readiness within a horizon [7], [8] supports it more directly than a point forecast. We study a gate on such a probability and report how often it is reached and how often it is wrong.

Gap. We know of no open, reproducible evaluation of database lock forecasting that uses organisation-held-out validation, a plan that is independent of the outcome, calibration, interval coverage and early warning of slippage. This paper provides one in simulation, with the limitations of a simulator stated.

METHODOLOGY

A. Problem Formulation

Let time be measured in weeks from the last patient last visit, which is week $t = 0$. At the start of week t, a study s is described by a snapshot of the backlog in eight categories, $\mathbf{c}_{s,t} \in \mathbb{N}^8$: open queries, missing pages, uncoded terms, safety discrepancies, outstanding external data reconciliations, unsigned case report forms, unclassified protocol deviations and unfinished medical review items. The study is ready to enter final review at the first week τ_s at which every category is at or below a small tolerance $\mathbf{a}$,

$$\tau_s = \min\{t:\ \mathbf{c}_{s,t} \le \mathbf{a}\} \qquad (1)$$

and the lock occurs after a sign-off period σ_s that covers quality review and approvals, $L_s = \tau_s + \sigma_s$. The remaining time at week t is $R_{s,t} = L_s - t$. A forecaster observes the history of snapshots up to week t, together with study attributes and a planned lock week P_s fixed at $t = 0$, and produces (i)

a point forecast $\widehat{R}_{s,t}$, (ii) an interval $\left[\widehat{R}^{0.1}_{s,t}, \widehat{R}^{0.9}_{s,t}\right]$ and (iii) the probability that the lock is within h weeks,

$$\widehat{\pi}_{s,t}(h) = \Pr\left[R_{s,t} \le h | \mathbf{c}_{s,1:t},\ \mathbf{z}_s\right] \qquad (2)$$

where $\mathbf{z}_s$ collects the study attributes. Readiness is declared at the first week at which $\widehat{\pi}_{s,t}(4) \ge 0.8$.

B. Architecture

The pipeline in Figure 1 runs weekly. Status counts are extracted from the clinical data management systems, a snapshot builder computes the features of Section IV.D, the predictors produce the forecast, the interval and the readiness probability, and a readiness gate combines the probability with the team's checklist before the study team decides. The drivers of each forecast are displayed with it. Models are retrained on closed studies when new ones become available.

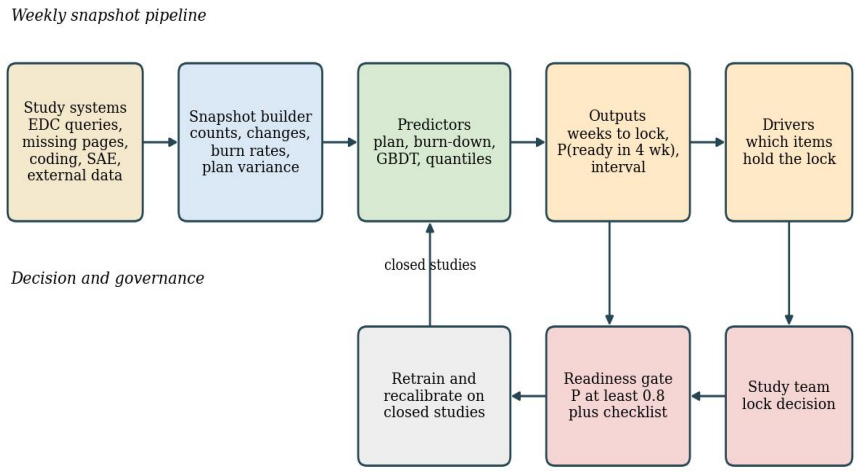


Figure 1: Architecture of the lock-readiness forecaster. Weekly status counts become a snapshot of counts, changes, burn rates and plan variance; predictors produce the weeks to lock, the probability of readiness within four weeks and an interval; a readiness gate and the study team make the decision, and closed studies are used to retrain.

C. Process Model Behind the Simulation

The simulator represents the work as eight parallel work streams. For category c of a study with N participants, the clearance in week t is a Poisson draw whose mean is the product of a nominal rate, a team capacity and a staffing multiplier,

$$D_{c,t} \sim \text{Poisson}\left(\mu_{c,t}\right), \qquad \mu_{c,t} = r_c \cdot \frac{N^{0.6}}{6} \cdot e_{o(s)}\, \eta_s \cdot m_t \qquad (3)$$

where r_c is the nominal rate of the category (Table 1), $N^{0.6}/6$ is a capacity that grows less than proportionally with study size, e_o is the productivity of the organisation o running the study (log-normal, with a standard deviation of 0.25 on the log scale), η_s is a study-level factor, and m_t is the staffing multiplier, which falls to 0.7 for three weeks with a probability of 0.05 each week. The backlog evolves as

$$C_{c,t+1} = C_{c,t} - \min\left(C_{c,t}, D_{c,t}\right) + A_{c,t} \qquad (4)$$

where $A_{c,t}$ are inflows: re-opened queries in proportion (0.12) to the queries cleared; with a probability of 0.12 per week a vendor data transfer, which adds external-reconciliation items equal to 10% of the initial number and queries equal to 5% of the initial number; and with a probability of 0.10 per week a late safety finding that adds one to three discrepancies. The sign-off period is the organisation's base sign-off time (1 to 3 weeks) plus a Poisson number of weeks (mean one), plus with probability 0.15 a further one or two weeks for re-opened items. The planned lock week is fixed at the last patient visit from the nominal clearance times at the nominal capacity, which ignores the productivity of the organisation, the staffing dips, the inflows and the sign-off variability,

$$P_s = \max_c \frac{C_{c,0}}{r_c N^{0.6}/6} + 2 + \varepsilon_s, \qquad \varepsilon_s \sim \mathcal{N}\left(0, 1.5^2\right) \qquad (5)$$

so that the plan contains information about size and initial backlog but none about the future.

TABLE 1. Backlog categories, nominal clearance rates and lock tolerance

Category	Nominal rate r_c (items per week per unit capacity)	Tolerance at readiness
Open queries	6.0	5
Missing pages	2.5	0
Uncoded terms	4.0	0
Safety discrepancies	0.5	0
External data reconciliation	3.0	3
Unsigned case report forms	5.0	0
Unclassified protocol deviations	1.5	0
Medical review items	1.2	0

D. Features

For each category the snapshot includes the logarithm of the open count, the share of the initial count that remains, the change over the last one and over the last four weeks as a share of the initial count, the burn rate, which is the mean number cleared in the last three weeks, and the projected clearance time,

$$\hat{\tau}_{c,t} = \min\left(\frac{\max\left(C_{c,t} - a_c,\ 0\right)}{\max\left(b_{c,t},\ 0.5\right)},\ 40\right), \qquad b_{c,t} = \frac{1}{3}\sum_{k=1}^{3} D_{c,t-k} \qquad (6)$$

where $b_{c,t}$ is the burn rate, the floor avoids division by zero, and the cap of 40 weeks avoids unbounded values at the start. The snapshot also includes the largest projected time over categories, the share of categories at tolerance, the week, the log study size, the therapeutic area and phase, the open items per participant, and the planned remaining time $P_s - t$. The organisation is deliberately excluded from the features so that the model must generalise across organisations.

E. Predictors

Planned date. $\widehat{R} = \max(P_s - t, 0)$. *Burn-down extrapolation.* The remaining time is the slowest category's projected clearance plus a fixed two weeks for sign-off,

$$\widehat{R}_{s,t}^{\mathrm{bd}} = \left\lceil \max_c \hat{\tau}_{c,t} \right\rceil + 2 \qquad (7)$$

Ridge regression. A linear model on the standardised counts and trajectories without the projected times, with an ℓ_2 penalty. *Gradient-boosted trees on current counts* use the counts, shares, week and study attributes but not the changes, burn rates or projected times. The *proposed model* uses all features with a gradient-boosted regression [5],

$$F(\mathbf{x}) = F_0 + \nu \sum_{m=1}^{M} h_m(\mathbf{x}), \qquad M = 250,\ \nu = 0.06,\ \text{depth } 5 \qquad (8)$$

implemented with scikit-learn [13]. Intervals come from two further boosted models that minimise the pinball loss of quantile regression [6] at levels $q = 0.1$ and $q = 0.9$,

$$\rho_q(u) = u\,(q - \mathbb{1}[u < 0]), \qquad u = R - \widehat{R}^{q} \qquad (9)$$

and the readiness probability (2) is the output of a boosted classifier trained on the label $\mathbb{1}[R \le 4]$.

F. Evaluation

Models are evaluated by leave-one-organisation-out validation: for each organisation, models are trained on the studies of the other four and tested on the studies of the held-out one. A second protocol assigns studies at random to five folds, with all snapshots of a study in the same fold, to show how much the first is more demanding. Point forecasts are evaluated at fixed distances from the true lock, which is the question a manager asks (“how good is the forecast eight, four and two weeks before lock?”), by the mean absolute error,

$$\mathrm{MAE}_k = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} |\,\widehat{R}_{s,L_s-k} - k|, \qquad k \in \{8,4,2\} \qquad (10)$$

and over all snapshots. Intervals are evaluated by their coverage and mean width,

$$\mathrm{Cov}_k = \frac{1}{|\mathcal{S}|} \sum_{s} \mathbb{1}\left[\widehat{R}^{0.1}_{s,L_s-k} \le k \le \widehat{R}^{0.9}_{s,L_s-k}\right] \qquad (11)$$

with the nominal coverage 80%. The readiness probability is evaluated by the area under the receiver operating characteristic curve, the average precision, the Brier score and the expected calibration error,

$$\mathrm{Brier} = \frac{1}{n} \sum_{i} (\,\widehat{\pi}_i - y_i)^2, \qquad \mathrm{ECE} = \sum_{b=1}^{10} \frac{n_b}{n} |\bar{y}_b - \bar{\pi}_b| \qquad (12)$$

where $y_i = \mathbb{1}[R_i \le 4]$ and b indexes ten bins of predicted probability [8]. The readiness gate is evaluated at the first week at which $\widehat{\pi}(4) \ge 0.8$ in each study: the share of studies in which the gate opens before lock, the *false-ready rate* among those, $\Pr[R > 6 | \text{gate open}]$, and the *lead error*, which is the difference between the true remaining time at the gate and four weeks. The early warning of slippage is evaluated at week four: the score of a model is its implied lock week minus the planned week,

$$\mathrm{score}_s = 4 + \widehat{R}_{s,4} - P_s \qquad (13)$$

and the area under the curve measures how well it separates studies that finish more than two weeks after plan, which is 65% of studies in the simulation. The planned date has a score of zero for every study by

construction. Finally, we report permutation importance, the increase in the error when a feature is shuffled in held-out data [14].

EXPERIMENTAL SETUP

A. Why a Simulation

Real database lock data, with weekly counts for many studies, are confidential and held by sponsors and contract research organisations, and the cause of a delay is never known with certainty. The simulator provides full ground truth, allows organisations to differ in a controlled way, and gives the same data to every predictor. The parameters are assumptions, chosen so that the portfolio resembles the kind of variation that data managers describe, and the absolute numbers belong to the simulator.

B. Portfolio

For each of five seeds a portfolio of 150 studies is generated, 30 for each of five organisations. A study has between 100 and 3,000 participants (log-normal with a median of about 400), one of four therapeutic areas and a phase of two or three, which increases the number of safety discrepancies. The initial backlog is drawn in proportion to the number of participants, for example open queries equal to the participants times a log-normal number of queries per participant (median 3.3) times a beta-distributed open share (mean 0.25). The mean time from last patient last visit to lock was 18.2 weeks (± 2.6 across seeds), and a portfolio contained about 2,300 weekly snapshots. 65% of studies locked more than two weeks after plan.

C. Procedure

For each protocol, the models are trained on the training folds and used to predict every snapshot of the held-out studies. Intervals and probabilities come from separate models trained on the same data. All results are means with 95% Student-t intervals over the five seeds. No hyperparameters were tuned beyond a first setting.

RESULTS

A. Remaining-Time Forecasts

Table 2 gives the mean absolute error at fixed distances from the true lock and over all snapshots (leave-one-organisation-out). Eight weeks before lock, the proposed model's error was 1.79 weeks (± 0.33), against 4.48 for the plan, 3.85 for burn-down extrapolation and 3.42 for the ridge regression. Four weeks before lock the errors were 1.30 for the proposed model, 1.34 for trees on counts, 1.75 for burn-down and 2.86 for the plan. Over all snapshots, the proposed model had an error of 3.80 weeks (± 1.27), against 4.72 for trees on counts, 4.90 for ridge regression, 5.85 for burn-down and 9.30 for the plan, whose large mean error comes from the early weeks, when the remaining time is longest. The trajectory features (changes, burn rates and projected times) lowered the error against trees on counts, from 2.13 to 1.79 weeks at eight weeks and from 4.72 to 3.80 overall, although the intervals overlap. Two weeks before lock, however, nothing beat the plan: the error was 1.99 weeks for the plan, 1.87 for burn-down and 2.51 for the proposed model. In the last two weeks the remaining time is

determined by the sign-off, whose duration the status counts do not reveal, and the forecasts are as accurate as a constant would be.

TABLE 2. Mean absolute error of the remaining time in weeks (leave-one-organisation-out; mean ± 95% CI over 5 seeds)

	8 weeks before lock	4 weeks before lock	2 weeks before lock	All snapshots
Planned date	4.48 ± 0.62	2.86 ± 0.23	1.99 ± 0.45	9.30 ± 3.56
Burn-down extrapolation	3.85 ± 0.88	1.75 ± 0.40	1.87 ± 0.12	5.85 ± 0.84
Ridge regression	3.42 ± 1.13	2.72 ± 0.78	2.92 ± 1.01	4.90 ± 1.77
GBDT, counts only	2.13 ± 0.35	1.34 ± 0.26	2.40 ± 0.21	4.72 ± 1.70
GBDT + trajectories (proposed)	**1.79 ± 0.33**	**1.30 ± 0.17**	2.51 ± 0.13	**3.80 ± 1.27**

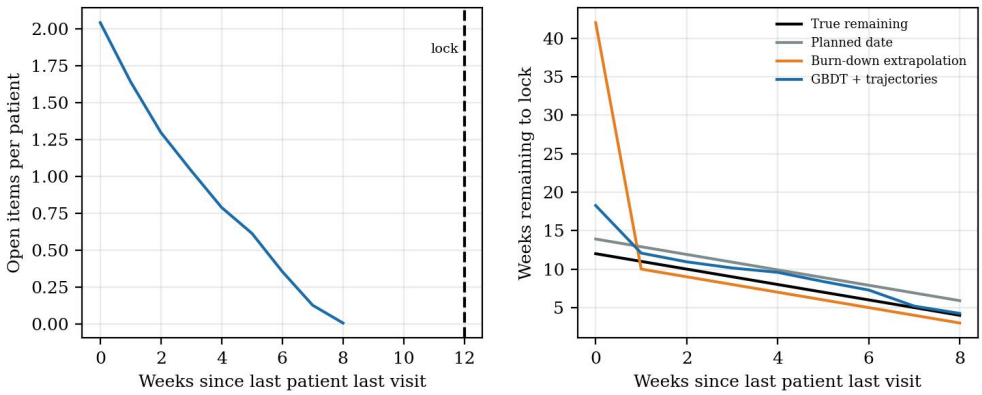


Figure 2: One study. Left: open items per participant after the last patient visit. Right: the true remaining weeks and the forecasts of the plan, burn-down extrapolation and the proposed model. The plan ignores the state of the backlog; the model's forecast tracks the true remaining time with an error that stays within a few weeks.

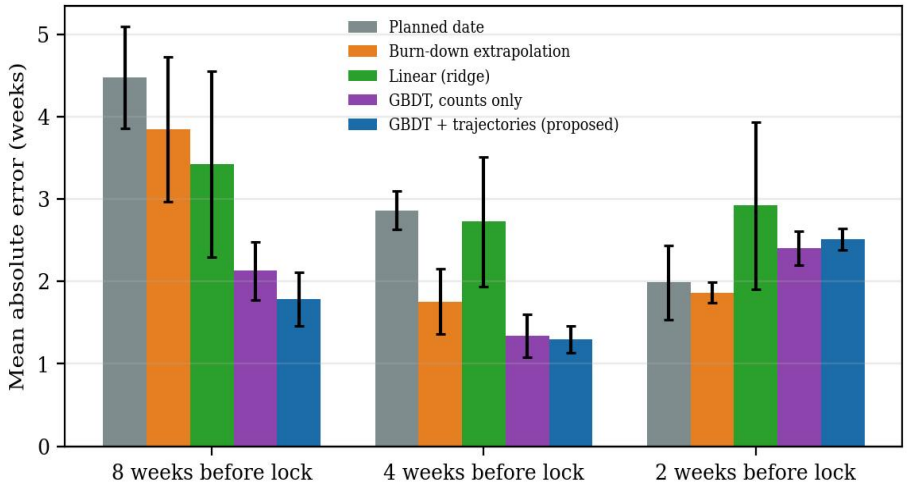


Figure 3: Mean absolute error at eight, four and two weeks before lock for the five forecasters (leave-one-organisation-out, mean over five seeds with 95% intervals). The learned models are best early; close to lock the plan and burn-down extrapolation are as good.

B. Held-Out Organisations

Leave-one-organisation-out evaluation was more demanding than random study-level folds, but the difference was small. The overall error of the proposed model was 3.80 weeks when an organisation was held out and 3.53 with random folds, and the area under the curve for readiness within four weeks was 0.930 and 0.941; the intervals overlap in both cases. By organisation, the proposed model's error ranged from 2.88 to 4.76 weeks, against 6.25 to 11.21 for the plan (Figure 4). The organisation with the highest error was the same for both forecasters. We did not examine why, but a plausible reason is that a team whose productivity is far from the training organisations' is harder to forecast until its burn rate has been observed for some weeks. This is the kind of generalisation failure that an organisation-held-out protocol is designed to expose, and it remains moderate here because the model sees the burn rate, which reveals the team's productivity within a few weeks.

C. Readiness Probability and Gate

Table 3 summarises the probability that the lock is within four weeks. The area under the curve was 0.930 (± 0.026) and the average precision 0.471 (± 0.074), which is modest because only a quarter of snapshots are within four weeks of lock. The Brier score was 0.061 and the expected calibration error 0.039, so the probabilities are reasonably calibrated. The gate, however, performed poorly in practice. A probability of at least 0.8 was reached before lock in only 28.8% of studies (± 10.4); in the others the model never became that confident, usually because the sign-off period dominates the last weeks and the model cannot be sure about it. When the gate opened, the true remaining time exceeded six weeks in 10.6% of cases (± 3.3), which is a false-ready rate that a study team may find too high, and the mean lead error was +0.69 weeks (± 0.26), which means that on average the gate opened 0.7 weeks too early relative to four weeks before lock, with a mean absolute error of 1.34 weeks.

TABLE 3. Readiness within four weeks and the gate (leave-one-organisation-out; mean ± 95% CI over 5 seeds)

Measure	Value
Area under the ROC curve	0.930 ± 0.026
Average precision	0.471 ± 0.074
Brier score	0.061 ± 0.014
Expected calibration error	0.039 ± 0.014
Studies in which the gate (probability at least 0.8) opens before lock	28.8% ± 10.4%
Gate opened when more than 6 weeks remained	10.6% ± 3.3%
Mean lead error of the gate (weeks)	+0.69 ± 0.26
Mean absolute lead error of the gate (weeks)	1.34 ± 0.25

D. Prediction Intervals

The 10–90 intervals were too narrow. Eight weeks before lock they covered the true remaining time in 70% of studies (± 5) at a nominal 80%, four weeks before lock 63% (± 8), and two weeks before lock in none of the studies, with a mean width of 4.8, 3.9 and 3.9 weeks. Two weeks before lock the forecasts of all models were centred two to three weeks away from the true value, and the quantile models, which were trained on all snapshots, could not capture the sign-off distribution at that point. A deployed system would need conformal or recalibrated intervals evaluated by lead time, and

the intervals presented here should not be relied on.

E. Early Warning of Slippage

Table 4 shows the early warning of slippage four weeks after the last visit. Because the plan has no information about the future, its score was constant and its area under the curve was 0.50. The burn-down extrapolation reached 0.904 (± 0.024), the proposed model 0.889 (± 0.065), trees on counts 0.801 (± 0.110) and the ridge regression 0.832 (± 0.094). The differences among the best three are not resolved. A simple extrapolation of the current burn rate therefore does about as well as the learned model at this task, which is evidence that most of the signal for early warning is in the burn rate of the largest backlog and not in complex interactions. The learned model's advantage is in the point forecasts early in the lock period and in the calibrated probabilities.

TABLE 4. Early warning of slippage at week 4: area under the ROC curve for studies that lock more than two weeks after plan (leave-one-organisation-out)

Forecaster	AUC (mean ± 95% CI)
Planned date	0.500 ± 0.000
Burn-down extrapolation	0.904 ± 0.024
Ridge regression	0.832 ± 0.094
GBDT, counts only	0.801 ± 0.110
GBDT + trajectories (proposed)	0.889 ± 0.065

F. What Drives the Forecast

The permutation importance (Figure 4, left) shows that the projected clearance time of the open queries was by far the most important feature, raising the error by 3.2 weeks when shuffled, followed by the number of open queries (0.85), the largest projected time over categories (0.53) and the burn rate of queries (0.49). Other categories mattered much less. The planned remaining time, which was dominant in an earlier version of the simulation in which the plan was derived from the outcome, had little importance once the plan was made independent of the outcome, which is a reminder that a plan that contains hindsight inflates the apparent value of the model. For a user, the practical reading is that the forecast is mostly a statement about how long the query backlog will take to clear at its current rate, together with a correction learned from the history of studies, and this makes it explicable.

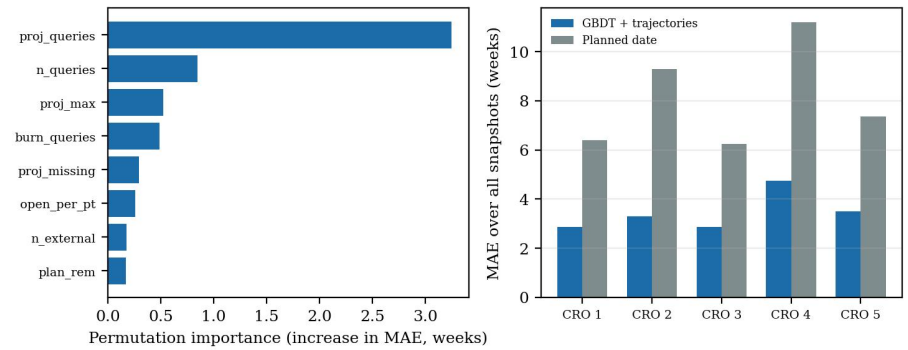


Figure 4: Left: permutation importance of the leading features for the proposed model. Right: mean absolute error of the proposed model and the plan, by organisation, over all snapshots.

G. Summary of Findings

Five findings summarise the results. State-based forecasts were far better than the plan early in the lock period and over all snapshots, and the trajectory features gave a further, unresolved improvement over counts alone. In the last two weeks before lock no learned forecast beat the plan or the burn-down extrapolation. The probability of readiness within four weeks was reasonably calibrated, but a gate at 0.8 opened in a minority of studies and was wrong by more than two weeks in about one in ten of them. The prediction intervals were too narrow at every distance and failed completely two weeks before lock. And a simple extrapolation of the burn rate of the largest backlog matched the learned model for the early warning of slippage. Each finding has a practical consequence for how a forecast of this kind should be used and presented, which the next section discusses.

DISCUSSION

The simulation supports four conclusions, each with qualifications. First, **state-based forecasts beat plan-based forecasts, mainly early in the lock period**. Eight weeks before lock the learned model's error was less than half that of the plan. The qualification is that the plan in the simulation is a simple function of size and initial backlog, and real plans embody more experience, so the margin of improvement in a real portfolio may be smaller.

Second, **in the last weeks the status counts carry little information**. Two weeks before lock, the plan and the extrapolation were as good as the learned model, since the remaining time is governed by the sign-off, which the counts do not describe. This argues for adding information that does describe it, for example the status of the final quality review, the availability of approvers and the number of changes since the last review, and for not claiming precision in the last two weeks.

Third, **a calibrated probability is not the same as a usable gate**. The probabilities were reasonably calibrated, but a threshold of 0.8 was reached in less than a third of studies, and was wrong by more than two weeks in about a tenth of those. A gate has to be designed as a decision, with the costs of premature and late declarations stated, and the threshold chosen to fit them. We did not optimise it.

Fourth, **simple extrapolation is a strong baseline for early warning of slippage**, and the learned model should be justified against it. A study team that wants to know whether it is behind plan will get most of the benefit from tracking the burn rate of the largest backlog. The learned model's additional value, in this simulation, is the better forecast early in the period and the calibrated probabilities.

Deployment considerations. The inputs of the forecaster, counts of open items by category, are available from the clinical data management and safety systems that a study already uses, and the weekly cadence matches the status meetings at which lock is discussed, so the system can be added without changing the data capture. Three practices follow from the results. The forecast should be shown with its drivers and with its track record, namely the error of past forecasts at the same distance from lock, so that the team can judge how far to trust it; the results of Table 2 show that this track record differs greatly between eight weeks and two weeks before lock.

The model should be retrained as studies close, and its calibration should be monitored on closed studies by organisation, because the evidence in Section VI.B is that transfer to a new team is the weakest point. And the forecast should inform a human decision and not replace it: a lock authorisation is a regulated act with accountability that rests on people, and Good Clinical Practice and electronic-records requirements already emphasise validated systems, traceability and accountable human decisions [1], [2]. A forecaster that makes the team feel that a study is on time when it is not could cause harm by relaxing attention, so the early warning of slippage deserves as much attention in a deployment as the point forecast.

The limitations are material. The data are simulated, with eight independent work streams, simple inflows and a sign-off period that is independent of the backlog, whereas real lock processes have dependencies among categories, priorities that change, and decisions that are made by people who respond to the forecast. The five organisations differ in one dimension, productivity, and real organisations differ in many. Study-level correlation means that the effective sample is the number of studies, 150 per portfolio, and the intervals across seeds are wide for several comparisons, in particular the overall error and the early warning. The plan is a stylised function, and the high rate of slippage (65%) was a consequence of that choice. We evaluated forecasts at fixed distances from the true lock, which uses knowledge of the lock date that a forecaster does not have; this is a standard way of evaluating forecasts of events but it should be complemented by a prospective evaluation. The intervals were poorly calibrated and the gate was not optimised. The effect of forecasts on behaviour, for example a team that speeds up when a slippage is predicted, was not modelled and would change the data on which models are retrained. Privacy, governance and validation of a model used in a regulated process are outside the scope.

For practitioners the evidence suggests four steps. Track the burn rate and projected clearance time of every backlog category weekly, because these simple quantities carry most of the early-warning signal. Use a learned forecast for the first weeks of the period, together with calibration checks, and do not trust it in the last two weeks. Treat any readiness gate as a decision with explicit costs, and evaluate how often it opens and how often it is wrong. And evaluate a model on organisations that it has not seen before deploying it.

CONCLUSION

This paper studied the prediction of clinical trial database lock readiness from weekly status snapshots, using a simulated portfolio of 150 studies run by five organisations and an evaluation that holds out each organisation in turn. Gradient-boosted models with burn-down trajectory features forecast the remaining time with an error of 1.8 weeks eight weeks before lock and 1.3 weeks four weeks before lock, against 4.5 and 2.9 weeks for the planned date, and gave a readiness probability with an area under the curve of 0.93 and an expected calibration error of 0.04. The results also set limits: two weeks before lock the learned model was no better than the plan, a readiness gate at probability 0.8 opened in less than a third of studies and was wrong by more than two weeks in about a tenth of them, the prediction intervals were too narrow, and a simple burn-down extrapolation matched the

learned model for early warning of slippage.

The practical message is that the status counts that data management teams already collect are informative about the time to lock, that most of the informative signal is the burn rate of the query backlog, and that a learned model adds most value early in the lock period and in calibrated probabilities, not in the final weeks. Future work should evaluate on real, de-identified portfolios with prospective forecasts, add information about the sign-off stage, recalibrate the intervals by lead time, design and optimise the readiness gate with explicit costs, and study how teams respond to the forecasts.

References

[1] Pawan Kumar Singh, Scaling Gate-All-Around (GAA) Transistor Architectures for Sub-2nm AI Accelerators Using Atomically Thin 2D Materials, Vol. 33 No. 08 (2024): JOCAAA, Journal Name: Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5713

[2] Pawan Kumar Singh, Energy-Efficient Task Scheduling in Green Cloud Computing Using Neuromorphic Spiking Neural Networks, Vol. 33 No. 08 (2024): JOCAAA, Journal Name: Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5712

[3] 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

[4] Ronakkumar shah , Cognitive Process Intelligence: Integrating Generative AI and Automated Workflow Discovery, Vol. 33 No. 08 (2024): JOCAAA, Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5676

[5] Ronakkumar shah, research Trends in Smart Workflow Automation: A Bibliometric Analysis of Emerging Themes, Home / Archives / Vol. 33 No. 05 (2024): JOCAAA, Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5675

[6] 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

[7] 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

[8] 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

[9] 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
[10] 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
[11] A. Srivastava, "Securing PHI in cloud-based healthcare systems: Challenges, frameworks, and future directions," J. Comput. Anal. Appl., vol. 33, no. 2, Aug. 2024. [Online]. Available: https://www.eudoxuspress.com/index.php/pub/article/view/4349/3190
[12] 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
[13] 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
[14] A. Srivastava, "A unified framework for secure cloud data management: Integrating governance, metadata intelligence, and privacy controls," J. Inf. Syst. Eng. Manage., vol. 9, no. 2, Apr. 2024. [Online]. Available: https://www.jisem-journal.com/download/45_A_Unified_Framework.pdf
[15] 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
[16] Abdul Aziz. “Intelligent Health Analytics: Mastering AI in Healthcare Data”. Acta Scientiae, vol. 25, no. 4, Aug. 2024, pp. 544-55, https://www.periodicos.ulbra.org/index.php/acta/article/view/687. https://doi.org/10.5281/zenodo.21699886
[17] Event-Aware Multi-Layer Storage Risk Forecasting for Oracle Database Estates Using HAPF, Raghu Gollapudi, International Journal of Computational and Experimental Science and Engineering, 10(4), 2024 ,10.22399/ijcesen.5183
[18] 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
[19] 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
[20] An Automated Risk Scoring Framework for SQL Execution Plan Analysis and Performance
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
[21] Manoj Kumar Kagitha, Hybrid Quantum-Classical Resource Scheduling: A Proposed Architecture for Next-Generation Hyperscale Data Centers, Publication year 2024, Volume 2024, Issue 12, https://computerfraudsecurity.com/index.php/journal/article/view/969
[22] 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
[23] 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
[24] 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
[25] 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
[26] Bandari, V. V., & Lekkala, H. B. (2024). 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
[27] Venumadhav Vavilala, Shankar Balla (2024, May). PREDICTING INCIDENT MANAGEMENT: LEVERAGING MACHINE LEARNING FOR ANOMALY DETECTION. Power System Protection and Control Scopus Q1 Journal. PSPC. https://pspac.info/index.php/dlbh/article/view/270
[28] Manikantha Varaprasad Inakollu, (2024, May), FINOPS-Driven Cloud optimization models for enterprise applications, Vol. 52 No. 2 (2024): April-June 2024, 99-110, Power System Protection and Control, ISSN-1674-3415, URL: https://pspac.info/index.php/dlbh/article/view/283,, DOI: https://doi.org/10.46121/pspc.52.2.10
[29] 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
[30] Naresh Lokiny, (2021), "Disaster Recovery and Business Continuity Planning in DevOps Cloud with AI", International Journal of Science and Research (IJSR), 10(3), 2024-2027. https://dx.doi.org/10.21275/SR24724151733, https://www.ijsr.net/getabstract.php?paperid=SR24724151733
[31] 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
[32] 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
[33] Jayanth Para, 6G Internet Technology Cyber Threat Notification & Alert System, Vol. 52 No. 4 (2024): October-December 2024, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/170 , DOI: https://doi.org/10.46121/pspc.52.4.9
[34] 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
[35] 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
[36] 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
[37] 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
[38] 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
[39] 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
[40] Vikas Suresh Bagora, SECURE EMBEDDED NETWORKING ARCHITECTURES USING TRUSTED EXECUTION ENVIRONMENTS IN BROADBAND GATEWAYS, Vol. 52 No. 2 (2024): April-June 2024, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/363
[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] Rakshitha Kiran P., Samitha Khaiyum, Aparna R. Palandye, Apoorva S. D., Arjun R., Akshaya S. A., Annapoorna S., Leveraging LLaMA3 and LangChain for Rapid AI Application Development, Vol. 20 No. 10s (2024), https://doi.org/10.52783/jes.5539, https://journal.esrgroups.org/jes/article/view/5539
[45] 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
[46] 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