

Machine Learning-Based Early Warning Systems for Predicting Cost and Schedule Overruns: A Conceptual Framework

Ulysses Chinenyeaka Ebizie 

Department of Computer Science, Federal Polytechnic Nekede, Owerri, Imo State, Nigeria 
ucebizie@fpno.edu.ng

Abstract

Cost and schedule overruns remain a persistent and costly feature of large-scale infrastructure, construction, information technology, and public-sector projects, with deviations frequently exceeding half of the originally approved budget or timeline. Conventional monitoring practices, anchored in periodic manual reporting and static earned-value formulas, typically flag deviations only after corrective action has become difficult or expensive. This paper presents a conceptual framework for machine learning-based early warning systems (ML-EWS) capable of identifying overrun risk at considerably earlier stages of project execution than conventional methods allow. Drawing upon a structured synthesis of existing literature (2015–2026) and established theoretical foundations including Risk Management Theory, Systems Theory, and the Technology Acceptance Model (TAM) the study classifies the categories of predictive features relevant to overrun forecasting, compares the performance profiles of algorithms commonly applied to the problem (such as random forests, gradient boosting machines, support vector machines, artificial neural networks, long short-term memory networks, and Bayesian networks), and sets out an integrated architecture that fuses structured project management data with unstructured textual indicators processed through natural language techniques. A five-layer early warning framework is proposed, spanning data ingestion, feature engineering, model training and validation, risk scoring, and decision-support alerting. To address critical conceptual gaps regarding algorithmic validation, empirical lead-time claims, and performance metrics, this paper articulates a rigorous mathematical formalization of lead-time gain (ΔL), dynamic evaluation metrics (AUC-PR, F β -score), and synthetic portfolio simulation frameworks tailored to imbalanced overrun datasets. The paper further examines data governance implications, an illustrative simulation of risk scoring across a project portfolio, and the practical constraints limiting adoption within developing-country project environments such as Nigeria. Findings indicate that hybrid ensemble models drawing on financial, schedule, contractual, organizational, and textual features consistently outperform single-technique models, while explainability mechanisms such as SHAP remain indispensable to sustaining stakeholder confidence in model output. The paper concludes with testable conceptual propositions, a structured future research agenda, recommendations directed at project management offices seeking to institutionalize predictive analytics, and policy-level suggestions for national infrastructure oversight bodies.

Keywords: machine learning, early warning system, cost overrun, schedule overrun, project risk prediction, earned value management, predictive analytics, conceptual framework

Introduction

Cost and schedule overruns rank among the most enduring and expensive difficulties confronting project management, cutting across construction, engineering, information technology, transportation infrastructure, and public-sector programmes in both developed and developing economies. Studies of megaprojects have repeatedly recorded average cost escalations of between 20 and 45 percent above approved budgets, with delays that frequently double the originally planned duration. Within Nigeria and comparable developing-country contexts, these overruns are further aggravated by currency volatility, inconsistent supply chains, bureaucratic procurement

delays, and weak contract enforcement, all of which make early risk detection still more critical to project viability.

Conventional project control techniques, most notably earned value management (EVM), yield retrospective indicators such as the Cost Performance Index (CPI) and Schedule Performance Index (SPI). While useful, these indices are inherently backward-looking: by the time a CPI or SPI reading signals a problem, the underlying causes have generally already gathered considerable negative momentum. EVM also presumes linear, well-structured progress reporting, a condition rarely met in practice, particularly within fragmented, multi-contractor project environments.

Machine learning (ML) offers an alternative, complementary paradigm. By learning patterns embedded in historical project data, ML models can surface subtle, nonlinear combinations of risk factors that precede overruns, often weeks or months before conventional indices begin to deteriorate. An early warning system (EWS) built on such models continuously ingests project data, computes a risk score, and triggers alerts once the probability of overrun crosses a defined threshold, thereby enabling project managers to act proactively rather than reactively.

This paper is conceptual in nature and does not present empirical validation through primary data collection or algorithmic experimentation on real-world project datasets. Rather, it undertakes a systematic synthesis of existing literature and theoretical foundations to develop a conceptual framework for ML-based early warning systems applicable to large-scale, multi-stakeholder projects. In response to fundamental methodological criticisms regarding conceptual ML research specifically regarding non-empirical lead-time claims, high-level descriptive comparisons, and static portfolio snapshots this paper establishes a formal analytical and mathematical grounding for ML-EWS operationalization.

The specific objectives are to:

1. Synthesise the current state of practice in ML-based overrun prediction;
2. Classify the data features most relevant to such prediction;
3. Propose an integrated early warning architecture suited to large-scale, multi-stakeholder projects;
4. Formalize the mathematical and evaluation constructs (lead-time gain, imbalanced metrics, dynamic risk dynamics) governing predictive early warning systems; and
5. Examine the practical, ethical, and governance considerations relevant to deployment, with particular attention to resource-constrained project environments.

The significance of the study lies in its attempt to connect two bodies of knowledge that have largely developed on separate tracks: project management theory, which has produced rich conceptual accounts of overrun causation such as optimism bias, strategic misrepresentation, scope creep, and external shocks; and computational data science, which has produced increasingly capable predictive algorithms not always grounded in the operational realities of project delivery. By explicitly mapping project management risk constructs onto machine-learning feature categories, this paper seeks a framework that is at once technically rigorous and practically implementable within existing project management offices (PMOs), including those operating with the limited digital infrastructure typical of many public-sector organisations across Sub-Saharan Africa.

The remainder of the paper proceeds as follows. Section 2 traces the evolution of overrun prediction research and establishes the theoretical grounding of the framework. Section 3 formalizes machine learning techniques and algorithmic comparative mechanics. Section 4 classifies the predictive feature space. Section 5 proposes an integrated early warning architecture. Section 6 presents the conceptual methodology. Section 7 discusses appropriate evaluation metrics

and lead-time formalization. Section 8 presents an illustrative simulation of risk scoring across a hypothetical project portfolio. Section 9 discusses implementation challenges in developing-country contexts. Section 10 examines ethical and data quality considerations. Section 11 advances conceptual propositions. Section 12 presents the limitations of the study. Section 13 outlines a future research agenda, and Section 14 closes with recommendations for practice and policy.

Literature Review and Theoretical Grounding

Evolution of Overrun Prediction Research

Research into project overrun prediction has progressed through three broad phases. The earliest phase relied on statistical regression models linking overrun magnitude to a limited set of project characteristics, such as project size, contract type, and duration. Such models offered interpretability but struggled to capture nonlinear interactions among risk factors and generalised poorly across project categories.

A second phase introduced classical machine learning classifiers, including decision trees, random forests, and support vector machines, which improved predictive accuracy by drawing on a wider range of features and by learning interaction effects automatically. Random forest ensembles, in particular, have been widely reported as effective for cost overrun classification because of their resilience to noisy and heterogeneous project data, although they remain comparatively opaque in the absence of post-hoc interpretability techniques.

The current, third phase draws on deep learning and hybrid approaches. Recurrent architectures such as Long Short-Term Memory (LSTM) networks have been applied to time-series progress data to capture temporal dependencies in schedule slippage, while convolutional and transformer-based models are beginning to be explored for extracting risk signals from unstructured site reports, meeting minutes, and correspondence. Parallel work has examined natural language processing (NLP) techniques, including sentiment analysis and topic modelling, to quantify early warning signals embedded in qualitative project documentation that would otherwise remain invisible to purely numerical models.

A theme recurring across this literature is the value of feature diversity: models combining financial, schedule, contractual, organizational, and environmental variables consistently outperform models restricted to a single data category. A further recurring theme concerns class imbalance, since overrun events, particularly severe ones, occur comparatively rarely relative to the population of successfully delivered project activities, a condition that necessitates specialized evaluation metrics and resampling strategies. Several studies additionally stress the importance of explainability, noting that project stakeholders, including sponsors, financiers, and regulators, are unlikely to act on a risk alert unless the contributing factors can be clearly articulated.

Empirical work specific to construction megaprojects has consistently found that cost escalation is not randomly distributed but is systematically associated with project type, procurement method, and the degree of scope definition at the point of budget approval, a finding that lends theoretical support to feature-based prediction rather than reliance on aggregate historical averages. Parallel research on schedule risk has emphasized the critical chain and buffer consumption rate as leading indicators of delay, both of which can be encoded numerically and supplied to supervised learning models. More recent studies have begun combining these engineering-management constructs with data-driven feature selection techniques, such as recursive feature elimination and mutual information ranking, to identify the smallest subset of variables that retains most of the predictive signal, an approach of particular value to organizations with limited data engineering capacity.

A smaller but expanding stream of literature has examined transfer learning and federated learning as mechanisms for overcoming the data scarcity that constrains many individual organizations.

Because a single public agency or contracting firm may complete only a limited number of large projects per decade, the resulting historical dataset is frequently too small to train a robust model in isolation. Federated learning allows multiple organizations, or multiple departments within a single government, to jointly train a shared model without transferring raw project records across institutional boundaries, addressing both the data-volume problem and the confidentiality concerns that commonly accompany inter-agency data sharing in public infrastructure delivery.

Notwithstanding this progress, the literature also reveals persistent gaps. Relatively few studies validate models across multiple project types or geographic regions, which raises questions about generalizability. Fewer still report lead-time analysis, namely, how far in advance of an overrun a model's warning would have been issued relative to conventional methods, despite this being the central practical justification for an early warning system rather than a purely retrospective classification exercise.

A further limitation evident in the existing body of work is the comparatively narrow treatment of contextual factors specific to developing-country project delivery. Much of the published literature draws on datasets from North American, European, or East Asian infrastructure programmes, where reporting standards, contractual norms, and digital record-keeping are considerably more mature than is typical across much of Sub-Saharan Africa. Consequently, models validated in these settings may not transfer cleanly to environments characterized by informal contracting arrangements, intermittent digital connectivity, and currency instability, underscoring the need for context-sensitive feature engineering rather than direct model transplantation.

Theoretical Foundations

The conceptual framework advanced in this paper is anchored in three established theoretical domains that collectively provide the intellectual scaffolding for understanding how machine learning can be operationalized within project management practice.

Risk Management Theory provides the foundational logic for identifying, assessing, and prioritizing risks that may compromise project objectives. Within this tradition, the risk management process encompasses risk identification, qualitative and quantitative analysis, response planning, and monitoring (PMI, 2021). The ML-EWS framework proposed herein extends this process by introducing predictive analytics as a continuous monitoring mechanism capable of surfacing risk signals earlier than conventional periodic reviews. Risk management theory thus justifies the feature categories identified in Section 4, as each category represents a distinct risk domain financial, schedule, contractual, organizational, and environmental whose interactions are hypothesized to generate overrun outcomes.

Systems Theory offers a lens through which the interdependencies among project subsystems can be understood. Projects are complex adaptive systems in which cost, schedule, and external factors interact in nonlinear ways (Bertalanffy, 1968). A perturbation in one subsystem, such as a delay in material delivery, can cascade through schedule dependencies, trigger contractual penalties, and ultimately inflate costs in ways that are difficult to predict through linear analysis. Systems theory therefore supports the use of machine learning algorithms that can capture emergent patterns across multiple feature categories simultaneously, rather than treating each risk factor in isolation. The five-layer architecture proposed in Section 5 is designed to mirror this systems perspective by integrating heterogeneous data streams into a unified risk signal.

The Technology Acceptance Model (TAM), originally developed by Davis (1989), provides insight into the conditions under which project managers and governance bodies are likely to adopt ML-based early warning systems. TAM posits that perceived usefulness and perceived ease of use are the primary determinants of technology adoption. In the context of ML-EWS, perceived

usefulness is enhanced when the system demonstrates lead-time advantage over conventional methods and when explainability mechanisms such as SHAP values render the model's reasoning transparent. Perceived ease of use, in turn, depends on the system's integration with existing project management information systems and the clarity of its alert outputs. The governance considerations discussed in Section 9 are informed by TAM, particularly with respect to organizational resistance and the framing of predictive alerts as diagnostic aids rather than instruments of blame allocation.

Together, these three theoretical pillars risk management theory, systems theory, and the technology acceptance model provide the conceptual justification for the framework developed in subsequent sections. They establish that overrun prediction is not merely a technical exercise in algorithm selection but a socio-technical endeavour requiring alignment between predictive capability, systems thinking, and human-centred design principles.

Recent Advances (2022–2026)

The period from 2022 to 2026 has witnessed significant developments that bear directly on the conceptual framework proposed in this paper. Four domains warrant particular attention:

Explainable Artificial Intelligence (XAI) in Project Management has matured considerably, moving from a niche concern to a mainstream requirement. Lundberg and Lee's (2017) SHAP framework and Ribeiro et al.'s (2016) LIME methodology have been increasingly applied to project management contexts to demystify model predictions. Recent studies have demonstrated that project stakeholders exhibit measurably higher trust in and willingness to act upon ML-generated risk alerts when accompanied by SHAP-based explanations identifying the top contributing features (Sanni-Anibire et al., 2023). This finding reinforces the centrality of Layer 4 (Risk Scoring and Explainability) in the proposed architecture.

Federated Learning for Data Privacy has gained traction as a solution to the data-sharing barriers that impede model training across organizational boundaries. Yang et al. (2019) established the conceptual foundations, and subsequent work has applied federated learning to construction project datasets, demonstrating that collaborative model training can achieve performance comparable to centralized training without exposing sensitive project records (Liu et al., 2024).

Digital Twins and Building Information Modelling (BIM) Integration with ML represents a convergence of physical and digital project representations. BIM-integrated ML models can leverage geometric, temporal, and cost data embedded in digital building models to generate real-time risk assessments (Xu et al., 2023). The conceptual framework proposed herein accommodates BIM-derived features within the structured data ingestion layer.

Reinforcement Learning for Dynamic Schedule Adjustment has emerged as a promising avenue for moving beyond static prediction to active intervention. Unlike supervised learning models that predict overrun probability, reinforcement learning agents can suggest optimal schedule adjustments in response to evolving project conditions (Zhang & Wang, 2025).

Research Gap

Despite the advances surveyed above, a significant gap remains in the integration of these disparate strands into a unified conceptual framework that is simultaneously theoretically grounded, practically implementable, and context-sensitive to developing-country project environments. Existing studies tend to fall into one of two categories: empirical studies that validate specific algorithms on particular datasets but offer limited generalisable architecture, and theoretical discussions of project risk that do not engage with the technical capabilities and constraints of machine learning.

Specifically, previous empirical and conceptual studies have not adequately addressed four interrelated lacunae:

- The absence of a theoretically anchored architecture that maps project management risk constructs onto ML feature categories in a systematic manner;
- Insufficient attention to the lead-time dimension of early warning, with most studies evaluating classification accuracy at a single point in time rather than measuring how far in advance a warning is issued relative to conventional methods;
- Limited treatment of the socio-technical barriers to adoption, including organisational resistance, skills gaps, and data governance constraints characteristic of developing-country PMOs; and
- A lack of conceptual propositions that can guide future empirical researchers in testing the framework under real-world conditions.

The conceptual framework proposed in this paper addresses these gaps by synthesising risk management theory, systems theory, and technology acceptance theory into a five-layer architecture that is explicitly designed for lead-time advantage, explainability, and contextual adaptability.

Machine Learning Techniques for Overrun Prediction

Mathematical and Algorithmic Foundations

To move beyond purely descriptive summaries, machine learning algorithms applied to overrun forecasting must be understood through their mathematical optimization mechanics, input processing, and computational trade-offs.

Random Forest (RF)

Random Forests construct an ensemble of B decision trees $\{T_1, T_2, \dots, T_B\}$ using bootstrap samples of the training set. For classification, the predicted probability of an overrun ($Y = 1$) given feature vector x is: $P(Y = 1 | x) = (1/B) \sum_{b=1}^B P_b(Y = 1 | x)$

Where P_b is the terminal node probability in tree T_b . Variance reduction is achieved by randomly selecting $m \approx \sqrt{p}$ features at each split point out of p total predictors. RF handles heterogeneous data matrices well but struggles with temporal trends.

Gradient Boosting Machines (XGBoost/LightGBM)

Gradient boosting minimizes a regularized objective function $L^{(t)}$ at iteration t :

$$L^{(t)} = \sum_i l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$$

Where l is a differentiable loss function (e.g., binary logistic loss for overrun occurrence), and $\Omega(f_t) = \gamma T + (1/2)\lambda \sum_j w_j^2$ penalizes tree complexity (T terminal nodes with weights w_j). Using a second-order Taylor expansion: $L^{(t)} \approx \sum_i [l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + (1/2) h_i f_t^2(x_i)] + \Omega(f_t)$

Where $g_i = \partial l / \partial \hat{y}^{(t-1)}$ and $h_i = \partial^2 l / \partial (\hat{y}^{(t-1)})^2$. This optimization allows gradient boosting to achieve high accuracy on tabular PMIS data, though it requires tuning to prevent overfitting on small datasets.

Long Short-Term Memory Networks (LSTM)

For temporal tracking of time-series project features (e.g., weekly SPI/CPI sequences $x_1, \dots, x_t$), LSTMs regulate information flow via gating mechanisms:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (\text{Forget Gate})$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (\text{Input Gate})$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (\text{Candidate Cell State})$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (\text{Updated Cell State})$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (\text{Output Gate})$$

$$h_t = o_t * \tanh(C_t) \quad (\text{Hidden State Output})$$

LSTMs preserve temporal dependencies across extended project lifecycles, capturing compound delays that static classifiers miss.

Bayesian Networks (BN)

Bayesian Networks represent joint probability distributions over a set of random variables $X = \{X_1, \dots, X_n\}$ via a Directed Acyclic Graph (DAG):

$$P(X_1, X_2, \dots, X_n) = \prod_i P(X_i | \text{Parents}(X_i))$$

BNs facilitate probabilistic forward forecasting and backward diagnostic reasoning under severe data missingness, making them suitable for early-stage project qualitative risk assessments.

Comparative Algorithmic Analysis

Table 1 provides a systematic comparison of these machine learning techniques, detailing their functional strengths, limitations, and operational suitability across different project stages.

Table 1. Comparative Characteristics and Operational Suitability of ML Techniques for Overrun Prediction

Algorithm	Computational Complexity	Data Requirements	Primary Application Stage	Key Strength	Critical Limitations
Random Forest	$O(B \cdot m \cdot n \log n)$	Medium ($N \approx 100-1,000$)	Planning & Execution	Robust to noisy/missing data; non-parametric; low tuning overhead.	Struggles with strong temporal trends; opaque without SHAP.
Gradient Boosting (XGBoost/LightGBM)	$O(K \cdot d \cdot n \log n)$	Medium-High ($N > 500$)	Active Execution	Superior accuracy on tabular PMIS data; handles non-linear interactions.	Prone to overfitting on small samples; sensitive to hyper-parameters.
Support Vector Machine (SVM)	$O(n^2 \cdot p + n^3)$	Low-Medium ($N < 500$)	Feasibility & Planning	Effective in high-dimensional feature spaces;	Computationally expensive for large N ; non-probabilistic
Artificial Neural Network (ANN)	$O(E \cdot n \cdot w)$	High ($N > 2,000$)	Mid-to-Late Execution	Captures highly complex non-linear combinations across multi-domain features.	Extreme "black box" nature; requires large sample sizes; high hardware overhead.
Long Short-Term Memory (LSTM)	$O(W)$ per time step	High Sequential ($T > 30$ periods)	Continuous Execution	Models sequential dynamics and temporal delay accumulation directly.	Long training durations; requires consistently sampled time-series logs.
Bayesian Networks	$O(2^k)$ exact inference	Low-Medium ($N \approx 50-500$)	Feasibility & Procurement	Transparent probabilistic reasoning under uncertainty; handles structural missingness.	NP-hard structure learning as node count grows; discretization loses continuous resolution.

Note. B = trees, m = sampled features, n = samples, p = features, K = boosting iterations, d = depth, E = epochs, w = weights, W = total parameters, k = max parents per node. Synthesised from Wauters and Vanhoucke (2016), Sanni-Anibire et al. (2020), and recent literature (2022–2026).

Stacked Generalization Ensembles

To maximize predictive performance across heterogeneous project contexts, single-model reliance is superseded by stacked generalization. A meta-learner M (e.g., regularized logistic regression) combines base predictions from RF, XGBoost, and LSTM models:

$$\hat{y}_{final} = \sigma(\alpha_0 + \alpha_1 \hat{y}_{RF} + \alpha_2 \hat{y}_{XGB} + \alpha_3 \hat{y}_{LSTM})$$

Where parameters α are fitted on out-of-fold cross-validation predictions, minimizing model variance and optimizing robustness against sparse data distributions.

Predictive Modelling Workflow

The overall predictive modelling pipeline spans data ingestion through model execution to alert generation, as illustrated below:

[Step 1: Data Collection & Integration] → Gather ERP, Schedule, Textual & Market Data

[Step 2: Data Preprocessing & NLP] → Clean data, tokenize text, handle missing values

[Step 3: Feature Engineering & Extraction] → Compute rolling metrics, sentiment scores, temporal alignment

[Step 4: Class Imbalance Handling] → Apply SMOTE or cost-sensitive reweighting

[Step 5: Model Training & Tuning] → Train Ensembles, LSTMs, Bayesian Tuning

[Step 6: Model Evaluation & Lead-Time Analysis] → Assess F1, AUC-PR, and Lead-Time Advantage

[Step 7: Explainability & Alert Generation] → Compute SHAP values, trigger tiered alerts L1–L3

Figure 1. Flowchart of the predictive modelling process within the ML-EWS framework.

Predictive Feature Categories and Data Sources

To achieve high predictive performance, feature extraction must span both structured operational logs and unstructured qualitative documentation. Table 2 details the predictive feature space, associated metrics, and extraction protocols.

Table 2. Predictive Feature Categories, Key Indicators, Mathematical Definitions, and Data Sources

Feature Category	Representative Indicators	Formulations / Extractions	Typical Data Source
Financial	CPI, Cost Variance (CV), Change Order Velocity	$CPI = EV/AC$; $CV = EV - AC$; $\Delta CO = d(\text{Approved CO})/dt$	ERP / Accounting Systems
Schedule	SPI, Critical Path Float Consumption	$SPI = EV/PV$; $\Delta \text{Float} = \text{Float}_{baseline} - \text{Float}_{current}$	Primavera P6 / MS Project Logs

Contractual	Procurement Type, Change Order Frequency, Subcontractor Density	Nominal Encoding (Lump-Sum, EPC, Design-Build); Ratio of Subcontractor Work Packages to Total	Procurement Records & Contracts
Organisational	Project Manager Experience, Team Meeting Turnover, Cadence	Manager Experience Index (Yrs); Turnover Rate = $\frac{N_{\text{departed}}}{N_{\text{average}}}$; Frequency (M_held / M_planned)	HR Systems & PMO Dashboards
External / Environmental	Material Fluctuation, Price Weather Inflation Delays, Volatility	Localized CPI/Material Price Indices; Rainfall/Temperature Anomaly Days (D_adverse)	Macroeconomic Databases & Meteorological Logs
Textual / Unstructured	Site Report Sentiment, Risk Register Escalation Signals	VADER/roBERTa Polarity Score $S \in [-1, 1]$; Risk Vector TF-IDF Density	Site Progress Reports, Minutes, Correspondence

Note. EV = Earned Value; AC = Actual Cost; PV = Planned Value; CO = Change Order. Synthesised from project management literature and ML-EWS studies (2018–2026).

Proposed Early Warning System Architecture

The proposed ML-EWS architecture spans five distinct operational layers designed to convert disparate organizational data into actionable, explainable governance decisions.

Layer 5: Decision-Support Alerting → Tiered Alerts: Informational, Cautionary, Critical | Routing | Protocols

Layer 4: Risk Scoring & Explainability → Probability Calculation | SHAP & LIME Feature Attribution | Root Cause

Layer 3: Model Training & Validation → Ensemble Classifiers | SMOTE Resampling | Project-Level Cross-Validation

Layer 2: Feature Engineering → Rolling CPI/SPI | NLP Sentiment Analysis | Change Velocity | Feature Selection

Layer 1: Data Ingestion → Structured ERP Logs | Unstructured Text Reports | Market Indicators

Figure 2. Conceptual model of the proposed five-layer ML-EWS architecture.

Architectural Layer Definitions

Layer 1: Data Ingestion. Automated ETL (Extract, Transform, Load) pipelines establish secure API connections to ERPs (SAP, Oracle), project scheduling logs, and unstructured text repositories (site logs, meeting minutes).

Layer 2: Feature Engineering. Transforms raw data into model inputs. Time-stamped sequences yield rolling performance metrics (CPI_{4w} , SPI_{4w}), velocity ratios, and NLP sentiment scores (S_{text}). Temporal leakage is strictly prevented by masking all data recorded after prediction timestamp t_{pred} .

Layer 3: Model Training and Validation. Executes stacked ensemble learning (Section 3.3) over resampled balanced feature spaces. Cross-validation is partitioned strictly at the project level (GroupKFold) to prevent spatial-temporal leakage between training and testing splits.

Layer 4: Risk Scoring and Explainability. Converts model probability outputs into a continuous Overrun Risk Index ($R_i \in [0, 100]$). For each prediction, tree SHAP values decompose R_i into additive feature contributions: $R_i = \phi_0 + \sum_j \phi_j(x_i)$

Where ϕ_0 is the baseline portfolio expectation and ϕ_j measures the risk contribution of variable j .

Layer 5: Decision-Support Alerting. Maps R_i scores to governance tiers (Informational, Cautionary, Critical) with explicit routing protocols to avoid alert fatigue and ensure organizational accountability.

Conceptual Methodology

This paper applies a formal theory-building conceptual methodology, following Doty and Glick (1994). The study structures conceptual propositions and architectural specifications through four sequential stages:

Stage 1: Systematic Literature Search (Scopus, Web of Science, IEEE Xplore, 2015–2026)

Stage 2: Conceptual Filtering & Relevance Screening (85 primary studies retained)

Stage 3: Theoretical Triangulation (Mapping Risk Theory, Systems Theory, TAM)

Stage 4: Synthesized Framework & Mathematical Formalization

Structured Literature Search: Systematic querying across Scopus, Web of Science, IEEE Xplore, and Google Scholar using combined terms ("machine learning", "cost overrun", "schedule delay", "early warning system", "earned value management") spanning 2015–2026.

Filtering and Screening: Articles were filtered to exclude single-domain studies (e.g., purely computational ML papers lacking project context, or qualitative risk studies lacking computational models), yielding a core set of 85 key publications.

Thematic and Theoretical Triangulation: Identified patterns across literature regarding algorithmic limitations, feature efficacy, and governance barriers. Findings were mapped onto Risk Management Theory, Systems Theory, and the Technology Acceptance Model.

Architectural & Mathematical Synthesis: Formalized the five-layer framework, integrated explainability protocols, and established lead-time evaluation formulas to address existing gaps in conceptual ML literature.

Model Evaluation Considerations & Lead-Time Formalization

Imbalanced Dataset Metrics

Evaluating overrun prediction models using traditional Accuracy $((TP + TN)/(TP + TN + FP + FN))$ is misleading due to severe class imbalance; a naive classifier predicting "No Overrun" in a portfolio with a 10% overrun rate achieves 90% accuracy while failing entirely as an early warning system. Robust evaluation requires metrics tailored to imbalanced data:

$$Precision = TP / (TP + FP), \quad Recall = TP / (TP + FN)$$

$$F_2 = (1 + \beta^2) \cdot (Precision \cdot Recall) / (\beta^2 \cdot Precision + Recall)$$

Setting $\beta = 2$ (F_2 -score) prioritizes Recall over Precision, minimizing missed overruns (False Negatives) while accounting for false alarm costs. Furthermore, Area Under the Precision-Recall Curve (AUC-PR) provides a robust metric for comparing classifiers under extreme class skewness.

Mathematical Formalization of Lead-Time Advantage

The operational value of an ML-EWS lies in its Lead-Time Advantage (ΔL), defined as the difference in time between when the ML system detects an impending overrun and when conventional EVM flags the issue.

Let T_{event} represent the date of actual overrun occurrence. Define t_{EVM} as the earliest reporting period where EVM indices breach threshold criteria ($CPI_t < 0.90$ or $SPI_t < 0.90$). Define t_{ML} as the earliest period where the ML-EWS score exceeds the threshold ($R_t \geq \tau$):

$$t_{EVM} = \min \{ t \mid CPI_t \leq \theta_{CPI} \vee SPI_t \leq \theta_{SPI} \}$$

$$t_{ML} = \min \{ t \mid P(Y = I \mid x_t) \geq \tau \}$$

The Lead-Time Gain (ΔL) expressed in time units (weeks or months) is formalized as:

$$\Delta L = t_{EVM} - t_{ML}$$

Where $\Delta L > 0$ demonstrates early warning capacity. The relative lead-time advantage ratio (ΔL_{rel}) over total planned project duration (D_P) is: $\Delta L_{rel} = ((t_{EVM} - t_{ML}) / D_P) \times 100\%$. Empirical validity requires establishing that ΔL provides sufficient operational lead-time for project managers to implement corrective interventions before cost or schedule variances become unrecoverable.

Illustrative Portfolio Simulation & Dynamic Risk Dynamics

Baseline Static Portfolio Snapshot

To illustrate the operational mechanics of Layer 4 and Layer 5, consider a portfolio of six active infrastructure projects monitored monthly. Table 3 details a single reporting snapshot, comparing conventional EVM indices against the ensemble ML-EWS score (R_i) and assigned alert tiers.

Table 3. Illustrative Risk Scores, EVM Metrics, and Alert Tiers across a Project Portfolio

Project ID	CPI	SPI	NLP Sentiment	ΔCO Velocity	ML Risk (R_i)	Alert Tier	Top SHAP Feature
Project Alpha	1.02	0.98	+0.42	Low	22	Informational	Positive Baseline Progress ($\phi = -12$)
Project Beta	0.97	1.00	+0.15	Low	31	Informational	Stable Float Reserve ($\phi = -8$)
Project Gamma	0.95	0.93	-0.05	Moderate	38	Informational	Minor CPI Deterioration ($\phi = +5$)
Project Delta	0.91	0.89	-0.38	High	54	Cautionary	Subcontractor Turnover Rate ($\phi = +18$)
Project Epsilon	0.88	0.81	-0.72	Critical	77	Critical	Negative Site Sentiment & ΔCO ($\phi = +29$)
Project Zeta	0.99	1.05	+0.55	Low	18	Informational	Ahead of Schedule ($\phi = -15$)

Note. Portfolio alert thresholds: Informational ($0 \leq R_i < 40$), Cautionary ($40 \leq R_i < 70$), Critical ($70 \leq R_i \leq 100$).

Dynamic Multi-Period Longitudinal Simulation

To demonstrate how early warning warnings, evolve prior to actual project failure, Table 4 simulates the dynamic trajectory of Project Epsilon over a 6-month evaluation timeline (T_1 to T_6). This multi-period view illustrates the temporal lead-time advantage (ΔL) of ML-EWS over conventional EVM indicators.

Table 4. Simulated Multi-Period Longitudinal Risk Dynamics for Project Epsilon (D_P = 24 Months)

Period (t)	CPI	SPI	Sentiment (S)	Subc. Turnover	ML Risk (R_t)	EVM Status	ML Tier	Alert	Action Window
Month 1 (T ₁)	1.00	1.00	+0.20	5%	24	On Track	Informational		Monitoring Baseline
Month 2 (T ₂)	0.98	0.96	-0.18	12%	42	On Track	Cautionary (t _{ML})		Intervention Window Open
Month 3 (T ₃)	0.95	0.92	-0.45	22%	61	On Track	Cautionary		Proactive PMO Mitigation
Month 4 (T ₄)	0.92	0.87	-0.68	31%	77	On Track	Critical		Major Contractual Realignment
Month 5 (T ₅)	0.88	0.81	-0.72	35%	85	Breach (t _{EVM})	Critical		Reactive Crisis Management
Month 6 (T ₆)	0.81	0.74	-0.85	40%	92	In Breach	Critical		Overrun Unrecoverable

Note. EVM Breach defined as CPI < 0.90 or SPI < 0.90. $\tau_{\text{Cautionary}} = 40$.

Analysis of Simulation Results

In Month 2 (T₂), while conventional EVM indicators remain within acceptable bounds (CPI = 0.98, SPI = 0.96), the ML-EWS flags a Cautionary Alert (R₂ = 42) driven by declining site sentiment (S = -0.18) and rising subcontractor turnover. EVM does not register a breach until Month 5 (T₅), when SPI falls to 0.81.

Applying the lead-time formulation from Section 7.2:

$$t_{\text{EVM}} = \text{Month 5}, \quad t_{\text{ML}} = \text{Month 2}$$

$$\Delta L = t_{\text{EVM}} - t_{\text{ML}} = 5 - 2 = 3 \text{ Months (12 Weeks)}$$

$$\Delta L_{\text{rel}} = (3 / 24) \times 100\% = 12.5\% \text{ of Total Project Schedule}$$

This 3-month window gives the PMO operational lead time to renegotiate subcontractor staffing, address supply bottlenecks, and stabilize project scope before the overrun becomes unrecoverable.

Implementation Challenges in Developing-Country Contexts

Deploying machine learning-based early warning systems in developing economies like Nigeria involves specific institutional, structural, and technical challenges:

- **Data Fragmentation and Paper-Based Records:** Many public-sector infrastructure projects lack centralized digital records. Historical reporting is often stored in physical archives, causing data loss, inconsistent variable definitions, and missing features.
- **Infrastructure Instability and Cloud Latency:** Intermittent electricity supply, limited high-speed internet in remote sites, and restricted access to cloud compute infrastructure impede real-time data ingestion and model inference.
- **Macroeconomic Volatility and Concept Drift:** High inflation, sudden currency devaluations, and fluctuating material prices cause rapid concept drift. An ML model trained on historical cost parameters may quickly degrade unless continuously recalibrated against real-time economic indicators.
- **Organisational Culture and Technology Acceptance:** In hierarchical governance structures, automated predictive alerts may be resisted if perceived as disciplinary tools rather than

decision-support aids. According to TAM, low perceived ease of use and fear of personal liability can hinder adoption unless accompanied by clear change-management protocols.

- **Skills Gap:** A shortage of professionals skilled in both project management and data science limits the internal capacity of PMOs to maintain ML architectures without external dependencies.

Ethical and Data Quality Considerations

Deploying predictive models in public infrastructure governance introduces critical ethical and operational considerations:

- **Algorithmic Bias:** Machine learning models trained on historical project logs may learn and perpetuate systemic biases. For instance, if past records reflect biased evaluations against specific local contractors or regional procurement entities, models may assign disproportionately high risk scores to future proposals from these groups. Regular fairness audits are necessary to assess predictive parity across project sub-groups.
- **Alert Fatigue and Accountability:** High false positive rates can cause project managers to ignore warnings. Conversely, over-reliance on automated outputs may create confusion regarding legal accountability when project interventions fail. Clear escalation protocols must establish that ML-EWS outputs serve as diagnostic aids, with final decision authority resting with human managers.
- **Data Privacy:** Integrating worker productivity logs, site communications, and vendor financial records requires strict compliance with data privacy regulations (e.g., NDACT / NDPR in Nigeria). Anonymization and differential privacy frameworks should be applied during Layer 1 ingestion.

Conceptual Propositions

To guide empirical validation, five testable conceptual propositions are advanced, linking theoretical constructs to observable outcomes.

Table 5. Summary of Conceptual Propositions and Underlying Theoretical Foundations

Proposition ID	Conceptual Proposition Statement	Theoretical Basis
Proposition 1	The integration of explainable AI techniques (e.g., SHAP) with structured project management data significantly reduces stakeholder resistance compared to black-box models, increasing the rate of timely interventions.	Technology Acceptance Model (TAM) & XAI Theory
Proposition 2	Hybrid ensemble models incorporating financial, schedule, contractual, organisational, and textual features demonstrate superior lead-time gain (ΔL) compared to single-domain models.	Systems Theory & Feature Diversity Principles
Proposition 3	Federated learning architectures enable resource-constrained PMOs to train accurate predictive models without consolidating sensitive proprietary data.	Collaborative Governance & Privacy-Preserving ML
Proposition 4	Project organizations implementing tiered, governance-aligned alerting protocols exhibit higher corrective action rates than those using binary alert systems.	Risk Governance & Human Decision Support Theory
Proposition 5	Incorporating NLP-derived sentiment metrics from qualitative progress reports improves predictive accuracy	Information Processing Theory & Natural Language Processing

	and lead time (ΔL) compared to relying solely on quantitative EVM metrics.	
--	--	--

Limitations of the Study

While this study provides a comprehensive theoretical and architectural framework for ML-based early warning systems, several inherent limitations must be acknowledged:

- **Conceptual Nature:** The paper is conceptual and theoretical. It does not present primary empirical validation, real-world algorithmic testing, or primary field performance data from live infrastructure projects.
- **Contextual Generalisability:** The framework heavily considers the operational constraints of Sub-Saharan Africa (specifically Nigeria). While highly relevant for developing economies, direct translation to highly digitized settings may require architectural recalibration.
- **Data Quality Reliance:** The performance of the proposed architecture fundamentally hinges on the availability, cleanliness, and accuracy of historical project records. In environments where digital data logging is fragmented, model training efficiency may be impaired.
- **Absence of Dynamic Optimization:** The framework focuses primarily on overrun risk prediction and diagnostic alerting rather than automated, real-time schedule or budget reallocation (e.g., via active reinforcement learning).

Future Research Agenda

To transition the ML-EWS framework from theoretical conceptualization to operational deployment, future research should prioritize five core empirical agendas:

1. **Empirical Architectural Validation:** Validate the five-layer architecture using multi-project datasets ($N > 200$) across diverse sectors (transport, energy, IT) to evaluate model generalizability and calibration performance.
2. **Empirical Lead-Time Benchmark Studies:** Quantify the actual lead-time gain (ΔL) of ML-EWS over conventional EVM across live construction projects, measuring the practical value of early alerts.
3. **Contextual Socio-Technical Evaluations:** Conduct longitudinal qualitative studies in developing-country PMOs to assess how organizational culture, trust, and skill levels impact the adoption of ML alerts.
4. **BIM and IoT Sensor Integration:** Research real-time feature extraction pipelines that feed automated Internet of Things (IoT) site data (e.g., equipment tracking, automated worker density logs) into Layer 1 ingestion layers.
5. **Prescriptive Analytics and Reinforcement Learning:** Extend Layer 5 beyond risk alerting by developing offline reinforcement learning models capable of recommending optimal corrective actions.

Conclusion and Recommendations

Machine learning-based early warning systems offer a proactive alternative to traditional earned-value techniques for detecting cost and schedule overrun risks in large-scale projects. By integrating structured project logs with unstructured text data, ML models can surface complex, non-linear risk patterns earlier than conventional indices. Ensembles such as gradient boosting and random forests provide a strong baseline for tabular PMIS data, while recurrent architectures (LSTM) and NLP techniques capture temporal and sentiment-based signals.

Recommendations for Practice

- **Standardize Digital Logging:** Establish standardized, machine-readable project management repositories as a prerequisite for predictive analytics.

- **Prioritize Explainability:** Integrate post-hoc XAI frameworks (e.g., SHAP) alongside complex models to build stakeholder trust and support root-cause analysis.
- **Implement Tiered Alerting:** Align predictive risk thresholds with specific governance protocols to prevent alert fatigue and define clear operational escalation paths.
- **Mitigate Concept Drift:** Continuously recalibrate models against external economic indicators, particularly in volatile markets.

Recommendations for Policy

National infrastructure oversight bodies (e.g., Bureau of Public Procurement) should establish standardized data-logging standards for publicly funded projects. Aggregating anonymized project data across agencies creates national-scale historical repositories needed to train robust predictive models, supporting a transition from reactive project monitoring to proactive risk management.

References

- Ahiaga-Dagbui, D. D., & Smith, S. D. (2014). Rethinking construction cost overruns: Cognition, learning and estimation. *Journal of Financial Management of Property and Construction*, 19(1), 38–54. <https://doi.org/10.1108/JFMPC-06-2013-0027>
- Bertalanffy, L. von. (1968). *General system theory: Foundations, development, applications*. George Braziller.
- Cheng, M.-Y., & Hoang, N.-D. (2016). Slope collapse prediction using Bayesian framework with k-nearest neighbor density estimation: Case study in Taiwan. *Journal of Computing in Civil Engineering*, 30(1), 04014116. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000468](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000468)
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3), 319–340. <https://doi.org/10.2307/249008>
- Doty, D. H., & Glick, W. H. (1994). Typologies as a unique form of theory building: Toward improved understanding and modeling. *Academy of Management Review*, 19(2), 230–251. <https://doi.org/10.5465/amr.1994.9410210748>
- Flyvbjerg, B. (2014). What you should know about megaproject planning and management. *Project Management Journal*, 45(2), 6–19. <https://doi.org/10.1002/pmj.21409>
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. <https://doi.org/10.1007/978-0-387-84858-7>
- Liu, Y., Zhang, W., & Chen, H. (2024). Federated learning for cross-organisational construction risk prediction: A privacy-preserving approach. *Automation in Construction*, 158, 105312. <https://doi.org/10.1016/j.autcon.2023.105312>
- Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 4765–4774.
- Odeyinka, H., Lowe, J., & Kaka, A. (2013). An evaluation of risk factors impacting construction cash flow forecast. *Journal of Financial Management of Property and Construction*, 18(3), 265–283. <https://doi.org/10.1108/JFMPC-06-2012-0023>
- Petrusheva, S., & Zileska Pancovska, V. (2017). Traditional and innovative methods for cost estimation of construction projects. *Balkan Journal of Applied Mathematics and Informatics*, 1, 15–20.
- PMI. (2021). *A guide to the Project Management Body of Knowledge (PMBOK® guide)* (7th ed.). Project Management Institute.
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. <https://doi.org/10.1145/2939672.2939778>

- Sanni-Anibire, M. O., Zin, R. M., & Olatunji, S. O. (2020). Machine learning model for delay risk assessment in tall building projects. *International Journal of Construction Management*, 20(5), 465–477. <https://doi.org/10.1080/15623599.2020.1768326>
- Sanni-Anibire, M. O., Hassanain, M. A., & Assaf, S. (2023). Explainable artificial intelligence in construction project management: Stakeholder trust and decision-making. *Journal of Construction Engineering and Management*, 149(4), 04023042. <https://doi.org/10.1061/JCEMD4.COENG-12821>
- Wauters, M., & Vanhoucke, M. (2016). A comparative study of Artificial Intelligence methods for project duration forecasting. *Expert Systems with Applications*, 46, 249–261. <https://doi.org/10.1016/j.eswa.2015.10.007>
- Xu, N., Ma, L., Wang, L., Deng, Y., & Ni, G. (2021). Extracting domain knowledge elements of construction safety management: Rule-based approach using Chinese natural language processing. *Journal of Management in Engineering*, 37(2), 04021001. [https://doi.org/10.1061/\(ASCE\)ME.1943-5479.0000881](https://doi.org/10.1061/(ASCE)ME.1943-5479.0000881)
- Xu, Y., Li, H., & Wang, Z. (2023). BIM-integrated machine learning for real-time construction risk assessment. *Advanced Engineering Informatics*, 55, 101923. <https://doi.org/10.1016/j.aei.2022.101923>
- Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2), 1–19. <https://doi.org/10.1145/3298981>
- Zhang, L., & Wang, H. (2025). Reinforcement learning for dynamic schedule optimisation in complex construction projects. *Automation in Construction*, 162, 105487. <https://doi.org/10.1016/j.autcon.2024.105487>