A Random-Forest-Based Framework for Predicting Schedule Achievement in Cloud Enterprise Resource Planning Projects Using the Critical Path Method and the Programme Evaluation and Review Technique

J. Haekal[1*], R. Mu'min[2], S.A. Lesmana[3], N.E. Triana[3], Y.D. Suwandari[3], D. Rukmayadi[4] & I.G. Prihanto[5]

ARTICLE INFO

Article details
Submitted by authors 3 Dec 2024
Accepted for publication 13 May 2026
Available online 31 Aug 2026

Contact details
* Corresponding author
jakfat.haekal@esaunggul.ac.id

Author affiliations
1 Department of Industrial Engineering, Universitas Esa Unggul, Jakarta, Indonesia
2 Department of Management, Universitas Negeri Jakarta, Jakarta, Indonesia
3 Faculty of Engineering, Universitas Mercubuana, Jakarta, Indonesia
4 Department of Industrial Engineering, Universitas Bhayangkara Jakarta Raya, Indonesia
5 Research Center for Testing Technology and Standards, National Research and Innovation Agency (BRIN), Tangerang Selatan, Indonesia

ORCID® identifiers
J. Haekal
https://orcid.org/0000-0001-8543-1683

R. Mu'min
https://orcid.org/0009-0007-2789-8016

S.A. Lesmana
https://orcid.org/0000-0002-7546-7778

N.E. Triana
https://orcid.org/0009-0004-2732-626X

Y.D. Suwandari
https://orcid.org/0000-0001-6072-915X

D. Rukmayadi
https://orcid.org/0000-0002-2745-5123

I.G. Prihanto
https://orcid.org/0000-0001-8959-0972

DOI
http://dx.doi.org/10.7166/37-2-3151

ABSTRACT

This paper introduces a robust framework for improving project timeline predictions in cloud-based enterprise resource planning (ERP) implementations by combining traditional project management techniques (the critical path method [CPM] and the programme evaluation and review technique [PERT]) with advanced machine learning methods. The primary problem is that traditional project management techniques such as CPM fail to handle uncertainty in project timelines effectively. The features for the dataset were derived from the CPM and PERT framework and were matched with 4M (man, method, machine, material) to identify which kinds of uncertainties happened in the project. Hypothetically, the random forest algorithm was used as the primary predictive tool, but the research also used a neural network model to do the benchmark in metrics such as accuracy, precision, recall, and F1-score. In the results, random forest outperformed as predicted. By integrating CPM and PERT-derived features, the model identified critical factors, including "method" and "machine", that strongly influence project efficiency. The framework achieved a near-perfect discrimination (AUC: 0.997), demonstrating its effectiveness in navigating uncertainties and enabling data-driven decision-making for enhanced project outcomes.

OPSOMMING

Hierdie artikel stel 'n robuuste raamwerk bekend om voorspellings van projektydlyne in wolkgebaseerde ondernemingshulpbronbeplanning-stelsels (ERP-stelsels) te verbeter deur tradisionele projekbestuurs-tegnieke, naamlik die kritieke-padmetode (CPM) en die program-evaluerings- en hersieningstegniek (PERT), met gevorderde masjienleermetodes te kombineer. Die kernprobleem is dat tradisionele projekbestuurstegnieke soos CPM nie onsekerheid in projektydlyne doeltreffend kan hanteer nie. Die kenmerke vir die datastel is uit die CPM- en PERT-raamwerk afgelei en met die 4M-raamwerk, naamlik mens, metode, masjien en materiaal, in verband gebring om te bepaal watter soorte onsekerhede tydens die projek voorgekom het. Die random forest-algoritme is as die primêre voorspellingsinstrument gebruik, terwyl 'n neurale netwerkmodel ook as maatstaf gebruik is om prestasie volgens metrieke soos akkuraatheid, presisie, sensitiwiteit en F1-telling te vergelyk. Die resultate het, soos verwag, getoon dat die random forest-model beter presteer het. Deur kenmerke wat uit CPM en PERT afgelei is, te integreer, kon die model kritieke faktore identifiseer, waaronder "metode" en "masjien", wat projekdoeltreffendheid sterk beïnvloed. Die raamwerk het 'n byna volmaakte klassifikasieakkuraatheid behaal (AUC: 0,997), wat die doeltreffendheid daarvan demonstreer om onsekerhede te hanteer en datagedrewe besluitneming ter verbetering van projekuitkomste moontlik te maak.

1. INTRODUCTION

In today's competitive landscape, businesses are increasingly relying on sophisticated information systems to drive efficiency and to enhance decision-making processes. For example, Amazon leverages cloud-based infrastructure and data analytics to optimise its supply chain and to deliver personalised customer experiences [1]. Information systems management (ISM) involves the strategic and technical approaches that organisations use to maximise the value of their information systems. Rather than just focusing on technology deployment, ISM ensures that information technology (IT) resources align with business goals to support daily operations and drive strategic initiatives. This alignment is crucial for maintaining competitiveness, especially during system upgrades. Users benefit from enhanced productivity, managers ensure alignment with broader organisational goals, and experts assess technical integration and strategic benefits [2], [3]. Thus, ISM ensures that upgrades improve both efficiency and long-term success by optimising IT resources. As part of ISM, organisations are turning to cloud-based enterprise resource planning (ERP) systems to streamline operations and enhance decision-making capabilities [4]. Cloud ERP represents a type of ERP system that relies on internet-based services and operates on a subscription model, rather than being implemented on-premises in an organisation [5]. This transition to cloud-based solutions reflects the evolving landscape of ISM, where leveraging scalable and flexible technologies is crucial for enhancing efficiency and achieving long-term success.

Implementing cloud ERP systems is pivotal to an organisation's success, particularly in enhancing the quality of decision-making processes. For small and medium-sized enterprises (SMEs), the adoption of cloud-based ERP offers significant benefits, including reduced costs and shorter deployment times. The scalability of cloud ERP allows businesses to expand their operations easily, while improved accessibility ensures that key stakeholders can access critical data anytime, anywhere. Overall, the integration of cloud ERP systems not only streamlines operations but also strengthens the foundation for informed strategic decision-making, driving sustainable growth for SMEs [6], [7], [8]. Cloud ERP implementation aims to ensure that employees can access real-time business information from anywhere, enabling continuity of business activities [5], [9]. ERP systems streamline processes, centralise data, and enhance communication, all of which contribute to increased productivity and greater operational efficiency. By adopting cloud ERP, organisations can further elevate their performance, achieving sustainable improvements in both efficiency and productivity. This combination of streamlined processes and centralised data access allows teams to work more effectively, driving consistent, long-term success [10].

While the benefits of cloud ERP systems are clear, successful implementation is not without difficulties. Delays in project timelines can disrupt workflows and undermine potential gains. Prolonged timelines can create cascading issues, with one of the most critical being the inefficient use of time. Delays in key processes disrupt the entire workflow, affecting interdependent tasks and causing a ripple effect throughout the organisation. As illustrated in Figure 1, a critical project delay occurred when the planned five-month schedule was not achieved. These setbacks can severely undermine the overall success of the ERP initiative, leading to wasted resources, increased costs, and missed opportunities for optimisation. Urgent corrective action is needed to prevent further derailment. ERP implementation projects are inherently complex and often experience delays [11]. Factors influencing implementation time can be identified using design planning techniques, while the critical path method (CPM) can pinpoint bottleneck activities. Project complexity, particularly in "subjective" and "uncertainty" dimensions, is a necessary condition for delays in ERP projects [11].

Therefore, time efficiency is crucial, where strict timelines and resource optimisation can have a significant impact on overall project success. Delays in ERP implementations can lead to substantial cost overruns, as highlighted by Bokovec *et al.*, emphasising the importance of timely project completion to maintain budgetary constraints [12]. Moreover, the findings by Sudhaman and Thangavel [13] demonstrate a robust correlation between time efficiency and project success, indicating that ERP projects completed on time are more likely to deliver enhanced performance metrics, including higher stakeholder satisfaction and increased cost savings [13]. This is especially vital in ERP projects, where alignment between system functionalities and organisational needs is crucial for achieving intended outcomes. Effective time management is also vital to mitigate risks associated with ERP projects. Efficient time management helps to reduce risks related to delays and unexpected complications [12]. In addition, Velcu argues that time-efficient projects enhance team productivity by establishing clear timelines, ensuring that team members are aligned in their efforts for successful deployment [14].

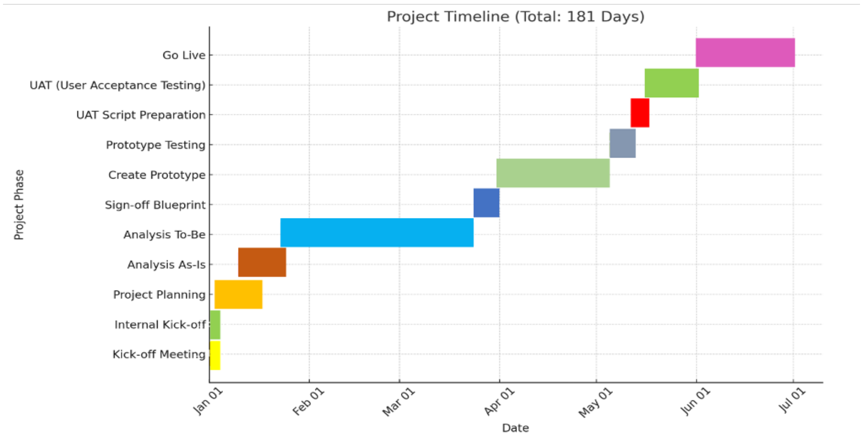

Figure 1: Project timeline (181 days)

Given the potential risks of time delays in ERP implementations, leveraging advanced project management techniques such as the programme evaluation and review technique (PERT) and CPM can help to mitigate these issues. These tools facilitate the efficient management of project activities from inception to completion, thereby enhancing overall project effectiveness [15]. According to Haekal and Masood, future research will benefit from integrating the CPM and PERT methods with random forest, as this combination can enhance the accuracy of time estimation and critical path identification by more deeply analysing historical data [16]. Besides the use of inappropriate methods by the project team, uncertainties that may affect project schedule performance are also present. Project management techniques such as CPM often face problems with uncertainty in complex projects, leading to inaccurate forecasts and resource inefficiencies [17].

In addition, the subdivision of activities in PERT networks can create biases in probability calculations, affecting the calculated probabilities based on the timing of scheduled events [18]. To mitigate these issues, researchers have suggested alternative approaches, such as uncertainty importance measures, to pinpoint activities that require more focus in reducing variability in project completion time [19]. Since uncertainty is an inevitable aspect of all projects, addressing these uncertainties effectively would be crucial for improving overall project outcomes. Random forests offer an advanced approach to addressing uncertainties by analysing historical data and providing deeper insights into factors influencing project outcomes. By aggregating the predictions of multiple decision trees [20], it not only reduces the risk of overfitting and improves the model's generalisation ability, leading to more accurate predictions, but also enhances the precision of time estimation and critical path identification, making it a valuable addition to project management frameworks. Therefore, it optimises project performance and decision-making, offering new avenues for improvement in cloud-based project execution.

The aim of this research is to conduct a comprehensive synergistic application of the random forest algorithm, combined with CPM and PERT, to leverage advanced machine learning techniques to enhance project scheduling, optimise task completion times, and improve the accuracy of project timeline predictions. This study proposes an integrated framework that combines traditional project scheduling techniques with machine learning to address uncertainty in cloud ERP implementation timelines. CPM and PERT are used to provide structured scheduling representations and uncertainty-related information derived from project activities [1], [2], [4]. These scheduling outputs are leveraged within a random-forest-based predictive framework to support the classification of project schedule achievement and to identify influential uncertainty factors affecting project performance [20]. In order to contextualise the performance of the proposed approach, a neural network model is included as a benchmark to support comparative evaluation, rather than as a primary methodological contribution [21]. Random forests are effectively used within modern machine learning frameworks as a powerful supervised algorithm, and are widely applied in data science for both regression and classification tasks [20], [22]. The algorithm can analyse patterns and provide refined probability distributions for task completion times, which can be used to update the CPM and PERT estimations. This innovative combination aims to address uncertainties and to improve project scheduling and decision-making, ultimately enhancing the success of cloud-based ERP initiatives.

This research introduces a refined approach to project management by merging traditional scheduling techniques with advanced machine learning, particularly in cloud ERP projects. While CPM and PERT are widely used to estimate project timelines, their predictive capabilities are often limited by static assumptions. By integrating the random forest algorithm, this approach leverages data-driven insights to adjust task durations dynamically and to uncover patterns that conventional methods might overlook. By combining traditional project management techniques with machine learning, this approach paves the way for a more adaptive, accurate, and intelligent system, capable of navigating uncertainties and delivering timely, cost-effective results. This innovative blend of methodologies stands to enhance significantly the success of cloud-based ERP projects and beyond.

In addition to ensemble-based models, neural networks have been widely applied in project scheduling and duration prediction because of their ability to model nonlinear relationships [23], [24]. In this study, a neural network model is included as a benchmark to evaluate the performance of the proposed random-forest–based framework, allowing a comparative assessment between ensemble learning and neural network approaches.

The remainder of this paper is organised as follows. Section 2 reviews the relevant literature on CPM, PERT, and machine-learning applications in project scheduling. Section 3 describes the proposed framework and research methodology. Section 4 presents the results and discussion, including the evaluation of the proposed random forest model with a supporting benchmark comparison using a neural network. Section 5 concludes the paper and outlines directions for future research.

2. LITERATURE REVIEW

2.1. Critical path method and programme evaluation and review technique

CPM and PERT are classic project scheduling techniques that are widely used to analyse project timelines and identify critical activities. CPM is a deterministic method that represents a project as a network of interdependent activities with fixed durations, enabling the identification of the critical path, which determines the minimum project completion time. Activities on the critical path have zero slack, implying that any delay will directly affect the overall project duration. Given its deterministic nature, CPM is particularly suitable for projects with clearly defined scopes and stable activity durations, and it is commonly applied to identify scheduling bottlenecks and to prioritise critical tasks [15], [25].

PERT extends network-based scheduling by incorporating uncertainty into activity duration estimates. It uses three time estimates (optimistic, most likely, and pessimistic) to compute the expected duration and variance of project activities, allowing for probabilistic analysis of project completion time. This makes PERT more appropriate for complex projects that are characterised by high uncertainty, such as information systems and ERP implementations, where task durations are difficult to estimate accurately [18], [26]. Although CPM and PERT have inherent limitations, including reliance on accurate estimates and assumptions of activity independence, their outputs –such as critical paths, expected project duration, and scheduling baselines – remain valuable inputs for advanced analytical approaches, including machine learning–based project scheduling and performance prediction.

2.2. Machine learning in project scheduling

Recent studies have increasingly explored the application of machine learning (ML) techniques to improve project scheduling, duration estimation, and schedule risk prediction. Traditional scheduling methods, such as CPM and PERT, rely on deterministic or probabilistic assumptions that may not adequately capture the complexity and nonlinearity of real-world projects. To address these limitations, ML models have been applied to predict task durations, schedule delays, and project performance by learning patterns from historical project data. Prior research has demonstrated the effectiveness of ML approaches, including artificial neural networks, support vector machines, and ensemble methods, in enhancing prediction accuracy for project scheduling and construction management problems [24], [27].

Several studies have also investigated hybrid approaches that combine classical scheduling techniques with data-driven models. For example, ML has been used to support schedule risk analysis, activity duration estimation, and early warning systems for project delays, complementing CPM- or PERT-based scheduling frameworks [26], [28]. However, much of the literature focuses on regression-based delay estimation or construction project contexts, with limited attention to classification-based schedule achievement prediction and explicit integration of CPM and PERT outputs as structured inputs for ML models. Furthermore, uncertainty factors related to organisational and technical dimensions are often underexplored. This study addresses these gaps by integrating CPM- and PERT-derived scheduling outputs with uncertainty factors within a random-forest–based classification framework to predict schedule achievement in cloud ERP projects.

Neural network models have been extensively applied in project scheduling and construction management to predict activity durations and overall project performance because of their ability to capture complex nonlinear relationships in project data [23], [29]. Prior studies demonstrate that neural networks can achieve high predictive accuracy in scheduling-related tasks when sufficient historical data are available [23]. However, neural network models often require extensive parameter tuning and relatively large training datasets to achieve stable and generalisable performance, which may limit their practical applicability in projects with limited data availability [29]. Consequently, neural networks are frequently used as benchmark models in comparative studies to evaluate the effectiveness of ensemble-based approaches, such as random forest, in project scheduling and prediction tasks [30].

2.3. The limitations of traditional project management

Use of CPM calculates the longest path of dependent tasks, which forms the critical path, and provides a roadmap for keeping the project on schedule. This method is particularly useful in well-defined projects with predictable timelines, as it ensures that essential tasks are prioritised to prevent bottlenecks. However, CPM's primary limitation is its inability to handle uncertainties in task durations. As Vanhoucke explains, CPM is particularly vulnerable in environments where task durations are uncertain, such as in cloud ERP systems, which often experience fluctuating demands and timelines [25]. When tasks deviate from their planned durations, CPM cannot adapt to these changes, resulting in inaccurate timeline predictions [31]. Although CPM provides a structured approach to managing project tasks, its deterministic nature limits its effectiveness in handling uncertainty.

PERT offers a probabilistic approach to project scheduling and allows for variability in task durations by incorporating three time estimates: optimistic, most likely, and pessimistic. This method offers greater flexibility in managing projects with unpredictable task durations, making it particularly useful for large projects with inherent risks. Despite this flexibility, PERT's reliance on subjective estimates can introduce bias, which, according to Moskowitz and Bullers, may reduce the precision of its predictions, especially in large, complex projects such as cloud ERP [32]. PERT assessments often yield looser subjective probability distributions, leading to fewer surprises but also introducing biases in the estimates. It can lead to cost overruns when projects are behind schedule and cost penalties when ahead of schedule [33]. Conventional PERT procedures introduce biases into time statistics [34]. Thus, while PERT improves on CPM by accounting for uncertainty, it still struggles to adjust dynamically to real-time changes in project conditions. Both CPM and PERT provide valuable frameworks for managing tasks and resources, but their static nature and reliance on estimates leave them ill-equipped to handle the evolving nature of complex projects such as cloud ERP [19].

Several authors have proposed modifications to the original PERT estimators to improve accuracy, such as new approximations for the mean and variance of activity durations. However, Sackey *et al*. argued that the complexity of new methods can make them less practical for real-world applications [35].

2.4. The challenges in a cloud ERP project

Cloud ERP systems are implemented to streamline operations and to improve decision-making in organisations. These systems offer scalability and flexibility, but the projects required to implement them are notoriously complex, involving multiple stakeholders, fluctuating requirements, and external risks. Implementing cloud ERP systems often requires significant organisational change and adaptation, including changes in work arrangements and business processes [7]. Delays in cloud ERP implementations are common, often stemming from resource constraints, technical failures, and evolving project scopes [28]. The impact of these delays is significant, as they can result in increased costs and missed business opportunities.

Meanwhile, CPM and PERT are often inadequate for managing the complexity of cloud ERP implementations [9]. These projects are dynamic, with task durations and dependencies constantly changing. The static nature of CPM and PERT limits their ability to adjust to these changes, making it difficult for project managers to predict timelines accurately and to allocate resources effectively [26]. As a result, there is a need for more advanced tools, such as random forest, to manage the uncertainty and variability inherent in cloud ERP projects. The original PERT model's simplicity is often preferred for ease of use, even if it is less accurate [36], [37], [38]. Integrating random forest algorithms with CPM and PERT methodologies potentially offers better accuracy and predictive capabilities than merely modifying traditional PERT estimators. Random forest's strengths in feature selection, model simplification, and high predictive accuracy make it a promising tool for enhancing project management techniques [39], [40].

2.5. Random forest algorithms in predictive analysis

Random forests can handle large datasets with numerous features without variable selection or dimensionality reduction. They are highly effective in dealing with missing values, and can estimate the importance of different features, which aids in feature selection and understanding the underlying data structure [41]. This capability makes random forests particularly valuable in complex applications in various domains. In project management, random forests can improve the accuracy of task duration estimation and schedule planning. The proposed calibration process using scaling algorithms enhances these models by reducing overfitting and avoiding unwanted biases, especially when dealing with uncertain factors such as resource constraints or task dependencies. For instance, when managing large projects such as ERP implementations, uncertainties in task durations and project schedules can be better managed by applying these calibrated random forest models [27]. This reduces the risk of inaccurate predictions and leads to more reliable decision-making throughout the project lifecycle.

Random forests benefit from the ensemble method's inherent capability to reduce variance and improve prediction accuracy. Each tree in the forest is built using a random subset of the data and features, which ensures diversity among the trees and reduces the correlation between them [20]. It leads to improved generalisation on unseen data, and enhances model performance. They can also be implemented using parallel computing frameworks, which allows for faster training and prediction times, especially on big data [42]. The ability to parallelise the construction of decision trees and the use of bootstrapped samples contributes to their efficiency and effectiveness. Techniques such as random forests with extra trees (extremely randomised trees) have also been developed to increase the randomness in tree building [43], which can improve performance and reduce training time. In addition, random forests have been integrated with other machine learning techniques, such as boosting and stacking, to enhance their predictive power and versatility [43], [44]. Advancements in machine learning techniques further illustrate the versatility and power of random forests in refining project management practices.

The literature reveals that the integration of traditional project management techniques such as CPM and PERT with machine learning methods such as random forests offers a promising approach to managing the complexities and uncertainties of modern projects [24], [45]. This hybrid model enhances predictive accuracy and adaptability, providing project managers with better tools to optimise project outcomes. Further research into these hybrid models is necessary to address both theoretical and methodological gaps, ultimately improving project management practices in various industries.

3. RESEARCH METHODS

The study began by collecting relevant project data from previously implemented cloud ERP systems, focusing on task durations, dependencies, and resource allocations. Historical project data, including delays, task complexity, resource utilisation, and external factors affecting the project timeline, were compiled from various sources, including project management databases and case studies, cloud ERP implementation case studies, project management reports, and other relevant literature; task durations, task dependencies, resource allocations, external factors (e.g., market conditions, technological limitations), and recorded delays. CPM and PERT were applied to the collected data to define the critical path, estimate task durations, and identify areas of uncertainty in the project timeline. The random forest algorithm was introduced to enhance the accuracy of task duration predictions by analysing complex interactions between variables.

3.1. Research procedure

This study followed a structured research workflow consisting of data collection, data preprocessing, model development, training, and evaluation. Project data were collected from previously implemented cloud ERP projects and included CPM- and PERT-derived scheduling outputs as well as uncertainty factors based on the 4M framework (man, method, machine, and material). In this study, CPM and PERT were applied solely as preliminary scheduling tools to derive baseline project duration indicators. The scheduling outputs obtained from CPM and PERT were used as input features for the subsequent machine learning models, rather than as standalone analytical methods.

Prior to modelling, the dataset was examined for inconsistencies and missing values. All variables were numerical and standardised using a scaling procedure to ensure comparable feature ranges and to improve model stability. The prediction task was formulated as a binary classification problem, in which the target variable indicated whether or not the project schedule target was achieved.

A random forest classifier was used as the primary predictive model, while a neural network model was included as a benchmark for comparison. Model training and evaluation were conducted using the Orange data mining platform. The dataset was divided into training and testing subsets using a hold-out validation approach.

Model performance was assessed using multiple evaluation metrics, including accuracy, precision, recall, F1-score, and the area under the receiver operating characteristic curve (AUC). Confusion matrices were also used to analyse classification outcomes and misclassification patterns. Data preparation, model development, training, and evaluation were conducted using the Orange data mining platform, which provides an integrated and reproducible workflow for machine learning experiments.

3.2. Dataset building

The dataset for this research was constructed using the 4M framework as independent variables to analyse their influence on project outcomes. Each of these variables was quantified using a structured scoring system (1–10 scale) to reflect their maturity, readiness, and alignment with ERP project requirements. The details are provided in Table 1.

Table 1: Scoring for uncertainty variable dataset

Variable	Dimension	Score	Description
(Man) x1	Skill & competence, change readiness, engagement, leadership support, communication, training Hours [46]	1-3	Workforce lacks ERP knowledge, engagement, and readiness; no leadership support or communication.
		4-6	Basic ERP skills; moderate engagement; some leadership support; inconsistent communication.
		7-9	Skilled, engaged, and adaptive workforce; strong leadership support and clear communication.
		10	Fully proficient, committed, and proactive workforce; excellent leadership and communication.
(Method) x2	Process alignment, SOPs, change management strategies, customisation effort, implementation timeline [14], [47]	1-3	ERP misaligned with processes; no SOPs or change management strategy.
		4-6	Partial alignment; basic SOPs and some change management strategies exist but need improvement.
		7-9	Well-aligned processes; detailed SOPs; effective change management with minor gaps.
		10	Processes fully aligned; comprehensive SOPs; excellent change management strategy.
(Machine) x3	System compatibility, performance metrics, scalability, data migration	1-3	Incompatible ERP system; poor performance and data integration; inadequate technical support.

Variable	Dimension	Score	Description
	success rate, integration, IT support [46]	4-6	Basic functionality; moderate compatibility; partial integration with other systems.
		7-9	Reliable ERP system with good compatibility, performance, and integration.
		10	Highly efficient, scalable, and seamlessly integrated ERP with excellent technical support.
(Material) x4	Budget use, resource availability, project cost, quality of input data, vendor support [48], [49]	1-3	Insufficient budget, poor resource allocation, and inaccurate data.
		4-6	Adequate budget and resources; some inefficiencies in allocation or data quality.
		7-9	Well-planned resource allocation; clean data and good vendor support.
		10	Optimal resources, budget, and data quality; excellent vendor collaboration.

The target completion time was established using CPM or PERT analysis, depending on the nature and complexity of the project under evaluation. CPM would be applied to projects with well-defined tasks and deterministic timelines, whereas PERT would be used for projects with probabilistic time estimates, allowing for an assessment of variability and uncertainty in task durations.

To evaluate the project status, the actual completion time was compared against the target time predicted by CPM or PERT. Projects that met or completed tasks within the predicted timeframe were classified as “achieved”, while those exceeding the target were labelled “not achieved”. This binary classification served as the basis for analysing the relationship between the 4M variables and the success of ERP implementation projects. By leveraging this dataset, the study aimed to identify critical factors influencing project timelines and to assess how the 4M dimensions affected the likelihood of achieving planned project objectives.

3.3. Machine learning integration: Random forest algorithms

These processes are presented in Figure 2, describing the initial step to obtain the required result from the data with the features derived from CPM & PERT analysis, as follows:

This framework delineated a structured approach to predictive analysis using a dataset generated from CPM and PERT analysis – methodologies widely applied in the project for scheduling tasks and estimating durations. The process began with the selection of significant features from the dataset, a critical step in reducing dimensionality and optimising model performance by concentrating on the most relevant variables. Next, the random forest algorithm was implemented; this ensemble learning method constructs multiple decision trees to enhance predictive accuracy and to mitigate the risk of overfitting. Following this, the model’s performance was rigorously evaluated through metrics to determine its effectiveness. The final phase involved interpreting the outcomes, which might include predictions and classifications derived from the random forest model, thus contributing valuable information for informed decision-making.

Figure 3 shows the flow of predictive analysis; the researcher used the machine learning (ML) software tool known as Orange, which supports the implementation of predictive models efficiently. To begin, the dataset was imported into Orange, where the “data training” feature was used to provide a clear, organised view of the data, facilitating initial inspection and analysis. Next, before initiating model processing, the researcher conducted a feature selection step by using it to identify and retain only the most representative columns, ensuring that the input data was both relevant and optimised for modelling. Once the dataset had been prepared, the random forest model processed it. This stage involved two key components: the first was bootstrap aggregation (bagging), in which the model performed random sampling with replacement of the training data, generating multiple subsets for training. This bagging technique reduced variance and enhanced model robustness. Second came the feature selection for decision trees: each decision tree in the ensemble was built with a randomly selected subset of features, creating diverse decision paths and minimising the risk of overfitting [20], [43], [44]. After that, multiple decision trees were constructed, each learning unique patterns in its subset.

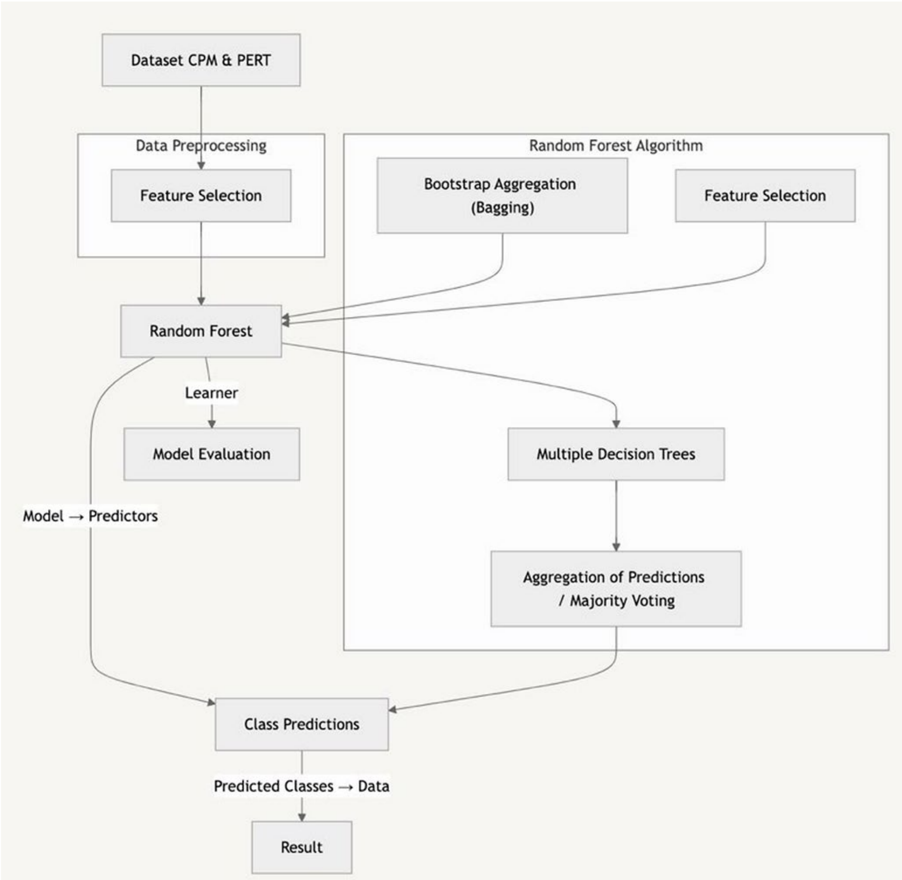

Figure 2: Analysis framework

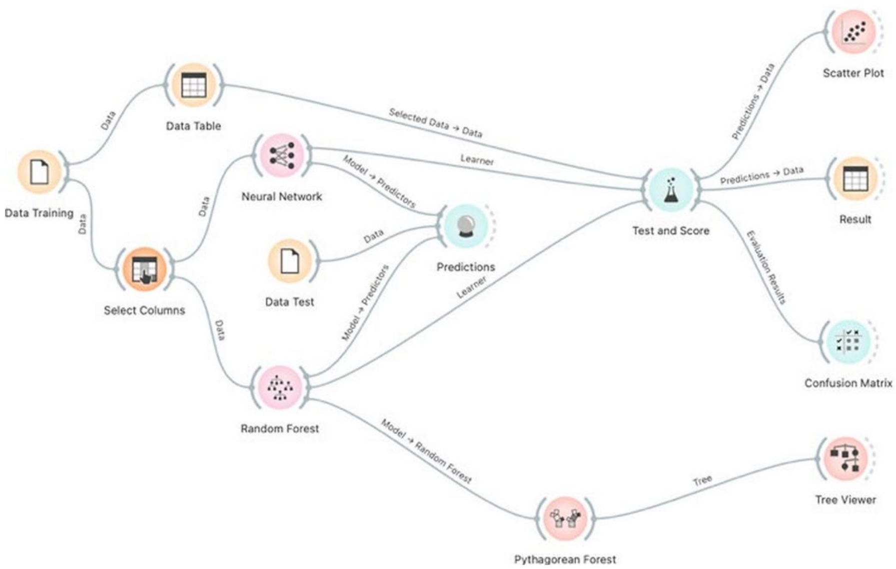

Figure 3: Analysis workflow on Orange

In addition to the random forest model, a neural network algorithm was incorporated as a comparative benchmark. Neural networks, known for their capability to model complex, non-linear relationships, serve as a robust alternative to random forest. However, based on the literature and on theoretical assumptions, it was hypothesised that the random forest model would yield superior predictive accuracy owing to its capacity to handle smaller datasets and its inherent resistance to overfitting compared with neural networks, which often require extensive tuning and larger datasets to achieve optimal performance. This hypothesis was rigorously tested by comparing the results of both models. Following model training, performance evaluation was conducted using established metrics to ensure the reliability and validity of the findings. The final phase involved the interpretation of the results, encompassing the predictions and classifications generated by the models.

3.3.1. Random forest classification formula

To classify a variable (e.g., predict whether an activity will be delayed or not), the formula used is:

$$\hat{y} = mode(y_1, y_2, \cdots, y_B) \quad (1)$$

where

$\hat{y}$	Predicted class for the target variable (e.g., delay: yes/no).
y_i	Predicted class from the i-th tree.
mode	Function that returns the most frequently occurring value among the tree predictions.

3.3.2. Performance analysis

In classification tasks, performance evaluation is crucial for understanding how well a model is doing. Here are some key metrics and their formulas used to evaluate classification models:

Accuracy measures the proportion of correctly classified instances out of the total instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

where

TP (true positives)	Correctly predicted positive instances
TN (true negatives)	Correctly predicted negative instances
FP (false positives)	Incorrectly predicted positive instances
FN (false negatives)	Incorrectly predicted negative instances

Precision measures the proportion of true positive predictions among all positive predictions.

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

Recall measures the proportion of true positive predictions among all actual positive instances.

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

The F1 Score is the harmonic mean of precision and recall, providing a balance between the two metrics.

$$F1\,Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (5)$$

A confusion matrix summarises the performance of a classification algorithm. It is a table with four combinations of predicted and actual values. Table 2 presents the confusion matrix used to evaluate the classification performance of the proposed model in predicting project schedule achievement. True positive (TP) represents cases where projects that successfully met the target schedule were correctly classified as "target achieved," while true negative (TN) corresponds to projects that failed to meet the target and were correctly classified as "not achieved". False positive (FP) indicates cases where delayed projects were incorrectly classified as achieving the target, whereas false negative (FN) represents projects that achieved the target but were misclassified as not achieved.

Table 2: Confusion matrix

		Prediction values	
		Predicted positive	Predicted negative
Actual values	Actual positive	TP	FN
	Actual negative	FP	TN

4. RESULTS AND DISCUSSION

4.1. Dataset description

This subsection describes the dataset used in the experimental analysis. It consisted of 100 project instances derived from cloud ERP implementation cases. Each instance was characterised using the 4M framework, while project status was defined as a binary classification outcome ("target achieved" or "not achieved"), based on the comparison between actual project duration and the target duration estimated using CPM or PERT.

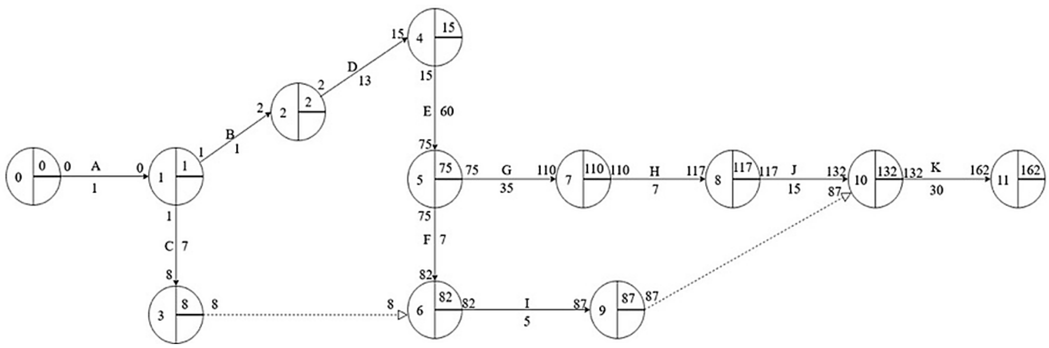

Figure 4: Map of critical path method

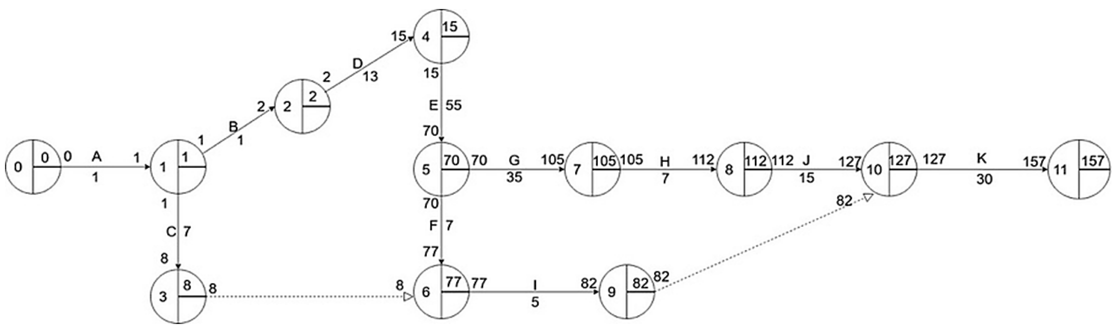

Figure 5: Map of programme evaluation and review technique

Activities A, B, D, E, G, H, J, and K were identified as critical activities located on the critical path, meaning that any delay in these activities would directly delay the overall project completion time. Activity A represents the project kickoff meeting, followed by Activity B, which corresponds to the internal team kickoff meeting. Activities D and E involve business process analysis, where Activity D captures the "as-is" process assessment and Activity E focuses on the "to-be" process design. Activity G covers prototype development and system configuration, which is subsequently evaluated through Activity H, the prototype testing phase. User validation is conducted in Activity J through user acceptance testing (UAT), and the implementation culminates in Activity K, which represents the "go live" phase.

Figure 4 presents the CPM analysis, while Figure 5 represents the PERT. Both diagrams illustrate the sequence from node 0 to 11, with CPM estimating a project duration of 162 working days and PERT indicating a shorter duration of 157 working days. This suggests that PERT was more efficient in reducing project time, and thus, the PERT duration would serve as reference data in building a random forest dataset for prediction and classification.

In this dataset, the 4M factors were incorporated as independent variables to identify which types of uncertainty most influenced project timelines. By analysing these factors, the model could detect the variability sources that had a significant impact on deviations from the PERT-based duration [19]. By examining each 4M variable in relation to the PERT duration, the model could capture patterns of uncertainty and reveal key risk areas. For each factor, a higher score indicated a greater influence on the actual project duration, meaning that addressing high-impact variables had the potential to reduce overall project time significantly. This insight was critical for prioritising areas where efficiencies could be gained, ultimately accelerating project completion. The predictive model assessed these factors individually to quantify their accuracy and impact, identifying the specific sources of delays and enabling more targeted risk mitigation.

Table 3: Dataset variables

	Project Status	Man	Method	Machine	Material	Actual	Target
1	Not Achieved	7	2	6	5	172.391	157
2	Not Achieved	4	1	1	8	172.603	157
3	Target Achie...	8	7	9	1	151.041	157
4	Target Achie...	5	7	6	5	153.14	157
5	Target Achie...	7	8	3	3	155.068	157
6	Not Achieved	10	5	4	1	162.889	157
7	Not Achieved	3	3	4	4	184.219	157
8	Target Achie...	7	8	3	5	149.425	157
9	Target Achie...	8	6	10	7	147.26	157
10	Not Achieved	5	3	3	1	184.842	157

The dataset in Table 3 provides an overview of various features, denoted as 4M, with “project status” as the target variable categorised as either “target achieved” or “not achieved”. These features were characterised by numerical data types; meanwhile the target variable was characterised by categorical data types, allowing for the direct identification of classification [20], [40]. The dataset comprised 100 data points, each representing a unique set of observations that corresponded to the target time to decide whether “achieved” or “not achieved” (project status), recorded based on the interaction of the aforementioned features. Table 3 presents quantitative scores for each of the 4M components, alongside the actual performance outcomes compared with a consistent target of 157. Projects achieving the target exhibited balanced contributions for all 4M factors, indicated by red bars, whereas projects that fell short highlight gaps (blue bars). The dataset served as a foundational resource for training the model, ensuring that the predictions were based on real-world observations and not merely on theoretical assumptions. The scoring system presented in Table 3 was derived from a structured questionnaire designed to assess the performance and efficiency of various factors influencing the project. Each 4M factor was evaluated through responses collected from related parties. The questionnaire used a ranking scale from 1 to 10, where respondents assigned higher scores to indicate greater efficiency and shorter completion times, and lower scores to reflect inefficiencies or delays in the respective factor.

Another dataset would serve as the test data, and would include only the independent variables, namely the factors from the 4M framework. This dataset would be used to evaluate the predictive capabilities of the trained model by determining the project status (either “target achieved” or “not achieved”) based on the patterns and relationships learned during the training process. It ensured that the model’s performance could be assessed objectively by comparing its predictions with the actual outcomes in unseen data.

Table 4: Table of comparison between random forest and neural network algorithms

	Random Forest	Neural Network	Man	Method	Machine	Material
1	0.14 : 0.86 → Target Achieved	0.00 : 1.00 → Target Achieved	10	10	10	10
2	0.44 : 0.56 → Target Achieved	0.65 : 0.35 → Not Achieved	5	5	5	5
3	0.01 : 0.99 → Target Achieved	0.01 : 0.99 → Target Achieved	7	7	7	7
4	0.99 : 0.01 → Not Achieved	1.00 : 0.00 → Not Achieved	1	1	1	1
5	0.95 : 0.05 → Not Achieved	1.00 : 0.00 → Not Achieved	2	2	2	2
6	0.99 : 0.01 → Not Achieved	0.99 : 0.01 → Not Achieved	3	3	3	3
7	0.88 : 0.12 → Not Achieved	0.96 : 0.04 → Not Achieved	4	4	4	4
8	0.44 : 0.56 → Target Achieved	0.65 : 0.35 → Not Achieved	5	5	5	5
9	0.99 : 0.01 → Not Achieved	1.00 : 0.00 → Not Achieved	1	1	1	1
10	0.58 : 0.42 → Not Achieved	0.72 : 0.28 → Not Achieved	9	2	5	4

Table 4 compares the predictive performance of two machine learning models, random forest and neural network, in classifying project status (“target achieved” or “not achieved”) based on the 4M factors. Each row represents a test instance with scores assigned by the models to indicate the probability of the project achieving its target. For example, the random forest model assigned probabilities to each class, such as 0.14 (not achieved) and 0.86 (achieved), with the higher value determining the classification. Similarly, the neural network provided its probabilities and corresponding predictions. The dataset highlighted differences in model outputs, demonstrating cases where the models agreed or differed, which could provide insight into their decision-making processes. The independent variables, 4M, served as input features, and their values indicated the resource allocation or efficiency for each test case.

4.2. Model configuration and benchmark setup

The primary predictive model used in this study was a random forest classifier, selected for its robustness in handling uncertainty, nonlinear relationships, and feature interactions in project scheduling data [20]. The random forest model served as the core analytical component for predicting project schedule achievement. To provide a supporting comparative evaluation, a neural network model was also implemented as a benchmark. Specifically, a feedforward multilayer perceptron (MLP) architecture with a single hidden layer was used, trained using the back-propagation learning algorithm. The neural network was included solely for benchmarking purposes, allowing the performance of the proposed random forest framework to be contextualised rather than serving as a primary methodological contribution.

4.3. Model training and evaluation

Model performance was evaluated using standard classification metrics commonly applied in predictive analytics, including accuracy, precision, recall, F1-score, area under the curve (AUC), and the Matthews correlation coefficient (MCC) [20], [44]. These metrics were used to assess the ability of the models to classify project schedule status correctly under uncertainty. Performance evaluation was conducted consistently for both the random forest and the benchmark neural network models to ensure a fair comparison. The resulting metrics provided the basis for analysing predictive effectiveness and for interpreting the relative strengths of the proposed framework in comparison with the benchmark approach.

Table 5: Accuracy rate comparison between random forest and neural network algorithms

Model	AUC	CA	F1	Precision	Recall	MCC
Random forest	0.997	0.980	0.980	0.980	0.980	0.960
Neural network	0.952	0.900	0.900	0.901	0.900	0.800

Table 5 presents the performance metrics of two machine learning models, random forest and neural network, that were used to classify project status. The metrics included AUC, classification accuracy (CA), F1-score (F1), precision (Prec), recall, and the MCC [44]. Random forest outperformed the neural network in all metrics, achieving an AUC of 0.997 compared with 0.952 for the neural network, indicating superior model discrimination ability. Similarly, the random forest demonstrated higher CA (0.980), F1 (0.980), Prec (0.980), recall (0.980), and MCC (0.960), showcasing its robustness in capturing the relationship between input features and the project status while maintaining a low misclassification rate. These results underlined the importance of model selection to achieve high predictive accuracy and reliability in classifying project outcomes. The significant performance gap in metrics such as MCC (0.960 vs 0.800) highlighted the random forest’s ability to provide more balanced and consistent predictions between classes. The superior AUC further validated its capability to distinguish between “target achieved” and “not achieved” cases with minimal overlap. The findings emphasised that random forest was a better-suited model for this task, providing reliable, high-quality predictions that could aid decision-making in this ERP project’s implementation duration.

The researcher used Pythagorean forest to visualise a random forest, providing an intuitive and structured representation of this machine learning model by leveraging the fractal and hierarchical properties of Pythagorean trees [43]. Each decision tree in the random forest could be depicted as a fractal structure, such that larger shapes at the base represented root nodes or higher-level splits, and smaller shapes corresponded to deeper, more granular decisions. This multi-scale representation aligned naturally with the recursive nature of decision trees, making it easier to grasp both the global structure and the finer details of how features contributed to predictions. It helped the random forest model to be more transparent and easier to analyse.

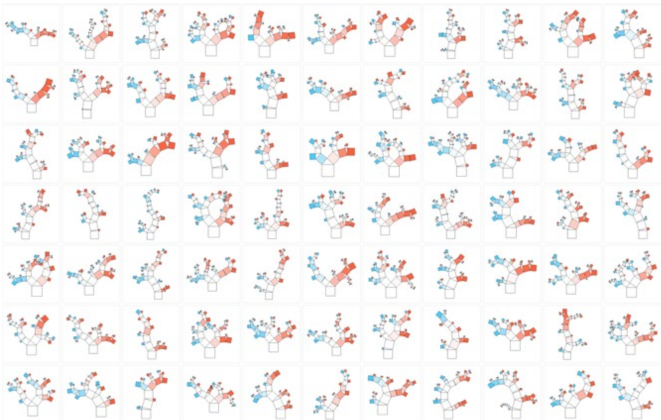

Figure 6: Pythagorean forest

Figure 6 represents the Pythagorean forest, which in consisted of 100 trees. Each tree is a fractal structure that models hierarchical or recursive relationships in data, and depicts a hierarchical dataset, where the branching structure represents parent-child relationships. Larger fractal elements at the base of the tree correspond to root nodes or higher-level data points, while progressively smaller elements illustrate deeper levels of recursion, representing subordinate or granular components. This recursive visual representation helps to convey the depth and complexity of hierarchical systems intuitively. The use of colour further enhances the interpretability of the visualisation. Distinct colours, such as red and blue, differentiate hierarchical layers. Parent nodes are represented in one colour, signifying higher-order data, while child nodes are shown in a contrasting colour, representing more detailed or dependent data. This colour-coding system effectively illustrates the structural relationships in each tree, facilitating the recognition of hierarchical layers.

The forest’s 100 trees each differed in structure, depth, and branching complexity. These variations illustrate the adaptability of Pythagorean trees to represent diverse datasets or hierarchical systems. By visualising different structures, the forest highlights the potential for fractal patterns [50], [51].

To show the details of each tree, the researcher used the four decision trees representing the 4M factors, which illustrated the structure of a decision tree. This tree consisted of 27 nodes and 14 leaves, and was simplified to a depth of three levels.

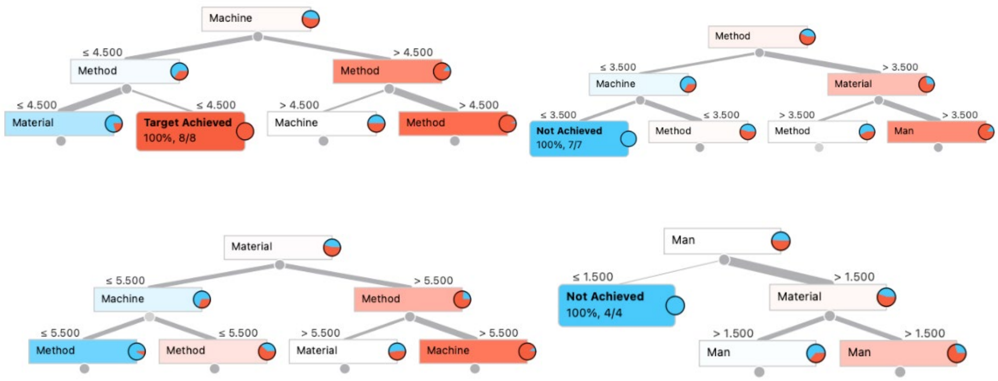

Figure 7: Decision tree

The decision trees showcased the hierarchical structure of nodes and leaves, with each node representing a decision point based on a feature's value and each leaf indicating the outcome (e.g., "target achieved" or "not achieved"). The red and blue colours signified different classes or outcomes, with the pie charts in the nodes further indicating the proportion of data supporting each classification. The branching logic in each tree was based on specific thresholds of feature values, which enabled the tree to focus on the most relevant splits at each level and to balance simplicity with predictive accuracy, represented by inequalities such as ≤ 4.500 or > 3.500. These thresholds determined the flow from parent nodes to child nodes, reflecting the decision-making process in splitting the data. Nodes at higher levels represented broader criteria, while deeper nodes captured more specific conditions.

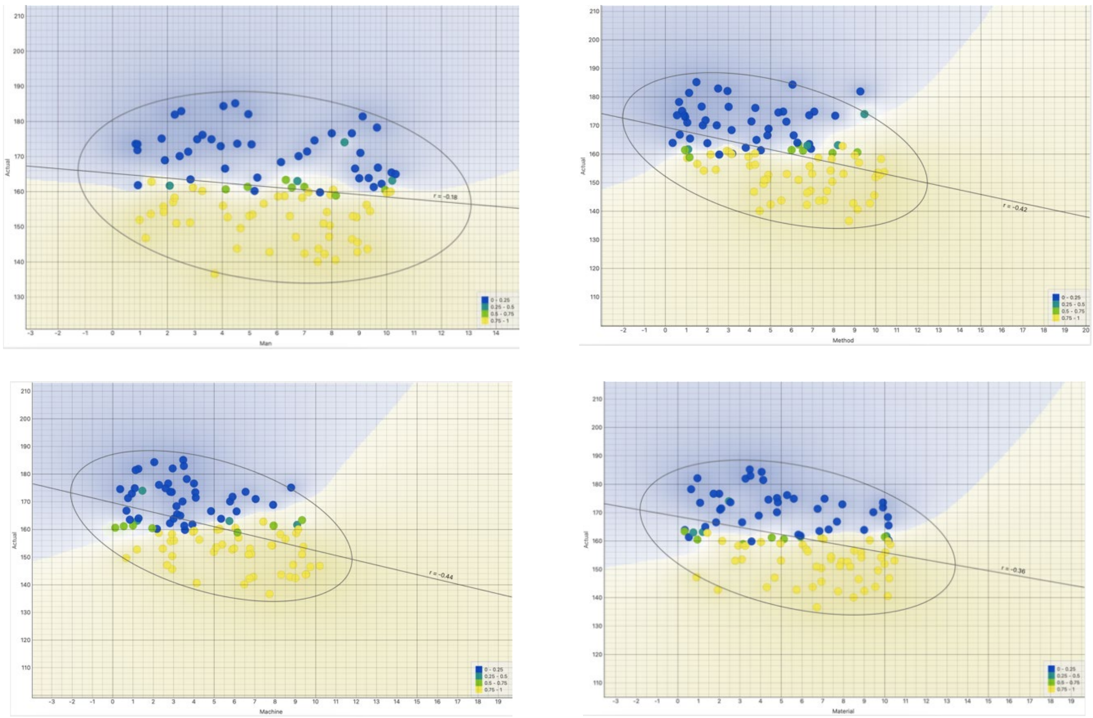

Figure 8: Scatter plot

In Figure 8, the "actual" variable refers to the actual project completion time, measured in working days, recorded for each ERP implementation instance and compared against the target duration estimated using CPM or PERT. This scatter plot visualisation has four panels, each comparing one 4M factor against the "actual" variable, which likely represented the time taken to complete a project or some other performance metric. The X-axes of the plots represent these factors, while the Y-axis remains consistent as the “actual” variable in all four panels. The plots aim to highlight correlations between the chosen factors and their influence on performance.

Each plot includes a correlation coefficient r, which quantifies the strength and direction of the relationship between the X and Y variables; it determines whether the relationship is positive (both variables increase together) or negative (one variable increases as the other decreases). A negative correlation ($r<0$) suggests that, as the factor increases, the “actual” variable decreases, which might correspond to shorter project times. In this case, “method” ($r=-0.42$) and “machine” ($r=-0.44$) show moderately negative correlations, indicating that these factors were significant contributors to reducing project time. Conversely, “man” ($r=-0.18$) had a weak negative correlation, suggesting that personnel or team performance had a minimal impact compared with other factors. “Material” ($r=-0.36$) also shows a moderate effect, highlighting its importance in achieving efficiency.

The background shading – blue and yellow – likely represents performance zones. The blue areas could signify high performance or shorter project times, while the yellow areas may represent lower performance or longer times. These zones, combined with the ellipses encircling data clusters, provide a visual indication of the spread and concentration of data points. The trend lines in each plot further clarify the overall relationship between the variables. Coloured data points add another layer of information, categorising observations into performance bands. Blue points indicate high-performing cases, green points represent moderate performance, and yellow points signify low performance. This categorisation helped to identify outliers and areas requiring improvement.

The practical implications of this analysis are significant for project time optimisation. The “method” and “machine” factors showed the strongest correlations with project time, suggesting that optimising processes and improving tools or machinery could yield the most substantial reductions in time. “Material” also played a key role, indicating that better material selection could contribute to efficiency. The weaker correlation for “Man” suggests that personnel adjustments alone may not lead to meaningful time savings.

Table 6: Comparison of ML models (random forest and neural network), whether “target achieved” or “not achieved”

	Project Status	Random Forest	Neural Network	m Forest (Not Ach	Forest (Target Ac	Network (Not Ach	letwork (Target Ac	Fold	Man	Method	Machine	Material	Actual	Target
1	Not Achieved	Not Achieved	Not Achieved	0.81	0.19	0.919026	0.080974	1	7	2	6	5	172.391	157
2	Target Achie...	Target Achie...	Target Achie...	0.09	0.91	0.110178	0.889822	1	7	8	3	3	155.068	157
3	Not Achieved	Not Achieved	Not Achieved	0.99	0.01	0.998745	0.00125505	1	5	3	3	1	184.842	157
4	Not Achieved	Not Achieved	Not Achieved	1	0	0.999458	0.000541915	1	4	1	3	3	185.773	157
5	Not Achieved	Not Achieved	Not Achieved	0.97	0.03	0.978573	0.0214267	1	8	3	4	2	174.666	157
6	Target Achie...	Target Achie...	Target Achie...	0	1	0.00185738	0.998143	1	8	5	7	9	141.479	157
7	Target Achie...	Target Achie...	Target Achie...	0.09	0.91	0.125691	0.874309	1	5	5	1	10	150.704	157
8	Not Achieved	Not Achieved	Target Achie...	0.81	0.19	0.396452	0.603548	1	6	7	2	8	165.794	157
9	Target Achie...	Target Achie...	Target Achie...	0.27	0.73	0.035052	0.964948	1	10	3	9	2	155.663	157
10	Not Achieved	Not Achieved	Not Achieved	0.97	0.03	0.88677	0.11323	1	10	4	3	1	165.491	157

Table 6 provides further insight into the relationships discussed in the scatter plot analysis by detailing the metrics related to project status, predictive model results, and the 4M factor values for each project instance. It also includes the actual project timelines and their corresponding targets. Together with the scatter plot findings, this table allows a deeper understanding of how individual factors and predictive modelling metrics influence project outcomes and timeline reductions. The project status column categorises outcomes as either “achieved” or “not achieved”, based on whether the actual timeline met or fell below the target. Instances where the target was achieved generally showed lower actual values compared with the target, which indicated successful timeline reductions. The predictive probabilities from the random forest and neural network models further highlighted the reliability of these predictions, with higher probabilities (e.g., close to 0.99) aligning with the achieved outcomes. This demonstrates the utility of machine learning models in identifying patterns and guiding decisions to prioritise impactful changes in project factors. The factor ratings for “man”, “method”, “machine”, and “material” aligned closely with the scatter plot findings regarding the influence of these variables. Projects that achieved their targets often showed optimised ratings for critical factors such as “method” and “machine”. For example, instances where “method” and “machine” values were high often corresponded to lower actual timelines, consistent with their moderate negative correlations ($r=-0.42$, and $r=-0.44$ respectively) observed in the scatter plots. Conversely, “man”, which had a weaker correlation ($r=-0.18$), exhibited less variation between rows, and did not appear as influential in achieving target timelines.

Table 6 also highlights the difference between the actual and target timelines. For rows where the actual timeline significantly exceeded the target, the corresponding factor ratings tended to be suboptimal, leading to a “not achieved” status. On the other hand, projects with “target achieved” status exhibited balanced or optimised factor ratings. These instances reflected the trends observed in the scatter plots, where points in the blue-shaded high-performance regions often corresponded to better factor values and shorter timelines. This consistency between the scatter plot and the table data underscored the importance of focusing on the most influential factors to achieve efficiency. The inclusion of ML model outputs in Table 6 further emphasised the role of predictive analytics in decision-making. High probabilities from the random forest and neural network models in cases where targets were achieved validated the accuracy of these models in identifying successful outcomes.

A confusion matrix is provided below, comprehensively representing a classification model's predictions compared with the actual outcomes, as mentioned above in the section on methods. It is a fundamental tool for evaluating the effectiveness of classification algorithms [43], as it provides a breakdown of correct and incorrect predictions for all classes. The confusion matrices below compare the performance of two machine learning models, neural network and random forest, in predicting whether a target is "achieved" or "not achieved".

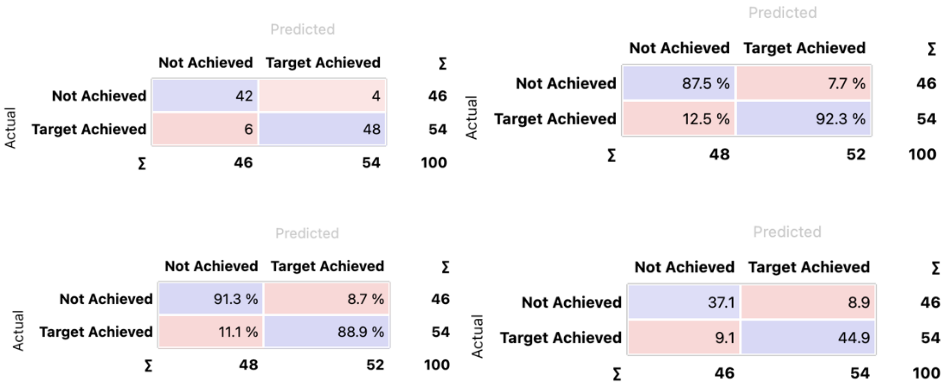

Figure 9: Neural network confusion matrix

In the numeric matrix, the neural network correctly predicted "not achieved" in 42 cases (true negatives) and "target achieved" in 48 cases (true positives). However, it incorrectly predicted "target achieved" for four cases of "not achieved" (false positives) and failed to predict "target achieved" for six cases (false negatives). These misclassifications translated into a classification accuracy of 87.5% for "not achieved" and 92.3% for "target achieved". Despite these figures, the presence of misclassifications slightly diminished the reliability of the neural network, particularly for predicting "not achieved", where it showed higher false positive rates. Overall, the model demonstrated reasonable performance, with an overall correct classification rate of about 90%.

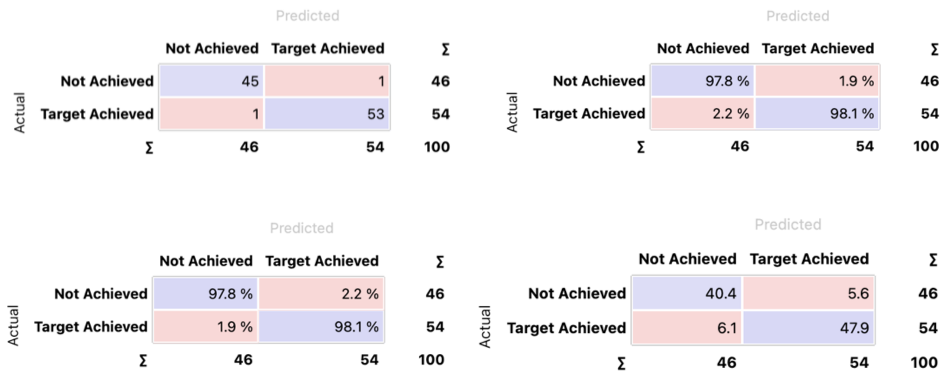

Figure 10: Random forest confusion matrix

By contrast, the random forest model exhibited markedly superior performance. The confusion matrix reveals that the random forest correctly classified 45 cases of "not achieved" (true negatives) and 53 cases of “target achieved” (true positives), with only a single misclassification in each category. Specifically, it misclassified one instance of “not achieved” as “target achieved” (false positive) and one instance of “target achieved” as “not achieved” (false negative). These results correspond to an accuracy of 97.8% for “not achieved” and 98.1% for “target achieved”. Moreover, the random forest achieved almost perfect sensitivity and specificity, making it highly reliable for both classes.

Based on these matrices, the random forest outperformed the neural network in all major performance metrics, including accuracy, sensitivity, and specificity. While the neural network showed acceptable predictive performance, its higher rates of false positives and false negatives indicated room for improvement. The random forest, on the other hand, demonstrated near-perfect classification capabilities with minimal error, suggesting that it was the more effective model for this task. These results suggest that the random forest would be the preferred choice when accuracy is critical, although the neural network may still be viable if additional optimisation is undertaken or if computational constraints favour its use.

5. CONCLUSION

This research demonstrates a robust framework for enhancing project timeline predictions in cloud-based ERP implementations through the integration of traditional project management techniques (CPM and PERT) with advanced machine learning methods. The random forest algorithm was used as the primary predictive tool, while a neural network model served as a comparative benchmark. The analysis revealed that random forest outperformed neural network in all key metrics, including accuracy, precision, recall, and the F1-score, underscoring its effectiveness in handling complex, feature-rich datasets with inherent uncertainties. The random forest model achieved a near-perfect discrimination with an AUC of 0.997 and high reliability in predicting project statuses (“achieved” or “not achieved”). In contrast, the neural network, while performing well, displayed higher misclassification rates, particularly in distinguishing “not achieved” cases, with an AUC of 0.952. These findings support the hypothesis that the random forest model was better suited for this task because of its ensemble nature, which mitigated overfitting and adapted effectively to smaller datasets and non-linear patterns in the data. By incorporating CPM- and PERT-derived features into the random forest model, the framework successfully identified critical variables influencing project delays, particularly the “method” and “machine" factors, which exhibited the strongest correlations with project timeline efficiency. The comprehensive performance evaluation and visualisations (e.g., scatter plots, confusion matrices) reinforced the practical utility of the random forest algorithm in prioritising areas for improvement.

Future work will include automated workflow optimisation for a natural extension of the proposed framework, enabling proactive adjustments to project plans based on predictive insights. By integrating prescriptive analytics, the system could recommend actions such as resource reallocation, timeline adjustments, or alternate methods based on the 4M features to mitigate delays. For instance, if the problem that mostly causes delay is “man”, then the mitigation that should be taken would probably be redistributing tasks, providing training, or reallocating personnel. These tailored actions directly address root causes, enabling proactive and effective project management. This approach shifts the focus from prediction to actionable strategies, reducing delays, optimising resources, and increasing overall project success rates.

REFERENCES

[1] **T. H. Davenport**, *The AI advantage: How to put the artificial intelligence revolution to work*. The MIT Press, 2018.

[2] **Y. A. Pollalis**, “Patterns of co-alignment in information-intensive organizations: Business performance through integration strategies,” *International Journal of Information Management*, vol. 23, no. 6, pp. 469–492, 2003. doi: https://doi.org/10.1016/S0268-4012(03)00063-X

[3] **G. S. Kearns and A. L. Lederer**, “The effect of strategic alignment on the use of IS-based resources for competitive advantage,” *The Journal of Strategic Information Systems*, vol. 9, no. 4, pp. 265–293, 2000. doi: https://doi.org/10.1016/S0963-8687(00)00049-4

[4] **B. K. Muslmani, S. Kazakzeh, E. Ayoubi, and S. Aljawarneh**, “Retracted on September 10, 2025: Reducing integration complexity of cloud-based ERP systems,” in *Proceedings of the First International Conference on Data Science, E-Learning and Information Systems*, in DATA ‘18. New York, NY, USA: Association for Computing Machinery, 2018. doi: 10.1145/3279996.3280033

[5] **S. Tongsuksai, S. Mathrani, and K. Weerasinghe**, "Influential characteristics and benefits of cloud ERP adoption in New Zealand SMEs: A vendors' perspective," *IEEE Access*, vol. 11, pp. 23956–23979, 2023. doi: 10.1109/ACCESS.2023.3254500

[6] **S. A. Salim and S. Jaffar**, "A review of cloud-based ERP systems in SMEs," *International Journal of Integrated Engineering*, vol. 12, no. 7, pp. 113–120, 2020. doi: 10.30880/ijie.2020.12.07.013

[7] **E. Hustad, D. H. Olsen, E. H. Jørgensen, and V. U. Sørheller**, "Creating business value from cloud-based ERP systems in small and medium-sized enterprises," in I. O. Pappas, P. Mikalef, Y. K. Dwivedi, L. Jaccheri, J. Krogstie, and M. Mäntymäki, eds, *Digital transformation for a sustainable society in the 21st century*, Cham: Springer International Publishing, 2019, pp. 691–703.

[8] **K. H. Salum and M. Z. Abd Rozan**, "Conceptual model for cloud ERP adoption for SMEs," *Journal of Theoretical and Applied Information Technology*, vol. 95, pp. 743–756, 2017.

[9] **N. M. Alsharari, M. Al-Shboul, and S. Alteneiji**, "Implementation of cloud ERP in the SME: Evidence from UAE," *Journal of Small Business and Enterprise Development*, vol. 27, no. 2, pp. 299–327, 2020. doi: 10.1108/JSBED-01-2019-0007

[10] **S. Gupta and S. C. Misra**, "Moderating effect of compliance, network, and security on the critical success factors in the implementation of cloud ERP," *IEEE Transactions on Cloud Computing*, vol. 4, no. 4, pp. 440–451, 2016. doi: 10.1109/TCC.2016.2617365

[11] **O. Françoise, M. Bourgault, and R. Pellerin**, "ERP implementation through critical success factors' management," *Business Process Management Journal*, vol. 15, no. 3, pp. 371–394, 2009. doi: 10.1108/14637150910960620

[12] **K. Bokovec, T. Damij, and T. Rajkovič**, "Evaluating ERP projects with multi-attribute decision support systems," *Computers in Industry*, vol. 73, pp. 93–104, 2015. https://doi.org/10.1016/j.compind.2015.07.004

[13] **P. Sudhaman and C. Thangavel**, "Efficiency analysis of ERP projects: Software quality perspective," *International Journal of Project Management*, vol. 33, no. 4, pp. 961–970, 2015. https://doi.org/10.1016/j.ijproman.2014.10.011

[14] **O. Velcu**, "Strategic alignment of ERP implementation stages: An empirical investigation," *Information & Management*, vol. 47, no. 3, pp. 158–166, 2010. https://doi.org/10.1016/j.im.2010.01.005

[15] **K. B. Bagshaw**, "NEW PERT and CPM in project management with practical examples," *American Journal of Operations Research*, vol. 11, no. 04, pp. 215–226, 2021. doi: 10.4236/ajor.2021.114013

[16] **J. Haekal and I. Masood**, "Simulation of ERP project scheduling using CPM and PERT method with promodel: A case studies in food and beverage companies in Jakarta Selatan, Indonesia," *AIP Conference Proceedings*, vol. 2530, no. 1, 040003, 2023. doi: 10.1063/5.0122303

[17] **S. Sommer and C. Loch**, "Selectionism and learning in projects with complexity and unforeseeable uncertainty," *Management Science*, vol. 50, pp. 1334–1347, 2004. doi: 10.1287/mnsc.1040.0274

[18] **T. L. Healy**, "Activity subdivision and PERT probability statements," *Operations Research*, vol. 9, no. 3, pp. 341-348, 1961.

[19] **J. G. Cho and B. J. Yum**, "An uncertainty importance measure of activities in PERT networks," *International Journal of Production Research*, vol. 35, no. 10, pp. 2737–2758, 1997. doi: 10.1080/002075497194426

[20] **L. Breiman**, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. doi: 10.1023/A:1010933404324

[21] **J. Haekal, R. Mu'min, Fauzan, A. Hamid, D. Sjarifudin, A. Turseno, Paduloh, and A. Adriansyah**, "Hybrid Feature-Based Critical Success Factors in Cloud Enterprise Resource Planning Through Artificial Neural Networks and Random Forest," *Mathematical Modelling of Engineering Problems*, vol. 12, no. 6, pp. 1911–1924, 2025. doi: 10.18280/mmep.120608

[22] **H. Tyralis, G. Papacharalampous, and A. Langousis**, "A brief review of random forests for water scientists and practitioners and their recent history in water resources," *Water*, vol. 11, no. 5, 2019. doi: 10.3390/w11050910

[23] **I. Lishner and A. Shtub**, "Using an artificial neural network for improving the prediction of project duration," *Mathematics*, vol. 10, no. 22, 2022. doi: 10.3390/math10224189

[24] **A. Sousa *et al.***, "Applying machine learning to estimate the effort and duration of individual tasks in software projects," *IEEE Access*, vol. 11, pp. 89933-89946, 2023. doi: 10.1109/ACCESS.2023.3307310

[25] **M. Vanhoucke**, "The critical path method," in M. Vanhoucke, ed., *Project management with dynamic scheduling: Baseline scheduling, risk analysis and project control*, Berlin, Heidelberg: Springer, 2012, pp. 37–56. doi: 10.1007/978-3-642-25175-7_3

[26] **D. Trietsch and K. R. Baker**, "PERT 21: Fitting PERT/CPM for use in the 21st century," *International Journal of Project Management*, vol. 30, no. 4, pp. 490–502, 2012. https://doi.org/10.1016/j.ijproman.2011.09.004

[27] **R. Feng**, “Improving uncertainty analysis in well log classification by machine learning with a scaling algorithm,” *Journal of Petroleum Science and Engineering*, vol. 196, 107995, 2021. https://doi.org/10.1016/j.petrol.2020.107995
[28] **I. Ali, K. Nguyen, and S. Gupta**, “A multi-disciplinary review of enablers and barriers to Cloud ERP implementation and innovation outcomes,” *Journal of Enterprise Information Management*, vol. 36, pp. 1209–1239, 2023. doi: 10.1108/JEIM-08-2022-0273
[29] **M.-Y. Cheng, Q.-T. Vu, and F. E. Gosal**, “Hybrid deep learning model for accurate cost and schedule estimation in construction projects using sequential and non-sequential data,” *Automation in Construction*, vol. 170, 105904, 2025. https://doi.org/10.1016/j.autcon.2024.105904
[30] **A. S. Mali *et al.***, “Application of artificial intelligence and machine learning in construction project management: A comparative study of predictive models,” *Asian Journal of Civil Engineering*, vol. 26, no. 6, pp. 2671–2686, 2025. doi: 10.1007/s42107-025-01335-6
[31] **S. Khanmohammadi and J. Jassbi**, “A fuzzy approach for risk analysis with application in project management,” in M. F. Azeem, ed., *Fuzzy inference system: Theory and applications*, London: IntechOpen, 2012. doi: 10.5772/37114
[32] **H. Moskowitz and W. I. Bullers**, “Modified PERT versus fractile assessment of subjective probability distributions,” *Organizational Behavior and Human Performance*, vol. 24, no. 2, pp. 167–194, 1979. https://doi.org/10.1016/0030-5073(79)90024-2
[33] **R. R. Britney**, “Bayesian point estimation and the PERT scheduling of stochastic activities,” *Management Science*, vol. 22, no. 9, pp. 938–948, 1976. [Online]. Available: http://www.jstor.org/stable/2629689
[34] **J. B. Wallace**, “Subjective estimation bias and PERT statistical procedures,” *Omega*, vol. 3, no. 1, pp. 79–85, 1975. https://doi.org/10.1016/0305-0483(75)90049-3
[35] **S. Sackey and B.-S. Kim**, “Schedule risk analysis using a proposed modified variance and mean of the original program evaluation and review technique model,” *KSCE Journal of Civil Engineering*, vol. 23, no. 4, pp. 1484–1492, 2019. https://doi.org/10.1007/s12205-019-1826-z
[36] **S. D. Kim, R. Hammond, and J. Bickel**, “Improved mean and variance estimating formulas for PERT analyses,” *IEEE Transactions on Engineering Management*, vol. 61, pp. 362–369, 2014. doi: 10.1109/TEM.2014.2304977
[37] **J. M. Herrerías-Velasco, R. Herrerías-Pleguezuelo, and J. R. van Dorp**, “Revisiting the PERT mean and variance,” *European Journal of Operational Research*, vol. 210, no. 2, pp. 448–451, 2011. https://doi.org/10.1016/j.ejor.2010.08.014
[38] **M. Hajdu and O. Bokor**, “Sensitivity analysis in PERT networks: Does activity duration distribution matter?” *Automation in Construction*, vol. 65, pp. 1–8, 2016. https://doi.org/10.1016/j.autcon.2016.01.003
[39] **M. Schonlau and R. Y. Zou**, “The random forest algorithm for statistical learning,” *The Stata Journal: Promoting Communications on Statistics and Stata,* vol. 20, no. 1, pp. 3–29, 2020. doi: 10.1177/1536867X20909688
[40] **Y. Chen, W. Zheng, W. Li, and Y. Huang**, “Large group activity security risk assessment and risk early warning based on random forest algorithm,” *Pattern Recognition Letters*, vol. 144, pp. 1–5, 2021. https://doi.org/10.1016/j.patrec.2021.01.008
[41] **A. Liaw and M. Wiener**, “Classification and regression by randomForest,” *R News,* vol. 2/3, pp. 18-22, 2002. [Online]. Available: http://www.stat.berkeley.edu/
[42] **D. Marron, A. Bifet, and G. De Francisci Morales**, “Random forests of very fast decision trees on GPU for mining evolving big data streams,” in T. Schaub, G. Friedrich, & B. O’Sullivan (eds.), *ECAI 2014 – 21st European Conference on Artificial Intelligence, Including Prestigious Applications of Intelligent Systems, PAIS 2014, Proceedings* (pp. 615-620). Frontiers in Artificial Intelligence and Applications, vol. 263, IOS Press, 2014. doi: 10.3233/978-1-61499-419-0-615
[43] **R. Genuer, J.-M. Poggi, C. Tuleau-Malot, and N. Villa-Vialaneix**, “Random forests for big data,” *Big Data Research*, vol. 9, pp. 28–46, 2017. https://doi.org/10.1016/j.bdr.2017.07.003
[44] **Q. Liu**, “Research on random forest algorithm based on big data in parallel load forecasting,” in *MATEC Web of Conferences*, EDP Sciences, vol. 228, 01020, 2018. doi: 10.1051/matecconf/201822801020
[45] **K. Itani**, “Mastering construction schedules: The power of CPM and PERT integration,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 10, pp. 868–875, 2023. doi: 10.22214/ijraset.2023.56111
[46] **J. K. Stratman and A. V. Roth**, “Enterprise resource planning (ERP) competence constructs: Two-stage multi-item scale development and validation,” *Decision Sciences*, vol. 33, pp. 601-628, 2002.
[47] **S. Mamoghli, V. Goepp, and V. Botta-Genoulaz**, “Aligning ERP systems with companies’ real needs: an ‘Operational model based’ method,” *Enterprise Information Systems*, vol. 11, no. 2, pp. 185–222, 2017. doi: 10.1080/17517575.2015.1014432

[48] **S. Hoermann, T. Hlavka, M. Schermann, and H. Krcmar**, “Determinants of vendor profitability in two contractual regimes: An empirical analysis of enterprise resource planning projects,” *Journal of Information Technology*, vol. 30, pp. 325-336, 2014. doi: 10.1057/jit.2014.13
[49] **M. Weiss, M. Hoegl, and M. Gibbert**, “How does material resource adequacy affect innovation project performance? A meta-analysis,” *Journal of Product Innovation Management*, vol. 34, no. 6, pp. 842-863, 2017. doi: 10.1111/jpim.12368
[50] **E. W. Ambarsari and H. Herlinda**, “Membangun Pythagoras Sebagai Visualisasi Random Forest Untuk Pemodelan Pohon Keputusan,” *Faktor Exacta*, vol. 13, no. 3, pp. 132–139, 2020. doi: 10.30998/faktorexacta.v13i3.6513
[51] **F. Beck, M. Burch, T. Munz, L. Di Silvestro, and D. Weiskopf**, “Generalized Pythagoras Trees: A Fractal Approach to Hierarchy Visualization,” *Computer Vision, Imaging and Computer Graphics – Theory and Applications*, Communications in Computer and Information Science, vol. 550, pp. 115–135, 2015. doi: 10.1007/978-3-319-25117-2_8