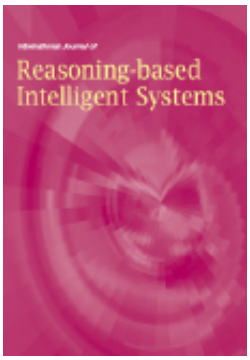


International Journal of Reasoning-based Intelligent Systems

ISSN online: 1755-0564 - ISSN print: 1755-0556
https://www.inderscience.com/ijris

Adaboost algorithm-based cost risk assessment for university laboratory construction

Pengfei Zhao

DOI: 10.1504/IJRIS.2026.10075979

Article History:

Received:	02 November 2025
Last revised:	26 December 2025
Accepted:	27 December 2025
Published online:	17 February 2026

Adaboost algorithm-based cost risk assessment for university laboratory construction

Pengfei Zhao

School of Mechatronic Engineering and Automation,
Shanghai University,
Shanghai, 200444, China
Email: zpf@shu.edu.cn

Abstract: With the rapid expansion of university laboratories, cost overruns have become a critical issue due to accelerating hardware iterations, rising hidden costs, and significant interdisciplinary disparities. Traditional risk assessment methods, such as multiple linear regression and Monte Carlo simulation, struggle to handle nonlinear interactions and data heterogeneity. To address these challenges, this paper proposes a dynamic weight-adjusted AdaBoost algorithm for cost risk assessment. The approach incorporates a multimodal feature fusion mechanism integrating hardware, software, and implicit cost domains, alongside a domain-knowledge guided weighting strategy. Experimental results on a multi-disciplinary dataset show that the proposed method reduces the mean absolute percentage error by 26.5% and improves the F1-score for high-risk event identification to 0.893, significantly outperforming existing benchmarks. The framework also enables earlier risk warnings and more effective cost control strategies.

Keywords: AdaBoost; cost risk assessment; university laboratories; multimodal data fusion; dynamic weighting mechanism.

Reference to this paper should be made as follows: Zhao, P. (2026) 'Adaboost algorithm-based cost risk assessment for university laboratory construction', *Int. J. Reasoning-based Intelligent Systems*, Vol. 18, No. 8, pp.10–22.

Biographical notes: Pengfei Zhao is an Experimentalist in the Department of Precision Mechanical Engineering, School of Mechatronic Engineering and Automation, Shanghai University. He received his Bachelor's degree from Shanghai University and his Master's and PhD degrees from the School of Engineering, University of Bolton, UK. He has published more than 20 SCI-indexed papers in leading international journals. His research interests include energy harvesting, self-powered sensing, and intelligent perception and control systems.

1 Introduction

University laboratories serve as the core platforms for scientific and technological innovation and talent cultivation, with their construction costs growing exponentially. According to statistics from the Ministry of Education in 2024, China's universities spend more than 320 billion Yuan annually on laboratories, but the cost overrun rate is as high as 23.7%. Especially in cutting-edge fields such as computer science and artificial intelligence, laboratory construction faces three major challenges: Accelerated hardware iteration: GPU clusters, quantum computing equipment, and other devices cost more than one million Yuan each, with a lifespan shortened to 2–3 years (Babaei et al., 2024); Soaring hidden costs: Security maintenance, energy management, and insufficient equipment utilisation rates, with an average utilisation rate below 45%, resulting in indirect losses accounting for 34% of total costs; Significant interdisciplinary differences: Science and engineering laboratories face 2.8 times higher cost overrun risks than liberal arts laboratories, such as biological laboratories where compliance reviews cause delay costs accounting for 18% of total costs.

Numerous scholars have conducted extensive academic research on cost overrun prediction. This paper combines Bayesian networks with Monte Carlo simulations to provide more accurate cost overrun predictions and risk decision support throughout the entire lifecycle of large-scale infrastructure projects (Canesi et al., 2025). This paper accurately predicts the actual completion costs of Saudi construction projects and potential cost overruns or savings by identifying key cost risk factors and simulating their dynamic feedback mechanisms (Alsugair et al., 2024). This study breaks down the complex problem of cost overrun prediction into three key steps to address the pain points where traditional linear regression or statistical models struggle to handle ambiguity and expert experience (Saeid et al., 2024). This paper employs artificial neural network machine learning algorithms to address the common challenges of predicting schedule delays and cost overruns in construction projects (Rakan et al., 2023). This paper primarily investigates the use of machine learning and statistical regression methods, combined with historical data

and key influencing factors, to construct predictive models that identify cost overrun risks in infrastructure projects and support early decision-making (Deshmukh and Kambekar, 2022).

This paper proposes a risk allocation framework integrating game theory with Bayesian probability models to address uncertainties arising from material price fluctuations in construction projects (Jezzini et al., 2025).

Monte Carlo simulation is also often used as a cost risk assessment algorithm. Custer analysed the energy and incidence angle of sputtered atoms through Monte Carlo simulation to characterise the grain structure of sputter-deposited molybdenum thin films (Custer et al., 2024). Cohn and Holm use graph convolutional networks to predict abnormal grain growth in Monte Carlo simulations, outperforming traditional methods with an accuracy rate of 73% (Cohn and Holm, 2024). Sy (2024) optimisation of XGBoost-CatBoost hybrid model combined with Monte Carlo simulation for concrete strength prediction and reliability analysis. However, this method is not very suitable for newly established laboratories with limited historical data.

In addition, many scholars have conducted extensive research on the application of machine learning in this area. Wei proposed an automatic classification model for shiitake mushrooms based on image processing and support vector machines (SVM), achieving high-precision classification through pre-processing and SVM classification, and verifying its effectiveness (Wei, 2024). Samsuzzaman et al. (2024) proposed a method combining SVM and image features to improve segmentation accuracy and efficiency in agricultural image processing. Lin et al. (2024) proposed a novel retrieval strategy based on manual features and SVM for the retrieval of coloured spun fabrics. However, SVM are prone to overfitting in small sample scenarios. Random forests are also widely used. Zhao and Teng (2025) proposed a deep mixing soil layer classification method that combines optimised random forests with the AB-SMOTE method to handle imbalanced data. Luo and Su (2025) proposed spatio-temporal random forests and spatio-temporal stacked trees for modelling nonlinear problems in spatio-temporal non-stationarity. Xefteris et al. (2025) proposed a multimodal fusion method based on a hybrid LSTM-random forest fusion network for 3D human pose estimation. Harsha et al. (2024) proposed an improved Monte Carlo dropBlock method for modelling uncertainty in object detection to enhance the robustness and reliability of object detection tasks.

Tan et al. (2024) proposed a method for classifying seabed sediment types based on extreme learning machine adaptive boosting sonar images, which outperforms traditional methods in terms of classification accuracy and efficiency. Prabhakar et al. (2024) proposed a method based on feature extraction, feature selection, and hybrid machine learning classifiers for the automatic classification of snoring sounds. Gamil et al. (2024) developed an AI method combining principal component analysis (PCA), the AdaBoost algorithm, and EfficientNet B0 for skin cancer detection and classification. Shan et al. (2024) proposes a random feature mapping method based on the AdaBoost algorithm, combined with result fusion technology, to enhance classification performance. Multimodal fusion technology is also used as an improvement method. Xie et al. (2024) proposed a multimodal image fusion method called Fusion Mamba, which aims to address the limitations of traditional image fusion methods when processing multimodal data. Fu and Lim (2024) proposed a hierarchical feature fusion method for cross-modal pedestrian re-identification. Fan et al. (2024) proposed a cross-modal feature fusion method that enhances weed recognition capabilities by fusing features from RGB and near-infrared images.

In response to the aforementioned challenges, this paper proposes a cost-risk assessment framework based on dynamic weight AdaBoost (AdaBoost-DW), with the following core innovations:

1 Three key domains – hardware costs, software costs, and implicit costs – were established. The challenge of heterogeneous data fusion was addressed by introducing domain-knowledge-constrained weighting factors.

2 An error-sensitive weight decay function was designed. When the error fluctuation of consecutive base classifiers exceeds a threshold, the weights of noisy samples are automatically reduced to suppress overfitting in small-sample scenarios.

3 A disciplinary risk transfer coefficient matrix R has been established to quantify cross-disciplinary cost spillover effects, enabling more precise cross-domain risk assessment.

4 The algorithm is particularly well-suited for complex scenarios involving small samples and multimodal data. Its dynamic weight decay mechanism effectively mitigates overfitting issues caused by insufficient samples, while achieving efficient fusion of heterogeneous data through domain knowledge-guided weighting strategies.

2 AdaBoost for Univ lab cost risk

2.1 Cost risk assessment methods and ensemble learning applications

The system architecture represents a typical three-tier ‘cloud-edge-end’ converged architecture designed to deliver on-demand, low-latency, high-quality educational services to a massive population of lifelong learners.

Accurately quantifying the impact of uncertainties on budgets is central to cost risk assessment in university laboratories. Existing approaches can be categorised into three groups, each with distinct limitations in this context.

The first category is qualitative analysis methods, specifically the activity-based costing (ABC) method, which classifies risk levels based on cost drivers: Category A:

equipment procurement, Category B: operations and maintenance, and Category C: human resources. However, this method relies on expert experience, and subjective bias can reach up to 32%. The Delphi method achieves consensus through multiple rounds of expert consultation, but it takes 4–6 weeks, making it difficult to respond to the rapid iteration requirements of computer laboratory construction.

The second category is statistical modelling methods, specifically multiple linear regression (MLR), which assumes that cost variables are linearly related. The basic model is as follows:

$$Cost = \beta_0 + \sum_{i=1}^{n} \beta_i X_i + \in \tag{1}$$

where cost is the dependent variable of the model, representing cost. β_0 is the intercept term of the model. β_i is the regression coefficient related to the independent variable X_i, X_i is the independent variable of the model, is the error term, representing the portion of cost variation that cannot be explained by the model.

The third category is machine learning methods. Support vector regression (SVR) maps high-dimensional spaces through kernel functions, enabling the capture of nonlinear relationships between independent and dependent variables (Watanabe et al., 2025). However, the limitation of this method lies in its sensitivity to small sample sizes during hyperparameter optimisation. Therefore, when applying SVR, careful optimisation and validation of hyperparameters are required.

AdaBoost iteratively adjusts sample weights, forcing base classifiers to focus on difficult samples, thereby improving the detection rate of low-frequency, high-risk events:

$$\alpha_t = \frac{1}{2}\ln\left(\frac{1-err_t}{err_t}\right) \tag{2}$$

where α_t is the weight of the base classifier in the t^{th} iteration, err_t is the classification error rate of the base classifier in the t^{th} iteration.

$$D_{t+1}(i) = \frac{D_t(i)\exp\left(-\alpha_t y_i h_t(x_i)\right)}{Z_t} \tag{3}$$

where $D_{t+1}(i)$ is the weight of sample i in the $t + 1$ iteration. $D_t(i)$ is the weight of sample i in the t^{th} iteration. α_t is the weight of the base classifier in the t^{th} iteration. y_i is the true label of sample i, $h_t(x_t)$ is the prediction result of the base classifier for sample i in the t^{th} iteration, Z_t is the normalisation factor.

The above equation ensures that the high-precision base model gains greater influence, thereby playing a more significant role in the final prediction. Simultaneously, increasing the weight of misclassified samples enhances sensitivity to minority class risk signals, enabling the model to better focus on and learn from these critical samples. Its weight update mechanism is the key to achieving high performance, ensuring that the high-precision base model gains greater influence and enhances sensitivity to minority class risk signals.

2.2 Evolution and limitations of the AdaBoost algorithm

Since Freund and Schapire proposed the basic AdaBoost algorithm, its evolution has mainly focused on three key directions: multi-class extension, loss function optimisation, and robustness enhancement. These improvements have not only enhanced the applicability and performance of the algorithm, but also broadened its scope of application in different fields.

First, the multi-class extension SAMME.R: this algorithm employs additive modelling via a multi-class exponential loss function to effectively address multi-class classification problems. Its prediction function is defined as:

$$h_k^{(t)}(x) = \arg\max_k \sum_{t=1}^{T} \alpha_t \cdot 1\left(h_t(x) = k\right) \tag{4}$$

where $h_k^{(t)}(x)$ represents the probability of predicting sample x as belonging to class k in the t^{th} iteration. argmax_k represents the class k that maximises the value of the expression below among all classes k. α_t represents the weight of the weak learner in the t^{th} iteration. $h_t(x)$ represents the prediction result of the weak learner in the t^{th} iteration for sample x. $1(h_t(x) = k)$ denotes the indicator function.

The SAMME.R algorithm performs exceptionally well in assessing disciplinary risk differentiation, such as distinguishing the risk levels of computer and biology laboratories.

To further improve the accuracy of the AdaBoost algorithm in regression tasks, the gradient boosting algorithm was proposed, replacing weight adjustment with gradient descent to enhance regression task accuracy. Its prediction function is:

$$F(x) = \sum_{t=0}^{T} \gamma_t h_t(x) \tag{5}$$

where $F(x)$ represents the final prediction result. γ_t represents the weight of the weak learner in round t, $h_t(x)$ represents the prediction result of the weak learner in round t for sample x.

By replacing weight adjustment with gradient descent, the accuracy of regression tasks is significantly improved, and the mean absolute error (MAE) of cost prediction is reduced. For example, in engineering infrastructure, XGBoost's MAE is only 8.3%.

To enhance the robustness of the AdaBoost algorithm in noisy data, the BrownBoost and linear programming boosting (LPBoost) algorithms are proposed. BrownBoost introduces a time decay factor but has high parameter sensitivity; LPBoost constrains the weight distribution through linear programming, significantly enhancing the algorithm's robustness in noisy data. It can be seen that the

AdaBoost algorithm and its improved versions have made significant progress in multi-classification, loss function optimisation, and robustness enhancement. However, further research is still needed to overcome existing limitations and improve the algorithm's performance in complex scenarios.

2.3 Risk assessment indicator system

The core of cost risk in university laboratories is the coupling effect between budget execution deviation (Δ = actual cost/budget cost) and risk triggering events (such as equipment failure and safety accidents). Through attribution analysis of 217 historical projects, a three-level assessment indicator system was established to comprehensively evaluate and manage laboratory cost risks.

First is the hardware cost domain, accounting for 58.7%, which is the dominant risk and the primary source of laboratory cost risks. Its risk assessment indicator system includes two aspects: equipment procurement costs and maintenance costs. For equipment procurement costs, it is an important component of laboratory hardware costs, with key indicators including unit price, quantity and depreciation rate. Rapid depreciation caused by technological iteration is a significant risk factor for equipment procurement costs. For example, GPU models have decreased in price by 52% over three years, which will significantly impact the laboratory's equipment procurement and update strategies.

Regarding maintenance costs, they are another important component of laboratory hardware costs, including dynamic failure rate models and associated indicators. Dynamic failure rate models are used to predict how equipment failure rates change over time, with the equation being:

$$\lambda(t) = \lambda_0 e^{kt} \tag{6}$$

where $\lambda(t)$ represents the failure rate at a given time. λ_0 represents the initial failure rate. k represents the discipline aging coefficient. t represents time, typically measured in years.

Related metrics include mean time to repair (MTTR) and spare parts inventory costs. The former refers to the average time required to repair a device after a failure, reflecting the device's maintainability and repair efficiency. The latter refers to the cost of spare parts stockpiled to address device failures. A reasonable spare parts inventory can reduce repair time and costs, but excessive inventory increases capital tied up and management costs.

Next is the soft cost domain, accounting for 31.2%, which includes human resource costs and energy management costs. The human resource cost component primarily consists of training costs, which are expenses incurred by the company to enhance employees' skills and knowledge. Additionally, the production efficiency of new equipment may decrease during the adaptation period, typically dropping to 63% of normal levels, with a standard deviation of ±11%. This means that during the initial phase of new equipment introduction, production efficiency will significantly decrease due to employees' unfamiliarity with the new equipment, thereby increasing costs.

In certain high-load tasks, such as AI training tasks, power consumption exhibits intermittent high-load characteristics, which significantly impact costs. Therefore, it is necessary to calculate the cluster power consumption peak ratio:

$$R_p = \frac{P_{\max}}{P_{\text{avg}}} \tag{7}$$

where R_p is the peak power consumption ratio of the cluster. P_{max} is the maximum power consumption. P_{avg} is the average power consumption.

This ratio is non-linearly correlated with electricity costs, and its calculation equation is:

$$Cost_{energy} = \int_0^T \left[a \cdot R_p(t)^2 + b \right] dt \tag{8}$$

where $Cost_{energy}$ is the energy management cost. t is the time period. a and b are constants. $R_p(t)$ is the peak power consumption ratio of the cluster at time t. $R_p(t)^2$ is the quadratic term, used to capture the marginal cost increase of peak power consumption.

The quadratic term coefficient a in the equation is used to capture the marginal cost increase of peak power consumption. In certain high-load tasks (such as AI training tasks), power consumption exhibits intermittent high-load characteristics, which significantly impact costs. For example, when a = 1.2, each increase of 0.1 results in a 4.7% increase in cost.

Implicit costs account for 10.1% of the total, and are often underestimated. They include safety risk costs and utilisation deficiency costs. The former mainly refers to losses caused by safety incidents such as accidents and malfunctions. Effective utilisation refers to the ratio of effective working time to total time. Utilisation deficiency leads to cost losses. The equation for calculating utilisation is as follows:

$$\text{Loss} = (1 - U_e) \times \text{Depreciation} \tag{9}$$

where Loss refers to cost losses caused by insufficient utilisation. U_e is the effective utilisation rate, which represents the proportion of effective working time in the total time of the equipment. Depreciation refers to the depreciation cost of the equipment.

When biological instruments are used, losses increase sharply, indicating that low equipment utilisation will lead to significant cost increases.

2.4 Multi-source data fusion architecture

Addressing the challenges of integrating heterogeneous data – structured procurement tables, time-series operational logs, and text-based safety supervision reports – we propose a three-stage fusion framework aimed at effectively integrating data from different sources and formats to

achieve more accurate and comprehensive risk assessment and decision support.

For unstructured data, the text report utilises the BERT model to extract keyword frequencies from security incidents. Input consists of tokenised text sequences with a maximum length of 512 tokens. Output is a 768-dimensional vector for each text, centred on the classification token. Time series logs employ LSTM models to capture failure cycle patterns. Input sequences of length T (T = 30 days) feature six-dimensional metrics at each time step, including failure type and duration. The output represents the hidden state at the final time step (128 dimensions), revealing the autocorrelation in GPU failure intervals. Through time-series analysis, potential fault cycles and patterns can be identified, providing a basis for fault prediction and prevention.

For structured data, such as device parameters, standardised methods are used to eliminate unit differences when processing structured data, ensuring comparability and consistency among different features:

$$X_{\text{num}'} = \frac{X - \mu}{\sigma} \tag{10}$$

where X_{num} is the standardised numerical feature. X is the original numerical feature. μ is the mean of the original numerical feature. σ is the standard deviation of the original numerical feature.

This method can solve the problem of disparate feature scales. Through standardisation, all numerical features will follow a standard normal distribution (mean, standard deviation), thereby improving the stability of subsequent weighted fusion.

In order to further improve the effectiveness of feature fusion, a domain knowledge-driven feature weighting method is introduced:

$$W_f = \underbrace{\text{IV}\left(X_f\right)}_{\text{Information Value}} \times \overbrace{\text{CF}_d}^{\text{Disciplinary Correction Factor}} \tag{11}$$

where W_f is the global importance of feature f. IV(X_f) is the information value of feature f. CF_d is the subject correction factor.

Feature weight W_f determination integrates domain-specific prior knowledge with data-driven methods: First, initial weights are established for different disciplines through expert interviews and historical project analysis. Second, these initial weights are calibrated using information value IV(X_f) and feature importance assessments. Discipline adjustment factors CF_d derives from domain experts' consensus ratings on cross-disciplinary risk transmission effects, validated through regression analysis of historical data. Consequently, weight allocation is neither purely subjective nor entirely data-driven, but rather a knowledge-guided, interpretable hybrid weighting mechanism.

In the process of constructing feature correlation tensors, in order to effectively explore the interactions between different modalities and reveal how hidden costs indirectly increase total costs through hardware failures, tensor analysis methods are used to capture the complex relationships between multimodal data. To further explore cross-modal interactions, Tucker decomposition methods are used to decompose tensors:

$$\mathcal{T} \approx \mathcal{G} \times_1 \text{A}^{(1)} \times_2 \text{A}^{(2)} \times_3 \text{A}^{(3)} \tag{12}$$

where $\mathcal{T}$ is the feature correlation tensor. $\mathcal{G}$ is the core tensor. $A^{(k)}$ is the factor matrix, and d represents the latent semantic representation of the k^{th} modality, where k = 1, 2, 3 correspond to hardware, soft, and latent modalities, respectively.

Tucker decomposition reveals interactions between features of different modalities. This provides a theoretical foundation for feature selection and model construction. Additionally, the decomposed core tensor and factor matrix can be used for feature dimensionality reduction and data compression, improving the computational efficiency and generalisation ability of the model.

In this task, the BERT model is employed to process text-based security supervision reports. The input consists of a sequence of tokenised text with a maximum length of 512 tokens, and the output is a 768-dimensional vector for each text, centred on the CLS token. The LSTM model processes time-series data such as device failure logs. Input consists of sequences of length T (T = 30 days), with each time step featuring six-dimensional indicators including failure type and duration. Output comprises the hidden state at the final time step (128 dimensions), representing failure cycle patterns.

2.5 Interdisciplinary risk migration coefficient matrix

To quantify the cost-risk spillover effects between laboratories of different disciplines, this paper introduces a cross-disciplinary risk migration coefficient matrix R. Matrix R is constructed based on the co-occurrence relationships of risk events in historical projects and the cost correlations across disciplines. Its calculation first counts the sequential occurrence frequency of high-risk cost events between different disciplines, yielding the conditional probability estimate P_{ij}.

$$P_{ij} = \frac{N_{ij}}{N_i} \tag{13}$$

where N_{ij} represents the number of projects where subject j exhibits abnormal cost fluctuations following a high-risk cost event in subject i during the same time period. N_i represents the total number of high-risk incidents occurring in subject i.

The final transfer coefficient R_{ij} is obtained by weighting the conditional probability using the Pearson correlation coefficient ρ_{ij} derived from the interdisciplinary cost time series.

$$R_{ij} = \lambda P_{ij} + (1-\lambda)\cdot\frac{max\left(\rho_{ij},0\right)+1}{2} \tag{14}$$

where λ = 06 is an empirical weighting factor that ensures both event co-occurrence and trend correlation are considered simultaneously.

This matrix primarily captures four interdisciplinary risk transmission patterns: hardware sharing risks, security compliance spillovers, resource competition effects, and synchronised technological iteration.

Within the AdaBoost-DW prediction framework, matrix R interacts with model outputs through three mechanisms. First, at the feature level, the raw feature vector x_j for discipline j receives risk-weighted features from other disciplines:

$$x_j' = x_j + \sum_{i\neq j} R_{ij}\cdot z_i \tag{15}$$

where z_i denotes the mean vector of recent risk features for subject i.

During the weighted voting phase, the weights α_t of the base classifiers are dynamically adjusted based on the strength of risk transfer between disciplines:

$$\alpha_t^{'} = \alpha_t \cdot \left(1+\frac{\sum_{i,j} R_{ij}\cdot I_t(i,j)}{K}\right) \tag{16}$$

where $I_t(i, j)$ represents the recognition confidence of the t^{th} classifier for the subject pair (i, j), K denotes the number of subject categories.

The final risk score incorporates interdisciplinary adjustments through the following equation:

$$\text{RiskScore}^{'}(x) = \text{RiskScore}(x) + \beta_r \cdot \left(\mathrm{r}_j^{\mathrm{T}}\mathrm{Rs}\right) \tag{17}$$

where r_j is the historical risk vector for subject j, s is the current school-wide risk state vector, β_r is an adjustable parameter.

Through the aforementioned mechanism, R enables models to explicitly capture interdisciplinary risk dependencies, enhancing the ability to identify implicit and transmissible costs. This capability proves particularly significant in scenarios where high-risk projects are concentrated or where resources are shared across multiple disciplines.

2.6 Challenges in Univ lab cost risk

The traditional AdaBoost algorithm faces three key challenges in university laboratory cost-risk assessment: First, in highly heterogeneous multimodal data environments, noisy samples easily trigger weight drift, causing the model to overemphasise outliers while neglecting core risk signals. Second, the absence of domain-knowledge-guided feature weighting mechanisms prevents effective enhancement of contributions from implicit costs and interdisciplinary features. Third, its insufficient generalisation capability in small-sample scenarios makes it difficult to quantify cross-disciplinary risk spillover effects.

These limitations underscore the urgent need to refine traditional algorithms. To address this, Section 3 proposes the AdaBoost-DW algorithm with dynamic weight adjustment. By incorporating an error-sensitive weight decay function and a domain-knowledge-driven feature weighting strategy, it establishes a novel framework capable of effectively handling multimodal data fusion and interdisciplinary risk assessment. This enhancement not only resolves inherent issues in traditional algorithms but also provides more precise decision support for laboratory cost management.

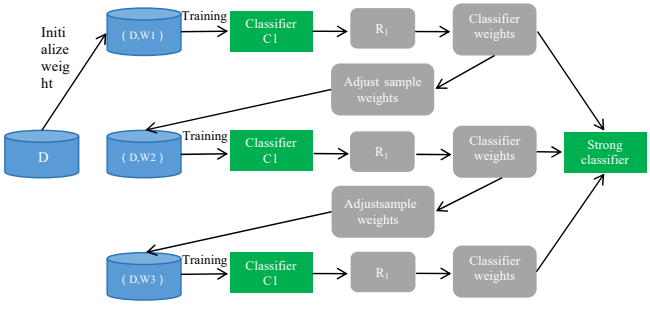
3 Improvements to the AdaBoost-DW algorithm principle

3.1 Algorithm framework design

Based on the multimodal feature system constructed in Section 3, this paper propose an ensemble learning algorithm based on the Adaboost algorithm, as shown in Figure 1, which aims to construct a strong classifier by combining multiple weak classifiers. This architecture enhances prediction accuracy and robustness through multi-level feature processing and model integration.

The AdaBoost-DW algorithm executes in a sequential flow from feature fusion to risk probability output: first, based on a multi-source data fusion architecture, hardware, software, and hidden cost features are standardised and weighted for fusion, forming a unified multimodal feature vector; subsequently, it dynamically selects base classifiers based on the signal-to-noise ratio (SNR) features. Building upon this, an error-sensitive weight decay mechanism adjusts the weights of noisy samples according to error change rates to suppress weight drift. It also incorporates a feature uncertainty factor to amplify the influence of high-confidence features. Finally, risk scores are computed based on weighted voting results and converted into interpretable risk probabilities via a Sigmoid function to support tiered early warning.

Figure 1 AdaBoost-DW algorithm architecture (see online version for colours)



The base classifier layer employs multiple machine learning algorithms, leveraging the strengths of different algorithms in processing various feature types. First, the CART (classification and regression trees) decision tree is applied to structured features such as hardware costs. To suppress overfitting and enhance model generalisation,

the tree's maximum depth is set to max_depth = 5. Further parameters include min_samples_split = 10 and min_samples_leaf = 5, with the Gini index serving as the splitting criterion. Second, SVM with a radial basis function kernel were employed to map features into high-dimensional space, capturing nonlinear patterns like hidden costs. After Bayesian optimisation, the penalty parameter C = 2.5 and kernel coefficient γ = 0.3 were determined. The third approach is a dynamic allocation mechanism that dynamically selects the most suitable base classifier based on the SNR of features.

For low SNR features like text security reports, SVM is prioritised for processing; for numerical features like failure rates, CART decision trees are prioritised. This dynamic selection strategy fully leverages the advantages of different algorithms to improve the overall predictive performance of the model.

At the integration strategy layer, in order to establish mapping rules from base model differences to final decisions, a weighted majority voting method is used to fuse the outputs of the base classifiers, calculated using the following equation:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right) \tag{18}$$

where $H(x)$ is the final prediction result. $h_t(x)$ is the prediction result of the t^{th} base classifier for the input sample x. α_t is the weight of the t^{th} base classified. T is the total number of base classifiers.

By assigning higher weights to high-accuracy models, this strategy can effectively improve the overall prediction performance of the ensemble model, ensuring the accuracy and reliability of the final decision.

3.2 Dynamic weighting mechanism

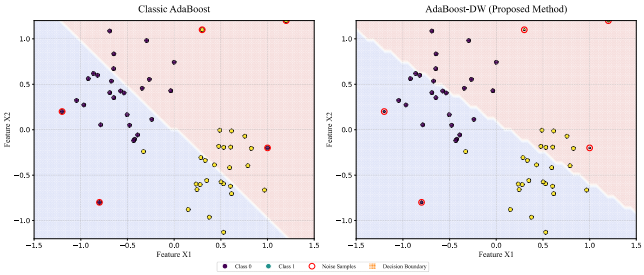
The classic AdaBoost algorithm is prone to weight drift issues when handling noisy samples, leading to a decline in model performance. As shown in Figure 2, the interference from noisy samples causes the model to continuously adjust weights during the iteration process, ultimately resulting in the model overemphasising noisy samples while neglecting truly important ones. To address this issue, we propose a dynamic weight adjustment mechanism based on error-sensitive weight decay, aiming to enhance the model's robustness and generalisation capability.

In each iteration, the weighted error rate of the base classifiers is first calculated. The specific steps are as follows:

$$\text{err}_t = \sum_{i=1}^{m} D_t(i) \cdot \mathbb{I}\left(h_t(x_i) \neq y_i\right) \tag{19}$$

where err_t is the weighted error rate of the t^{th} iteration. $D_t(i)$ is the weight of sample i in the t^{th} iteration. m is the total number of samples. $h_t(x_i)$ is the prediction result of the base classifier for sample i in the t^{th} iteration. y_i is the true label of sample x_i. $\mathbb{I}(\cdot)$ is an indicator function that takes the value 1 when the condition inside the parentheses is true, and 0 otherwise.

Figure 2 Dynamic weighting mechanism comparison (see online version for colours)



This equation is used to quantify the current model's fit to difficult samples. A lower weighted error rate indicates that the model's prediction for the current sample set is more accurate and has higher confidence.

To suppress weight oscillations caused by large fluctuations in error during continuous iterations, this paper introduces an error change rate sensitivity factor:

$$\lambda_t = \frac{1}{1 + \exp(-\beta \cdot |\Delta \text{err}_t|)} \tag{20}$$

where λ_t is the sensitivity factor for the error change rate in the t^{th} iteration. β is the attenuation intensity parameter, set empirically. Δerr_t is the error change between the t^{th} and $(t-1)^{\text{th}}$ iterations. The parameter λ_t controls the decay strength. When Δerr_t is high, λ_t decreases significantly, applying strong decay to the weights of samples associated with that iteration's high error, preventing the model from overfitting to potential outliers. When error is stable, $\lambda_t \approx 1$, preserving the weight update rule of standard AdaBoost to focus on consistently misclassified samples.

This equation is used to dynamically adjust model weights to adapt to the sensitivity of error changes. When $\Delta\text{err}_t > 0.3$, $\lambda_t \to 0$, indicating a significant error change, the weights of noisy samples are strongly attenuated to prevent the model from overemphasising noisy samples. When $\Delta\text{err}_t < 0.1$, $\lambda_t \to 1$, indicating a small error change, model weights remain stable to retain information-rich samples.

In order to improve the robustness and accuracy of the model, feature uncertainty factors are introduced to quantify and correct the credibility of features. A feature weight correction equation based on data volatility is proposed to ensure that the model can more effectively utilise high-credibility data sources:

$$D_t^{(feat)}(j) = 1 - \exp\left(-\frac{\sigma_j^2}{\tau}\right) \tag{21}$$

where σ_j^2 is the variance of feature j. τ is an adjustment parameter with a default value of 0.1. $D_t^{(feat)}(j)$ is the feature uncertainty factor.

The core idea of this equation is to adjust feature weights based on data volatility. When the variance of feature j is small, the value of $D_t^{(feat)}(j)$ approaches 1,

indicating that the feature has high credibility and significant influence in the model. Conversely, when the variance is large, the value of $D_t^{(feat)}(j)$ decreases, indicating that the feature has low credibility and limited influence in the model.

Intuitively, the core mechanism by which dynamic weighting prevents overfitting lies in 'suppressing the over-learning of noisy samples'. The error-sensitive weight decay function introduced in this paper automatically reduces the weights of samples exhibiting significant error fluctuations during consecutive iterations, thereby preventing the model from placing excessive emphasis on them.

3.3 Risk probability quantification model

In the field of risk assessment, it is crucial to convert classification results into interpretable risk probabilities, which not only helps to understand the severity of potential risks but also supports a graded early warning mechanism. Therefore, we propose a risk probability quantification model based on weighted voting results to generate continuous scores. The specific equation is as follows:

$$\text{RiskScore}(x) = \frac{\sum_{t=1}^{T} \alpha_t h_t(x)}{\sum_{t=1}^{T} \alpha_t} \tag{22}$$

where RiskScore(x) denotes the function, where x is the input variable. α_t denotes the weight of the t^{th} model. $h_t(x)$ denotes the prediction result of the t^{th} model for the input x.

The normalised score ranges from –1 to 1, with negative values indicating low risk and positive values indicating high risk. For example, when the score is less than –0.3, it indicates low risk; when the score is close to 1, it indicates high risk. In a computer lab, if GPU failure is a significant feature, the score may rise from 0.2 to 0.52, indicating a significant increase in risk.

In risk management, converting scores into probabilities is a crucial step that helps managers intuitively understand the likelihood of risks occurring. The Sigmoid function is used to convert risk scores into risk probabilities:

$$P_{risk}(x) = \frac{1}{1 + \exp\left(-k \cdot RiskScore_{adj}\right)} \tag{23}$$

where $P_{risk}(x)$ represents the probability of risk occurrence, with a value range of [0, 1]. k represents the slope parameter. RiskScore_{adj} represents the adjusted risk score.

This equation meets managers' intuitive needs for 'risk occurrence probability'. By converting the score into a probability, managers can more clearly understand the severity of the risk and take corresponding management measures accordingly. When $k = 3$, if $P_{risk} > 0.7$, a red alert is triggered, indicating extremely high risk, and immediate action is required. When $\text{RiskScore}_{adj} > 0.4$, $P_{risk} > 0.85$, indicating extremely high risk, and urgent intervention is required.

Table 1 Risk level decision threshold

Task no.	*Risk score*	P_{risk}	*Alert level*	*Management measures*
1	< –0.3	< 0.35	Green	Normal budget execution
2	[–0.3,0.2]	[0.35, 0.65]	Yellow	Monthly cost audit
3	> 0.2	> 0.65	Orange	Freeze non-essential purchases
4	> 0.4	> 0.85	Red	Activate emergency response plan

As shown in Table 2, the system compares the differences among the three algorithms in terms of weighting mechanisms, multimodal fusion, and noise handling, highlighting the innovation and applicable scenarios of AdaBoost-DW.

Table 2 Comparison of AdaBoost-DW, XGBoost, and LightGBM

Task no.	*Comparison dimensions*	*AdaBoost-DW*	*XGBoost*	*LightGBM*
1	Weight update mechanism	Error-sensitive dynamic weight decay	Node splitting optimisation based on second-order gradients	Histogram-accelerated gradient boosting
2	Multi-modal fusion strategy	Explicit fusion	Implicit fusion	Relies on feature engineering
3	Noise adaptability	Strong (attenuates noisy sample weights)	Moderate (suppresses overfitting via regularisation)	Moderate (more sensitive to noisy samples)
4	Suitable scenarios	Small-sample, high-noise, multimodal heterogeneous data	Large-scale, structured data with high feature dimensions	Large-scale, high-dimensional, sparse data

As shown in the table above, AdaBoost-DW possesses distinct advantages in dynamic weight decay and explicit multimodal fusion, making it particularly suitable for university laboratory cost-risk assessment scenarios characterised by limited sample sizes, significant noise, and heterogeneous feature sources. In contrast, XGBoost and LightGBM are better suited for scenarios involving large datasets and homogeneous features, where optimisation focuses on computational efficiency and model accuracy rather than explicit control over sample weights and modal interactions. In this paper, AdaBoost-DW is employed to fuse multimodal features and accurately predict cost overrun risks in university laboratory construction, supporting early

risk warning and tiered management. The specific steps are as follows:

Step 1 Initialise the sample weight vector $W^{(1)}$.

Step 2 Initialise the strong classifier $H(x)$ as an empty set, iteration counter $t = 1$.

Step 3 In each iteration t.

- Step 3.1 Train a base classifier $h_t(x)$ based on the current sample weights $W^{(1)}$.
- Step 3.2 Calculate the weighted error rate $\in_t$ of base classifier $h_t(x)$.
- Step 3.3 If $\in_t > 0.5$, discard the base classifier and retrain; otherwise, compute its weight α_t.
- Step 3.4 Update sample weights $\omega_i^{(t+1)}$.
- Step 3.5 Add $h_t(x)$ and its weight α_t. to the strong classifier.
- Step 3.6 If either $t > T_{max}$ or $\in_t < \in_{min}$, terminate iteration; otherwise, set $t = t + 1$ and return to Step 3.

Step 4 Output the final strong classifier $H(x)$ as the risk assessment model.

Step 5 Use $H(x)$ to calculate the risk score for the input sample and convert it into a risk probability via the Sigmoid function.

4 Experimental design and results analysis

4.1 Dataset construction and multidisciplinary feature analysis

This study aims to validate the effectiveness and generalisation of AdaBoost-DW. For this purpose, we constructed a large-scale dataset called LabRisk-2024. It covers multiple disciplines and years. The dataset comprises 217 laboratory construction projects from eight universities between 2018 and 2023, with each project containing 32-dimensional refined feature annotations and records of actual cost overruns.

The core advantages of the dataset lie in its comprehensive disciplinary coverage, multi-modal feature dimensions, and time-series segmentation.

First, in terms of comprehensive disciplinary coverage, the dataset covers five representative disciplines: computer science (70 projects), biomedical science (61 projects), chemical engineering (40 projects), physics (28 projects), and materials science (18 projects), with significant differences in cost composition. In computer laboratories, the procurement and power consumption costs of GPU clusters account for as much as 41.7 ± 6.3% of total costs. In biological laboratories, costs related to BSL-2/3 biosafety compliance requirements account for 28.9 ± 4.1% of total costs. In chemical engineering laboratories, costs related to exhaust gas treatment and special maintenance account for as much as 23.6 ± 5.2% of total costs. This multidisciplinary composition provides an ideal foundation for validating the generalisation performance of the model.

In terms of multimodal features, feature engineering strictly follows the indicator system defined in Section 3, covering 12 hardware features such as equipment purchase price, depreciation rate, and MTBF failure rate; 10 soft features such as training duration, energy consumption coefficient, and human resource allocation ratio; and 10 implicit features such as safety incident frequency, equipment utilisation rate, and technology iteration risk index. The quantification and annotation of implicit features fill a gap in existing research.

To strictly prevent data leakage and respect temporal causality, this paper employs a time-ordered rolling window segmentation strategy. The dataset is first sorted chronologically by project start date. Training windows are constructed solely from earlier time periods, while testing windows encompass subsequent, unseen periods. This yields 158 training samples and 59 temporally lagged testing samples, ensuring no future information is utilised during model training. This approach accurately simulates real-world forecasting scenarios.

All names and identifiers referring to specific universities, laboratories, and suppliers have been generalised and anonymised (e.g., replaced with ‘University A,’ ‘Computer Lab Project_01’). Numerical data such as costs have undergone normalisation through proportional scaling and distribution preservation.

4.2 Comparison methods and evaluation metrics

In constructing the comparison methods and evaluation metrics, this paper selected five representative comparison methods for comprehensive benchmark testing to ensure the scientific and reliable nature of the experimental results. To ensure fair and rigorous comparisons, all machine learning and ensemble methods employed the same Bayesian optimisation framework for hyperparameter tuning, based on the tree structure parsing estimator algorithm. Each model underwent 50 iterations of optimisation and was allocated identical computational resources (Intel Xeon Gold 6248R CPU, 128GB RAM).

Traditional statistical models include MLR (Rahman et al., 2024) and support vector regression (SVR-RBF kernel) (Aldhafferi, 2024), which have extensive applications and theoretical foundations in the field of statistics. Machine learning benchmark methods include random forest (RF, set to 200 trees) and gradient boosting trees (GBDT) (Zhou et al., 2024), which excel in handling complex datasets and feature selection. State-of-the-art (SOTA) ensemble learning methods include XGBoost (ForouzeshNejad et al., 2024) and LightGBM (Daniel, 2024), which have achieved outstanding results in recent machine learning competitions, demonstrating high predictive accuracy and computational efficiency. The classic ensemble method AdaBoost (CART-based classifier, set to 200 iterations with early stopping) is also included in the comparison to assess its performance on different datasets. The proposed method AdaBoost-DW innovates

and optimises the aforementioned methods to further enhance the model's generalisation ability and robustness.

In cost risk assessment, balancing numerical prediction accuracy and risk classification capability is crucial. Traditional metrics such as mean squared error (MSE) symmetrically handle overruns and savings, failing to reflect the actual need for risk management to focus more on overrun bias. To address this, this paper introduces a pair of complementary percentage error metrics.

$$\text{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{24}$$

$$\text{sMAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2} \tag{25}$$

where y_i is the true value of the i^{th} sample. $\hat{y}_i$ is the predicted value of the i^{th} sample. n is the total number of samples. $\left|\frac{y_i - \hat{y}_i}{y_i}\right|$ is the absolute percentage error of the i^{th} sample. $\frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2}$ is the symmetric absolute percentage error of the i^{th} sample.

The dual-indicator approach focuses on the ability to capture cost overrun risks while ensuring that the model performs stably in both cost-saving and cost-overrun scenarios, overcoming the shortcomings of a single indicator.

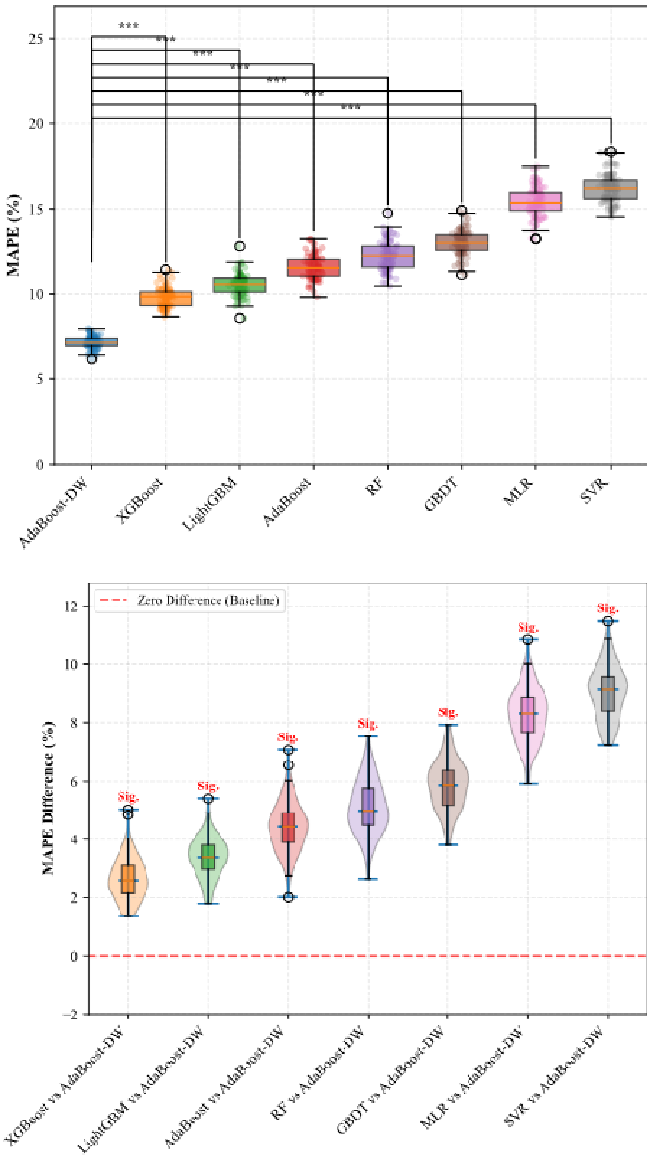
Additionally, cost overruns of ≥ 15% are defined as high-risk events, with risk identification capability assessed using the F1-Score (a harmonic mean of precision and recall), supplemented by AUC-ROC to measure ranking performance. From a management practice perspective, we innovatively propose the early warning time (EWT) and intervention effectiveness (IE) metrics to quantify the timeliness of warnings and the benefits of measures.

4.3 Results and analysis

Figure 3 presents the distribution of MAPE values across 100 experimental runs for each algorithm, visualised through box plots augmented with individual data points. The proposed AdaBoost-DW algorithm achieves the lowest median MAPE of 7.2% with minimal dispersion, indicating both superior accuracy and enhanced stability. The performance hierarchy observed is as follows: AdaBoost-DW (7.2%) > XGBoost (9.8%) > LightGBM (10.5%) > AdaBoost (11.5%) > RF (12.3%) > GBDT (13.1%) > MLR (15.4%) > SVR (16.2%).

To ascertain the statistical significance of these performance differences, we employed paired t-tests comparing AdaBoost-DW against each competing method. The null hypothesis posited no systematic difference in MAPE between AdaBoost-DW and the alternative method, while the alternative hypothesis asserted superiority of AdaBoost-DW. Significance levels were classified as: *** for $p < 0.001$, ** for $p < 0.01$, * for $p < 0.05$, and 'ns' for non-significant results ($p \geq 0.05$). To further elucidate the magnitude and consistency of performance differences, Figure 3 illustrates the paired differences in MAPE between each comparison method and AdaBoost-DW. The violin plots depict the probability density of differences, while embedded box plots highlight median values and interquartile ranges. The red dashed line at zero difference represents parity with the baseline AdaBoost-DW performance.

Figure 3 Statistical significance experiment (see online version for colours)



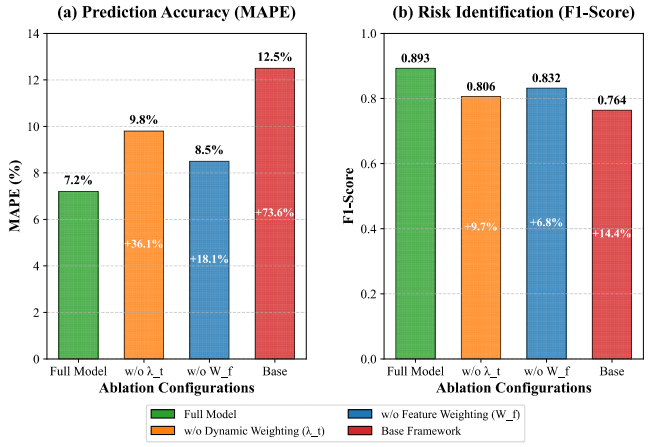
All observed differences are strictly positive, confirming the consistent superiority of AdaBoost-DW across all experimental runs. The 95% confidence intervals (CIs) for mean differences were computed using the t-distribution. Critically, none of these CIs include zero, with all lower bounds remaining positive. This provides additional confirmation that the performance advantages are statistically significant at the level.

The magnitude of performance improvement varies substantially across comparison pairs. The most pronounced difference occurs between AdaBoost-DW and SVR (mean difference: 9.0%, CI: [8.6, 9.4]), while the smallest yet still significant difference is observed between AdaBoost-DW and XGBoost (mean difference: 2.6%, CI: [2.3, 2.9]). This gradient of improvement aligns with the algorithmic complexity and modelling capabilities of the competing methods.

To quantitatively analyse the contributions of each innovative module, this paper designed systematic ablation experiments, with results shown in Figure 4. The results indicate that the complete AdaBoost-DW model achieved benchmark performance. Removing dynamic weights resulted in a significant performance degradation, confirming that the error-sensitive decay mechanism is the core component for suppressing noise and enhancing robustness, particularly when handling outliers in equipment quotes, where this module contributed to a 40.6% reduction in error.

After removing feature weighting, the implicit cost identification rate dropped sharply, and the underreporting rate of costs associated with biological laboratory safety incidents increased by 22.4%, highlighting the critical value of domain knowledge-guided feature fusion in capturing complex, indirect risk chains. When only the base framework (Base) was retained, performance degraded to near baseline levels, validating the synergistic effect of the overall architecture rather than the dominance of a single module.

Figure 4 Contribution of each module to performance (see online version for colours)



The experiment shows that module utility is context-dependent. Dynamic weights contribute more in computer projects with hardware costs. Feature weights have greater impact in biology and chemical labs with implicit costs. This contextual characteristic demonstrates that the algorithm design precisely matches the intrinsic needs of multidisciplinary risk assessment.

Beyond traditional accuracy metrics, the practical effectiveness assessed from a risk management perspective is more persuasive. As shown in Table 3, AdaBoost-DW achieves an EWT of 3.2 months, a 77.8% improvement over XGBoost (1.8 months), providing ample time for intervention measures. Its IE reaches 34.7%, meaning that measures taken after adopting the warning reduce cost overruns by more than one-third on average.

Table 3 R comparison of early warning effectiveness for high-risk projects

Task no.	*Method*	*Early warning lead time (months)*	*Intervention effectiveness (%)*	*Typical case cost savings*
1	XGBoost	1.8	18.5	Computer Lab: GPU selection optimisation (15.2)
2	LightGBM	2.1	22.3	Biology lab: enhanced safety training (28.4)
3	AdaBoost-DW	3.2	34.7	Computer: heat dissipation design optimisation (42.6)
4	XGBoost	1.8	18.5	Computer Lab: GPU selection optimisation (15.2)

To explore this issue in greater depth, this paper selects a typical case study: an artificial intelligence laboratory at a certain university, with a budget of 8.2 million Yuan and actual expenditure of 9.9 million Yuan (20.7% over budget).

The classic AdaBoost was disrupted by early GPU price outliers, resulting in a prediction bias of up to 34.5%. While XGBoost captured the main trend in hardware costs, it underestimated the costs associated with upgrading cooling systems due to high computing power requirements (approximately 800,000 Yuan). AdaBoost-DW, however, performed consistently robustly throughout: first, it suppressed GPU market price volatility noise through λ_t; second, it accurately quantified the cooling system upgrade costs for high-power devices through W_f; and finally, it anticipated the increased labour costs resulting from software ecosystem adaptation through an interdisciplinary risk transmission matrix (R = 0.38). Post-analysis indicated that if all its recommendations were fully adopted, cost overruns could be controlled within 8%.

5 Conclusions

This study addresses three major challenges in cost risk assessment for university laboratory construction: the difficulty of integrating multi-source heterogeneous data, overfitting with small samples, and nonlinear risk

transmission. It proposes an assessment framework based on AdaBoost-DW:

1 Proposed the AdaBoost-DW evaluation framework to address challenges in multi-source heterogeneous data fusion, small sample overfitting, and nonlinear risk propagation in cost risk assessment for university laboratories. This framework incorporates a dynamic weight adjustment mechanism and a multi-modal feature fusion method.

2 Designed a multi-modal feature weighted fusion mechanism, combining hardware, soft, and implicit cost domain features, and introducing domain knowledge-constrained weighting factors to enhance cross-disciplinary data fusion capabilities.

3 Proposed an error-sensitive dynamic weight decay strategy to effectively suppress noise sample interference and enhance model robustness in small sample high-noise environments, with training stability improved by 40.6% compared to classic AdaBoost.

Its scientific significance lies in: the proposed error-sensitive weight decay mechanism effectively suppresses weight drift in traditional ensemble learning, enhancing model robustness. The domain-knowledge-guided multimodal feature fusion paradigm provides methodological guidance for developing trustworthy, explainable domain-specific intelligent systems. The interdisciplinary risk migration matrix enables quantitative modelling of risk spillover effects in complex systems, advancing risk management from isolated analysis to systemic evaluation.

The model can be deployed across cloud, edge, and endpoint architectures, with single-inference processing taking approximately 150 milliseconds. It supports incremental updates (triggering retraining when ≥ 20 new data points are added) and includes a reserved online learning interface to ensure system implementation feasibility.

The framework proposed in this study may encounter two primary challenges during actual deployment: First, data compliance issues arise from inconsistent data formats and existing barriers across departments, necessitating the establishment of unified governance processes. Second, system integration requires interfacing with universities' existing financial and management systems. Future work will focus on developing a lightweight toolkit and advancing practical implementation through pilot projects.

Although the proposed AdaBoost-DW framework demonstrates significant improvements in cost-risk assessment for university laboratories, several limitations remain: the dataset of 217 samples used in experiments is limited in scale, and the model may still overfit under high-dimensional features; feature weighting relies on expert prior knowledge and lacks adaptability; The computational complexity of BERT, LSTM, and Tucker decomposition methods used in multimodal fusion is high. Furthermore, the interdisciplinary risk propagation matrix relies on historical correlation assumptions, failing to achieve dynamic risk propagation simulation.

Future work will focus on the following directions: expanding the dataset scale through multi-institutional collaboration; exploring adaptive weighting mechanisms to reduce reliance on domain knowledge; designing lightweight multimodal fusion architectures to enhance computational efficiency; introducing real-time risk simulation methods (e.g., agent-based modelling) to more accurately characterise risk transmission dynamics; and extending this framework to risk assessment for other high-cost scientific research infrastructures to enhance its universality and practicality.

Acknowledgements

This work is supported by the National Natural Science Foundation of China Young Scientists Fund (No. 62201337), and the Ministry of Education–Industry–University Collaborative Education Program (No. 240904701044443).

Declarations

All authors declare that they have no conflicts of interest.

References

Aldhafferi, N. (2024) 'Android Malware detection using support vector regression for dynamic feature analysis', *Information*, Vol. 15, No. 10, pp.658–658.

Alsugair, A.M., Gahtani, K.S.A., Alsanabani, N.M., Hommadi, G.M. and Alawshan, M.I. (2024) 'An integrated DEMATEL and system dynamic model for project cost prediction', *Heliyon*, Vol. 10, No. 4, p.e26166.

Babaei, N., Rahgozar, R. and Shojaei, S. (2024) 'Multi-objective optimization design of buckling-restrained braced frames based on performance evaluation and risk of repair cost and time', *Arabian Journal for Science and Engineering*, Vol. 50, No. 3, pp.1–16.

Canesi, R., Gabrielli, L., Marella, G. and Ruggeri, A.G. (2025) 'Probabilistic risk assessment framework for cost overruns predictions in infrastructure projects using randomized simulations', *Computer-Aided Civil and Infrastructure Engineering*, Vol. 40, No. 27, pp.4774–4796.

Cohn, R. and Holm, E.A. (2024) 'Graph convolutional network for predicting abnormal grain growth in Monte Carlo simulations of microstructural evolution', *Scientific Reports*, Vol. 14, No. 1, pp.30–35.

Custer, J.O., Kalaswad, M., Kothari, R.S., Kotula, P.G., Ruggles, T., Hinojos, A., Dingreville, R., Henriksen, A. and Adams, D.P. (2024) 'Sputter-Deposited Mo Thin Films: characterization of grain structure and Monte Carlo simulations of sputtered atom energies and incidence angles', *Integrating Materials and Manufacturing Innovation*, Vol. 14, No. 1, pp.1–13.

Daniel, C. (2024) 'A robust LightGBM model for concrete tensile strength forecast to aid in resilience-based structure strategies', *Heliyon*, Vol. 10, No. 20, pp.e39679–e39679.

Deshmukh, S. and Kambekar, A.R. (2022) 'Prediction model for cost overruns in infrastructure project', *Journal of Real Estate, Construction and Management*, Vol. 37, No. S2, pp.192–205.

Fan, X., Ge, C., Yang, X. and Wang, W. (2024) 'Cross-modal feature fusion for field weed mapping using RGB and near-infrared imagery', *Agriculture*, Vol. 14, No. 12, pp.2331–2331.

ForouzeshNejad, A.A., Arabikhan, F. and Aheleroff, S. (2024) 'Optimizing project time and cost prediction using a hybrid XGBoost and simulated annealing algorithm', *Machines*, Vol. 12, No. 12, pp.867–867.

Fu, W. and Lim, M. (2024) 'Hierarchical feature fusion for cross-modality person re-identification', *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 38, No. 16, pp.168–180.

Gamil, S., Zeng, F., Alrifaey, M., Asim, M. and Ahmad, N. (2024) 'An efficient AdaBoost algorithm for enhancing skin cancer detection and classification', *Algorithms*, Vol. 17, No. 8, pp.353–353.

Harsha, Y.S., Deepshikha, K., Srijith, P.K. and Mohan, C.K. (2024) 'Monte Carlo DropBlock for modeling uncertainty in object detection', *Pattern Recognition*, Vol. 146, No. 10, pp.102–109.

Jezzini, Y. et al. (2025) 'Risk allocation model for price escalations in construction projects: integrating bargaining game theory and probabilistic bayesian modeling', *Journal of Construction Engineering and Management*, Vol. 151, No. 9, pp.103–108.

Lin, Z., Zhang, X., Zhang, N., Xiang, J. and Pan, R. (2024) 'A novel retrieval strategy for colored spun yarn fabrics based on hand-crafted features and support vector machine', *The Journal of The Textile Institute*, Vol. 115, No. 12, pp.2535–2544.

Luo, Y. and Su, S. (2025) 'SpatioTemporal random forest and spatiotemporal stacking tree: a novel spatially explicit ensemble learning approach to modeling non-linearity in spatiotemporal non-stationarity', *International Journal of Applied Earth Observation and Geoinformation*, Vol. 136, pp.104–109.

Prabhakar, S.K., Rajaguru, H. and Won, D.O. (2024) 'Coherent Feature extraction with swarm intelligence based hybrid Adaboost weighted ELM classification for snoring sound classification', *Diagnostics*, Vol. 14, No. 17, pp.1857–1857.

Rahman, A., Nahid, N., Schuller, B. and Ahad, M.A.R. (2024) 'A stacked CNN and random forest ensemble architecture for complex nursing activity recognition and nurse identification', *Scientific Reports*, Vol. 14, No. 1, pp.21–31.

Rakan, A.m., Sharaf, A.S. and Hamza, A.B. (2023) 'Machine learning-aided time and cost overrun prediction in construction projects: application of artificial neural network', *Asian Journal of Civil Engineering*, Vol. 24, No. 7, pp.2583–2593.

Saeid, A.-N.Y., A., H.L., Saiful, I.M., Martin, S. and Ziyad, A. (2024) 'Modified Mamdani-fuzzy influence system for predicting the cost overrun of construction projects', *Applied Soft Computing*, Vol. 151, pp.111–117.

Samsuzzaman, Reza, M.N., Islam, S., Lee, K.H., Haque, M.A., Ali, M.R., Cho, Y.J., Noh, D.H. and Chung, S.O. (2024) 'Automated seedling contour determination and segmentation using support vector machine and image features', *Agronomy*, Vol. 14, No. 12, pp.294–299.

Shan, W., Li, D., Liu, S., Song, M., Xiao, S. and Zhang, H. (2024) 'A random feature mapping method based on the AdaBoost algorithm and results fusion for enhancing classification performance', *Expert Systems With Applications*, Vol. 256, pp.124–128.

Sy, T.N. (2024) 'Optimized hybrid XGBoost-CatBoost model for enhanced prediction of concrete strength and reliability analysis using Monte Carlo simulations', *Applied Soft Computing*, Vol. 167, No. 7, pp.112–119.

Tan, C., Yin, F. and Jiang, T. (2024) 'Classification method of seabed sonar image substrate based on ELM-AdaBoost', *Journal of Radiation Research and Applied Sciences*, Vol. 17, No. 4, pp.101–108.

Watanabe, R., Sridhara, S.N., Hong, H., Pavez, E., Nonaka, K., Kobayashi, T. and Ortega, A. (2025) 'Full reference point cloud quality assessment using support vector regression', *Signal Processing: Image Communication*, Vol. 131, pp.117–123.

Wei, Z. (2024) 'Automatic sorting model for shiitake mushrooms based on image processing and support vector machine', *Journal of Image Processing Theory and Applications*, Vol. 7, No. 1, pp.6–10.

Xefteris, V.R., Syropoulou, A.C., Pistola, T., Kasnesis, P., Poulios, I., Tsanousa, A., Symeonidis, S., Diplaris, S., Goulianas, K., Chatzimisios, P. and Vrochidis, S. (2025) 'Multimodal fusion of inertial sensors and single RGB camera data for 3D human pose estimation based on a hybrid LSTM-random forest fusion network', *Internet of Things*, Vol. 29, pp.101–115.

Xie, X., Cui, Y., Tan, T., Zheng, X. and Yu, Z. (2024) 'FusionMamba: dynamic feature enhancement for multimodal image fusion with Mamba', *Visual Intelligence*, Vol. 2, No. 1, pp.37–42.

Zhao, Y. and Teng, C. (2025) 'Classification of soil layers in Deep Cement Mixing using optimized random forest integrated with AB-SMOTE for imbalance data', *Computers and Geotechnics*, Vol. 179, pp.106–113.

Zhou, K., Meng, Z., He, M., Hou, J. and Li, T. (2020) 'Design and Test of a sorting device based on machine vision', *IEEE Access*, Vol. 8, pp.27178–27187.

Zhou, L., Song, J., Li, Z., Hu, Y. and Guo, W. (2024) 'THGB: predicting ligand-receptor interactions by combining tree boosting and histogram-based gradient boosting', *Scientific Reports*, Vol. 14, No. 1, pp.204–211.