International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

Predictive Risk Analytics in Project Management Using Graph-Based Lightweight AI and Counterfactual Risk Mitigation

S. Balaji , N. Poyyamozhi
Department of Computer Science and Engineering, St. Peter's Institute of Higher Education and Research, Avadi, Chennai, India

Abstract- — Currently, the field of project management faces increasing uncertainty as projects must deal with changing requirements, resource shortages, and the unpredictable effects of human actions, technical systems, and external events. However, existing data-driven models have failed to provide interpretable results, preventing project managers from identifying the factors that create risks. Thus, this research presents a lightweight and explainable data-driven decision support system that enables project risk prediction and risk management in complex project management environments. The devised methodology employs a Project Management Risk Dataset, which includes project demographics and operational metrics, human factors, organizational context, technical aspects, and external influences. Moreover, a comprehensive data reliability testing is conducted through pre-processing methods for categorical attributes, one-hot encoding, and Min-Max normalization of budget and timeline, and risk metrics. Advanced feature engineering uses graph-based feature relationships to identify hidden project attribute dependencies, Graph Signal Processing to create project attribute dependencies, and LASSO with polynomial feature expansion to achieve optimal results. The proposed TAM-Lite architecture integrates TabNet, a mini autoencoder, and a shallow multilayer for project risk prediction. Moreover, stage-wise training is conducted based on Gradient Boosted Rule Sets with Extreme Learning Machines and fuzzy logic classification. The model generates risk level probabilities, which are evaluated through Bayesian Networks and counterfactual explanations to deliver clear and actionable risk reduction recommendations.

Keywords: Project Risk Prediction, Explainable Decision Support System, TAM-Lite Architecture, Graph-Based Feature Engineering, Counterfactual Risk Mitigation.

I. INTRODUCTION

Modern society depends on software systems that automate personal and industrial processes. Software systems need to function with dependable performance and operational efficiency while users depend on their services [1]. Software development projects maintain their inherent complexity, which software development professionals need to manage through development processes that face both resource limitations and user needs, which keep changing while they work on projects. Testing of software development proceeds through multiple stages which these exact factors impact. The need to identify and manage all potential software failure risks has gained major significance in recent times [2] [3]. The organization faces two types of risks which include both technical risks and risks that impact all parts of their organizational framework from their management systems to their interaction with outside environments. Proactive risk identification and mitigation in software development processes lead to lower rework costs and project delays while enabling software systems to achieve their operational performance targets [1]. Modern software development relies on failure assessment and management as fundamental components of its development process. The project activities aim to detect possible risks which they will handle through developed strategies that will minimize the detected risks [4].

The success of a project becomes more likely when teams use effective systems to assess failures because those systems enable teams to make decisions that prevent problems. Organizations must identify possible failure points before they can develop suitable risk mitigation plans which must include process improvements and testing betterment and enhanced stakeholder communication and essential resource allocation [5] [6]. The software industry lacks sufficient tools which enable decision-makers to evaluate and control project risks although they have developed better understanding of software risk management. Existing methods operate through single case study validation, which uses particular risk metrics, which makes them unsuitable for different project environments. The software industry needs to develop standardized failure assessment tools which can evaluate any software project throughout all phases of its Software Development Life Cycle [1]. The proposed model establishes a complete solution which outmatches all current failure assessment tools because it provides a complete software project risk management framework which utilizes data-driven methods to show its operational procedures. The system identifies potential failures throughout the software development life cycle while delivering detailed risk reduction suggestions which help users make choices that result in project success [7].

This study presents a lightweight and explainable data-driven decision support system for effective project risk prediction and management in complex project environments. Initially, the input data collected from the Project Management Risk Dataset is pre-processed, and feature engineering is done with a TAM-Lite architecture. Here, the TAM-Lite architecture is developed by combining TabNet, a mini autoencoder, and a shallow multilayer perceptron. The method enables interpretable risk prediction through Gradient Boosted Rule Sets and Extreme Learning Machines, and fuzzy logic is used for stage-wise training. Moreover, the Bayesian Networks and counterfactual explanations analyse the resulting risk probabilities to create specific risk mitigation recommendations.

The major contribution of this research is given below.

The work introduces an efficient decision support system that uses data to explain its decision-making process while it successfully predicts project risks throughout complex project environments.

The research uses graph-based feature relationships with Graph Signal Processing and LASSO-based polynomial feature selection to discover hidden connections between project attributes that enhance risk prediction results.

The study combines a TAM-Lite predictive system with Bayesian Networks and counterfactual explanations to provide clear risk evaluations and practical solutions for successful project risk management.

The remainder of this paper is structured as follows. The related literature is reviewed in Section 2. Section 3 outlines the development of the project risk assessment model. The implementation process, with the description of the dataset, the experimental setup, and the results are explained in Section 4. Section 6 presents research directions for future studies.

II. LITERATURE REVIEW

The existing literature presents several approaches for project risk assessment and mitigation using statistical models, machine learning, and explainable artificial intelligence techniques. ForouzeshNejad et al. (2025) proposed a data-driven framework for evaluating agile project performance through the combination of schedule data, cost information, and productivity metrics. The project management research demonstrated that data-driven insights improve decision-making results. However, the framework used conventional analytical models without including explainable AI methods or dynamic project environment risk management systems. Xu et al. (2024) developed a hybrid risk prediction model that combines DL techniques with attention mechanisms and neural network systems to analyse the interaction of project characteristics with each other in non-linear ways. The experiments demonstrated that the system accurately predicted both project delays and cost overruns with better precision. The system operated as a black box that prevented users from understanding its functions, which made it difficult for managers to use it as a decision-making tool.

Kula et al. (2023) presented a resource allocation and risk assessment model, which used fuzzy logic to manage uncertain elements in software project planning. The model used linguistic rules with membership functions to create decision-making options which functioned in uncertain situations. The approach required expert-defined rules for its functioning because it did not have historical project data and learning-based optimization methods which could enhance its prediction capabilities. Bauskar et al. (2024) developed an explainable AI framework that uses SHAP to identify primary risk factors which impact software project risk assessment processes. The research established feature-based explanations that improved transparency and increased trust among stakeholders, yet the study concentrated on interpretability without implementing predictive optimization and automated decision support systems, which would have enabled proactive risk management. A summary of the major conventional approaches is presented in Table 1.

Table 1. Literature review of Conventional approaches related to project risk prediction and mitigation

References	**Methodology**	**Merits**	**Demerits**
Forouzesh Nejad et al. (2025) [8]	Explainable AI–Based Agile Project Risk Prediction Model	Utilized historical project data to support objective and evidence-based	struggled to adapt to rapidly changing project environments.

		decision-making.	
Xu et al. (2024) [9]	Machine Learning–Driven Software Risk Prediction Framework	Provides high predictive accuracy for project risk and failure prediction.	Prone to overfitting while processing small or imbalanced datasets.
Kula et al. (2023) [10]	Bayesian Delay Prediction Model for Software Projects	Effectively handled uncertainty and ambiguity using linguistic rules.	Attained low scalability when the number of rules increased.
Bauskar et al. (2024) [11]	Predictive Analytics Framework for Project Risk Management	Improved trust and acceptance among project managers and stakeholders.	Explanation quality depends on the underlying predictive systems.

III. PROPOSED METHODOLOGY

Recently, the project management field has experienced growing challenges since project needs keep changing, while projects deal with resource limitations, human factors, technical difficulties, and outside disturbances. The classical project risk assessment models failed to identify the intricate connections between project elements. Hence, the system requires development as a lightweight solution that uses data to create transparent decision support for predicting project risks. This research presents a lightweight decision support framework for enabling better project risk management practices. Initially, the complete project management information, including demographic, operational, human and organizational, and technical and external factors are gathered. The pre-processing stage establishes data accuracy through its comprehensive procedures, which include using median and mode value imputation to treat missing data and converting categorical data through one-hot encoding and Min-Max normalization for budget, timeline, and risk measurement. The advanced feature engineering process creates graph-based models that show project attribute connections, while Graph Signal Processing is used to identify hidden project element connections. Moreover, the polynomial feature expansion and LASSO selection techniques are integrated to keep the best predictive attributes. The development of domain-specific risk indices improves the effectiveness of risk assessment methods. Furthermore, the TAM-Lite architectural framework enables project risk prediction through its combination of TabNet and a mini autoencoder and a shallow multilayer perceptron, which undergoes stage-wise training using Gradient Boosted Rule Sets and Extreme Learning Machines and fuzzy logic classification to determine risk levels. Finally, the risk probability results are investigated through Bayesian Networks, which create causal relationship models, while explainable decision support uses counterfactual analysis to identify necessary changes that will decrease project risk and assist project managers in making informed decisions that adapt to changing project conditions. Figure 1 shows the block diagram of the proposed risk prediction and mitigation system.

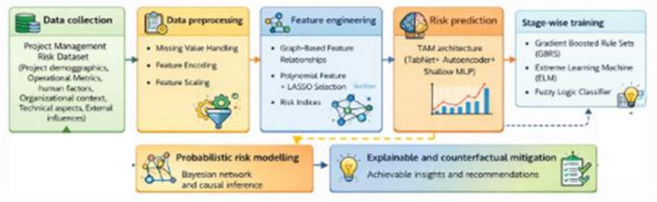


Figure 1. General structure of the proposed Methodology

3.1 Data Collection

The Project Management Risk Dataset [12] functions as an extensive multi-dimensional dataset that enables precise risk prediction with informed decision-making processes within software and project management settings. The system combines multiple elements that determine whether a project will succeed or fail, which include project demographics (project type, budget, timeline, team size, and project complexity) with operational metrics (change request frequency, budget utilization, resource availability, and phase duration) and human factors (team and project manager experience, stakeholder engagement, and team turnover).

The dataset includes different organizational factors, which include process maturity and regulatory compliance, and funding source and risk management practices, with technical aspects that include technology familiarity, technical debt, integration complexity, and system stability. The system includes external factors, which include market volatility, industry uncertainty, external dependencies, and client experience, to create a model that accurately reproduces actual project environments. The combined attributes create an all-encompassing model that describes project settings and supports effective data-based risk evaluation with transparent project risk management.

3.2 Data Pre-processing

Data pre-processing is the process of cleaning, transforming, and organizing raw data into a suitable and consistent format to enable effective analysis and modelling. The process requires handling missing or noisy values while encoding categorical variables and scaling numerical features to enhance data quality. The purpose of data pre-processing in predictive modelling is to ensure that input data remains reliable and comparable while eliminating all data inconsistencies, which results in improved model accuracy and stability, and model interpretability. A list of essential data pre-processing techniques is provided below.

3.2.1 Missing Value Handling

Missing value handling [12] is a crucial step in data pre-processing that addresses incomplete or absent entries in a dataset to ensure data consistency and reliable model performance. Missing values can occur due to data entry errors or system limitations, and untreated missing values will create prediction biases while decreasing model accuracy. Here, the numerical and categorical features are handled by median and mode imputation techniques, and this is explained below.

a) Median imputation

The method replaces missing numeric data by using the median value derived from existing data within the feature [12]. The median value functions as the central point of a data distribution; it maintains stability against extreme data points. The median imputation method displays three patterns that show budget and timeline and utilization rate distribution through its measurement. The process maintains complete data distribution, which enables risk prediction models to maintain predictive accuracy.

b) Mode imputation

Mode imputation [13] replaces missing categorical values with the most frequently occurring category in that feature. The method uses the most common category as the typical representation of project type, funding source, and organizational maturity level attributes. The mode imputation method maintains existing categorical patterns while stopping the emergence of nonexisting categories which would interfere with the educational process.

3.2.2 Feature Encoding

The process of feature encoding transforms categorical data into numeric values which machine learning models can use [14]. The system requires categorical attributes to be encoded through a method that maintains their original meaning while preventing the creation of extra relationships between their values. The research uses one-hot encoding to convert Project_Type and Funding_Source elements from its categorical data. One-hot encoding splits each categorical attribute into multiple binary (0 or 1) indicator variables. The approach prevents the creation of false ordinal or numeric relationships between categories which serves as a critical requirement for accurate risk assessment. The expression of one-hot encoding is given as in Eqn. (1)

$$\underset{\{w_m^{(s)}\}}{\arg\min} \sum_{m=1}^{M-1} A_{m,m+1} \sum_{n=1}^{N} w_m^{(s)} w_{m+1}^{(s)} \tag{1}$$

wherew_m^((s))implies a binary variable, Mdenotes total features, A_mrepresents the interaction between two features. One-hot encoding enables machines to understand categorical data, which leads to better model interpretability and increased project risk prediction accuracy and fairness.

3.2.3 Feature Scaling

The feature scaling process transforms numerical features to a common scale while maintaining their original value differences. The process of feature scaling standardizes project budget and duration and risk metric values, which enables balanced learning and accelerates model convergence while improving prediction accuracy. The process of feature scaling applies Min-Max normalization [15] to transform numerical variables into a standardized range, which includes budget, timeline, and risk-related metrics. The expression of min-max normalization is given in Eqn. (2)

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{2}$$

Where X_scaleddenotes normalized result, X implies the original value of the input feature, X_max,X_minindicates the minimum and maximum value of the input feature. This transformation maintains the original distances between data points while it enhances numerical stability, which results in quicker and more dependable model training. The pre-processing process establishes data reliability while maintaining equal feature importance, which creates a stable base for precise and understandable project risk prediction.

3.3 Feature Engineering

The process of feature engineering develops new data features and transforms existing features to enhance model performance and boost prediction accuracy. The project risk prediction method creates significant variables through its graph-based relationships, which reveal hidden dependencies. The method uses Graph Signal Processing to model hidden interactions, while it employs polynomial feature expansion with LASSO for selecting important features, and it generates risk indices that include Operational, Technical, and Human Factor Scores. The model successfully detects risk patterns while providing an accurate assessment of project risk.

3.3.1 Graph-Based Feature Relationships

Graph-based feature relationships present an advanced method of feature engineering which establishes graph-based relationships between project attributes. The model uses nodes to represent project features which include budget and team experience and technical complexity [16]. The graph structure enables the model to track how one feature's alteration affects all other features. The study uses Graph Signal Processing GSP to examine graph signals which will help uncover hidden relationships and undocumented dependencies between project elements. The approach improves project risk evaluation and critical factor detection through its implementation of intricate networks in the predictive model.

3.3.2 Polynomial Feature Expansion + LASSO Selection

The process of polynomial feature expansion is vital for creating new features that demonstrate complex connections between numerical project attributes while eliminating duplicate information. The polynomial feature expansion method creates new features by generating all possible combinations of two existing numeric variables and their cubic forms which enables the model to detect non-linear project risk factors and their associated interactions. The process of producing all possible attribute combinations results in an excessive number of derived features as most of them turn out to be either unnecessary or duplicated. The LASSO method selects essential features through its penalization method which decreases predictive value to enable dimensionality reduction while stopping overfitting.

The LASSO operator works by adjusting the absolute values of feature coefficients; less important coefficients get pushed toward zero. The selected subset removes features that have zero or negative coefficients since LASSO works best with low coefficient values. The method retains both important features and irrelevant features at times. LASSO determines which features are highly correlated to be important while treating other features as unimportant. This method boosts the model's capacity to identify complex data patterns, which it processes with efficient computing power while achieving strong prediction accuracy.

3.3.3 Risk Indices

The Risk indices function as unified assessment tools that measure multiple project risk factors through their assessment of various related project characteristics.

Operational Risk Score: Calculates execution uncertainty through its computation of change request frequency, which it multiplies by budget utilization rate to show how project stability depends on changes to budget and scope. The expression of the operational risk score is given in Eqn. (3)

$$\beta_1=\varepsilon\times\omega \qquad (3)$$

where $\beta_1$implies operational risk score, εdenotes change request frequency, and ωindicates the budget utilization rate.

Technical Complexity Score: It measures project technical challenges through the assessment of technical debt and integration difficulties, and technology familiarity, which predict potential technical problems that will hinder project success. The computation of the technical complexity score is done using Eqn. (4)

$$\beta_2=\rho\times\vartheta\times\gamma \qquad (4)$$

where β_2 indicates technical complexity score, ρ specifies Technical_Debt_Level, ϑ represents integration complexity, and γ indicates technology familiarity.

Human Factor Score: This evaluates personnel and management risks through its analysis of team experience and team turnover rate, and project manager experience, which demonstrate the impact of human resources on project results. The expression of the human factor score is given in Eqn. (5)

$$\beta_3=A\times\mu\times\tau \qquad (5)$$

where$\beta_3$indicates human factor score, Aindicates team experience level, μdenotes the team Turnover rate, and τimplies PM_experience.

The indices create a numerical representation system that enables risk factors to be displayed at multiple levels of complexity, which helps the predictive model assess total project risk while determining critical focus areas.

International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

3.4 Lightweight Deep Learning Model for Project Risk Prediction

The Lightweight Deep Learning Model for Project Risk Prediction, referred to as TAM-Lite, is a hybrid architecture that efficiently forecasts project risk while retaining its ability to be understood. The system uses three main elements, which include TabNet and Mini Autoencoder, and Shallow Multilayer Perceptron (MLP), to achieve effective risk assessment.

3.4.1 TabNet

TabNet [18] serves as a dedicated DL model that processes tabular data while generating interpretable results. The system excels at handling structured datasets through its three main components, which include feature selection, attention mechanisms, and decision-making processes. TabNet uses dynamic feature selection to determine relevant features during its decision-making process, which results in both accurate outcomes and understandable results.

The structure of TabNet consists of the following components:

- Input Features and Embedding: The network requires processing of raw tabular input, which includes project demographics, operational metrics, human factors, technical aspects, and external influences after it undergoes normalization and transformation into a high-dimensional embedded space.
- Decision Steps: TabNet processes the input through a sequence of decision steps, which execute two tasks since each step selects different features and produces its final prediction. At each step, a subset of features is chosen using an attentive transformer module. The system uses this method to identify important features during the current phase while it hides features that hold less value.
- Attentive Transformer: The attentive transformer computes feature masks using learned attention scores. The masks control which features proceed to the next stage, while they also decide which features the system will disregard. The system creates feature importance scores, which enable model assessment through demonstration of the project attributes that most affect risk prediction. The expression of the attentive transformer and masking procedure is given in Eqn. (6)

$$k[p-1]: isparesemax(K[p-1].C_p[k-1])) \tag{6}$$

where k[p]represents the previous step, k[p]denotes prior scale, and E_kimplies a trainable function. The Attentive Transformer identifies and emphasizes the most relevant features to construct a refined feature representation, which is then forwarded to the learnable mask for further processing.

• Feature Transformer: After selecting the relevant features, the feature transformer applies non-linear transformations to extract higher-level representations. TabNet uses this capability to detect complex relationships between project characteristics that influence risk outcomes. The outcome of the feature transformer is formulated in Eqn. (7)

$$k[p] = \cup_{i=1}^{j}(\delta - G[i]) \tag{7}$$

Where δexplains the relationship among feature enforcement at one or more decision steps, G[i]indicates a learnable mask.

- Aggregation: The outputs of all decision steps are aggregated to produce the final prediction, which results in a probability distribution that shows three different risk levels (High, Medium, Low). The aggregation process combines information from different decision points to create a strong predictive model that delivers accurate results.

- Sparse Regularization: TabNet employs a sparsity-inducing regularization term during its training process, which directs the network to utilize only a narrow selection of features during each training iteration. The decision-making process relies solely on essential project attributes, which results in reduced overfitting while improving model interpretability. The overall structure of TabNet is shown in Figure 3.

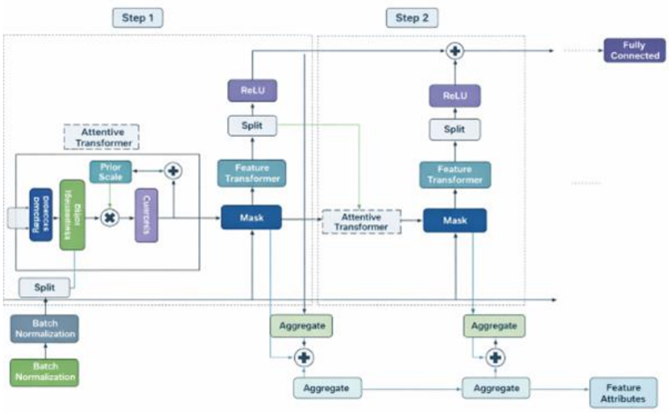


Figure 2. Architecture of TabNet

International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

3.4.2 Mini Autoencoder

The mini autoencoder component captures complex feature interactions and reduces data dimensionality. The system detects hidden data patterns through its ability to learn a compact representation of input features. The process establishes better class balance and reduces dataset noise, which results in more robust model performance.

3.4.3 Shallow Multilayer Perceptron (MLP)

The last predictor of the system uses shallow MLP as its final prediction method which combines TabNet transformed features and mini autoencoder features to calculate project risk probability. The TAM-Lite system maintains lightweight performance, and its basic structure enables rapid processing while it still detects non-linear connections between different features.

The stage-wise training approach combines interpretable rule-based learning with Gradient Boosted Rule Sets and fast neural network probability estimation to enhance model performance through Extreme Learning Machine and fuzzy logic classification uncertainty management. TAM-Lite generates project risk probabilities which it presents in three categories High Medium and Low while delivering precise and understandable insights that project risk management teams can use to make decisions.

3.4.3 Training process

The training process enables a predictive model to acquire knowledge about input data patterns which it then uses to create precise predictions. The research uses a stage-wise training method to predict project risk which achieves three objectives through its three stages of interpretability and computational efficiency and uncertainty management. The first stage uses Gradient Boosted Rule Sets (GBRS) to create rule-based classifications that show how essential project elements affect risk and provide understandable results. The second stage employs an Extreme Learning Machine (ELM), a single-layer neural network, to rapidly estimate the probability of each risk level.

The Fuzzy Logic Classifier converts essential performance indicators into three distinct linguistic levels which enables the model to handle uncertainty and produce risk assessments that people can understand. The model uses a cross-entropy loss function to achieve proper classification training, while the Adam optimizer with the early stopping method enables an effective learning process and protects from overfitting.

The Cross-Entropy loss function [19] operates as assessment tool which evaluates the accuracy of classification models to predict probability results for actual class labels. The system assesses project risk through two separate components which consist of actual project risk category and model output probability distribution. The cross-entropy loss function is expressed through the mathematical expression presented in Eqn. (8)

$$C_{loss} = -\sum_{v=1}^{V} t_v \log\left(t'_v\right) \tag{8}$$

wheret_vdenotes true labels and 〖t'〗_vrepresents the predicted probability of the classv. The training process produces a projected project risk probability, which classifies risks into three levels of High, Medium, and Low that can help organizations make strategic project risk management choices.

3.5 Probabilistic Risk Modelling

Bayesian Network construction is used to model the probabilistic and causal relationships among key project attributes and the predicted project risk level. The Bayesian Network structure has been created to show uncertainty and causal connections which exist in software development projects. The network uses nodes to represent essential project characteristics which include operational risk and technical complexity and human factors and organizational maturity and the estimated project risk level. The directed edges establish conditional dependencies between these nodes by showing how one element affects the probability of another element. The Bayesian Network establishes a system for probabilistic reasoning that enables scenario assessments and helps project managers understand the connections between different risk factors to enhance their decision-making processes and risk reduction methods.

3.6 Explainable Decision Support & Counterfactual Risk Mitigation

The Explainable Decision Support and Counterfactual Risk Mitigation module develops understandable project management solutions from model predictions. The system uses Gradient Boosted Rule Sets (GBRS) to create decision rules which explain project risk assessment through three risk categories which include high risk and medium risk and low risk. The counterfactual analysis identifies which small and achievable changes will lead to reduced risk predictions.

The model identifies three project-level changes which include improving team performance and reducing project difficulty and increasing resource accessibility as methods to decrease project risks from their current high level to a medium state and finally to a low risk assessment. The counterfactual explanations enable decision-makers to see how each element affects project risk, so they can develop precise solutions that require minimal interruptions. The framework provides organizations with risk predictions which include data-based recommendations that follow actual project limitations to assist organizations in their risk management process.

IV. RESULT AND DISCUSSION

The section provides a comparative assessment of the proposed model based on significant evaluation metrics. The research study evaluates the system performance and efficiency testing through direct comparison with established traditional methods to demonstrate the successful operation of the developed method.

4.1 Experimental setup

The assessment of the developed model is conducted in the Python platform, where the essential performance metrics, including Accuracy, Precision, Recall, F1-Score, Balanced Accuracy, Brier score, Cumulative gain, Scenario Sensitivity Index (SSI), are used to evaluate the model's capability to classify data and its overall system performance.

4.2 Dataset description

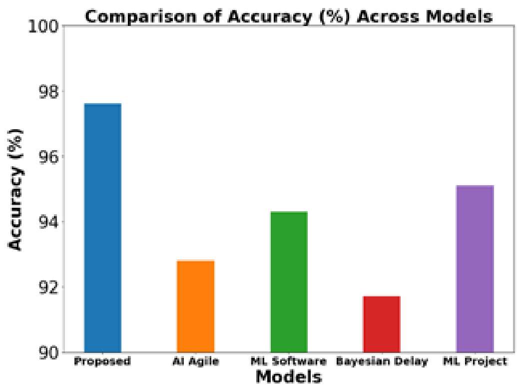
The Project Management Risk Dataset [20] contains organized data that documents potential risks that arise during every stage of project development, from the initial planning stage to the final stakeholder management phase. The dataset typically includes attributes such as risk category, likelihood of occurrence, impact severity, risk priority level, mitigation strategies, and project outcomes. The system demonstrates the ability to assess internal risks which include scope creep and resource shortages and technical complexities together with external risks that originate from market fluctuations and regulatory changes and environmental factors.

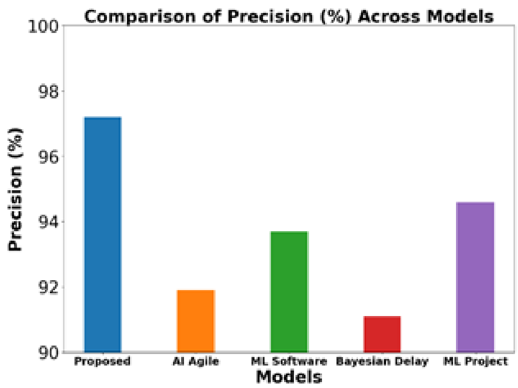
4.3 Performance analysis

The results shown in Table 2 demonstrate that the proposed model achieves its highest performance level through an accuracy of 97.6% and, a precision rating of 97.2% and, a recall measurement of 97.8%, and an F1-score value of 97.5%, which establishes its high classification performance. The system demonstrates consistent performance throughout all risk categories, which is shown through its 97.5% balanced accuracy score, with its 0.024 Brier score, which shows exceptional probability assessment accuracy. The model demonstrates its abilities to detect key risk situations through its total achievement of 96.4% and its scenario sensitivity index (SSI) performance, which reached 0.91. The AI Agile model achieves an accuracy rating of 92.8% with a total F1-score of 92.1%, and it produces a Brier score that exceeds 0.072. The Bayesian Delay method achieves lower results than the previous techniques with 91.7% accuracy and 91.3% F1-score, with 0.081 Brier score, which represents the highest value. The ML Software and ML Project models achieve moderate accuracy results, which include 94.3% and 95.1%, with F1-scores of 93.9% and 94.8%, and Brier scores of 0.053 and 0.048. The proposed model demonstrates better accuracy, more dependable results, and higher sensitivity during project management risk assessment. Figure 3 illustrates a comparative assessment of the proposed TAM-Lite framework.

Table 2. Comparative discussion

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Balanced Accuracy (%)	Brier Score	Cumulative Gain (%)	Scenario Sensitivity Index (SSI)
Proposed	97.6	97.2	97.8	97.5	97.5	0.024	96.4	0.91
AI Agile	92.8	91.9	92.4	92.1	92.3	0.072	88.2	0.76
ML Software	94.3	93.7	94.1	93.9	94	0.053	90.7	0.81
Bayesian Delay	91.7	91.1	91.6	91.3	91.5	0.081	86.4	0.73
ML Project	95.1	94.6	95	94.8	94.9	0.048	92.3	0.84





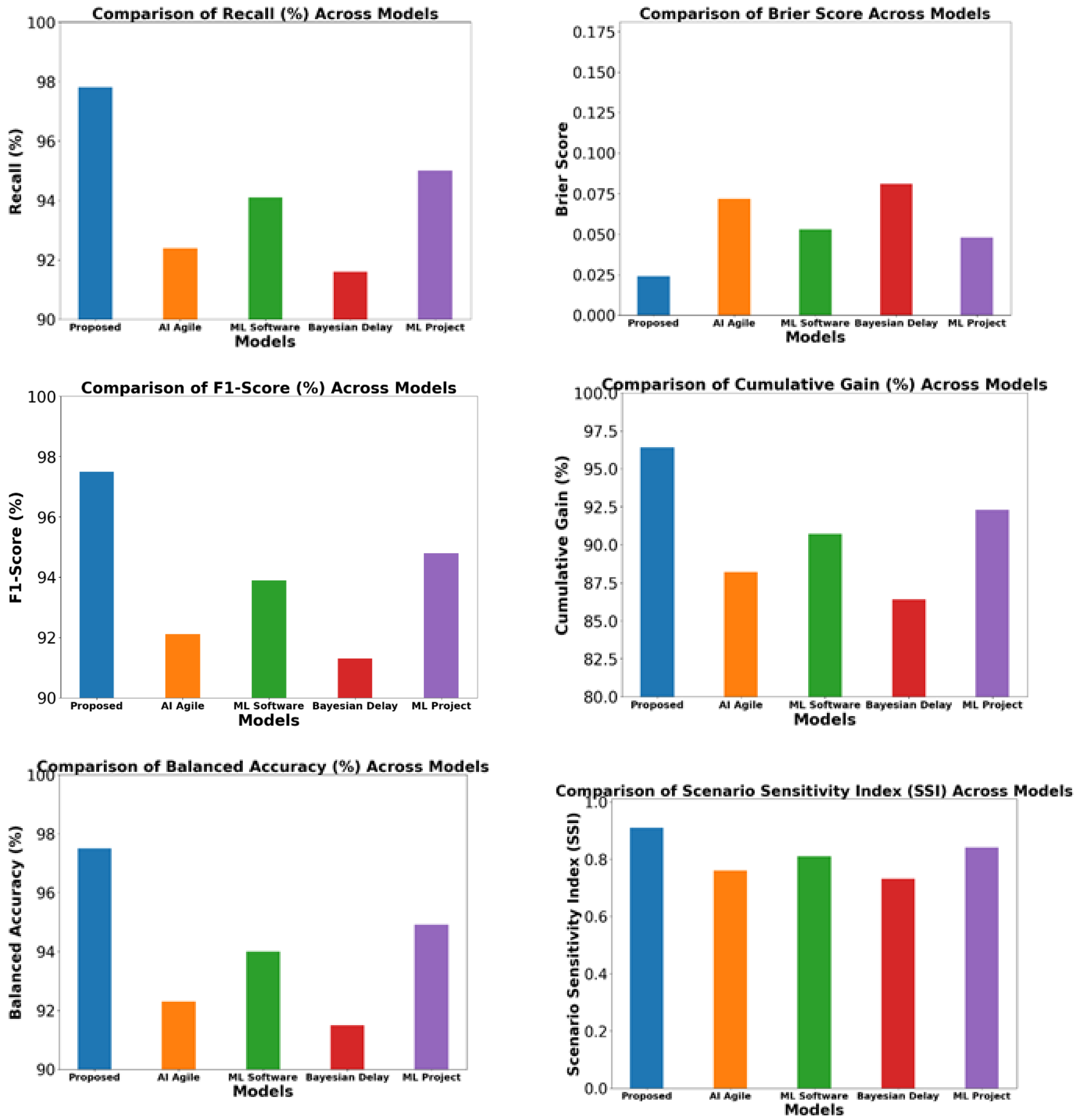


Figure 3. Performance analysis of the proposed model

V. Conclusion

The complexity of software projects results in various risks that emerge from technical challenges, changing project needs, human behaviour, organizational issues, and unpredictable external factors. Existing predictive models operate as black boxes, which provide project managers with insufficient system insights and operational guidance, thus decreasing user confidence and real-world application. Hence, this research introduces a lightweight system that uses data analysis to help make project risk assessments and management decisions in complex project settings. The system employs a complete Project Management Risk Dataset, which includes data on project demographics and operational metrics, human factors, organizational context, technical specifications, and external project-related factors. Then, the model used an effective pre-processing method, which used one-hot encoding to process categorical data and Min-Max normalization to handle budget, timeline, and risk-related data. The system conducts advanced feature engineering by creating graph models that represent project attributes with Graph Signal Processing to identify hidden system dependencies and polynomial feature expansion with LASSO to determine essential features. Moreover, the proposed TAM-Lite architecture is used to provide risk prediction for a project risk-based system, which uses Gradient Boosted Rule Sets, Extreme Learning Machines, and fuzzy logic classification. The model generates risk level probabilities, which analysts use to study the data through Bayesian Networks and counterfactual explanations to provide detailed insights and practical solutions for successful risk management. The proposed framework will be extended by adding real-time project monitoring data, which will allow assessment of project risks to continue throughout the entire project lifecycle.

REFERENCES

1. M. Ibraigheeth, “Assessment tool for software failure and risk mitigation in project management,” in Proc. 2024 IEEE Int. Conf. on Data and Software Engineering (ICoDSE), pp. 49–54, 2024.
2. T. Gilles, “Leveraging AI-driven risk prediction models to enhance software project success,” Int. J. Software Project and Quality Management, vol. 3, no. 1, pp. 1–8, 2024.
3. S. R. Bauskar, C. R. Madhavaram, E. P. Galla, and J. R. Sunkara, “Predictive analytics for project risk management using machine learning,” J. Data Analysis and Information Processing, vol. 12, no. 4, pp. 566–580, 2024.
4. E. Johnson, “Leveraging machine learning for risk prediction and mitigation in complex project environments,” in Distributed Learning and Broad Applications in Scientific Research, 2025.
5. E. Kula, E. Greuter, A. van Deursen, and G. Gousios, “Dynamic prediction of delays in software projects using delay patterns and Bayesian modeling,” arXiv preprint, arXiv:2309.12449, 2023.
6. M. Pilliang and M. Munawar, “Risk management in software development projects: A systematic literature review,” Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika, vol. 8, no. 2, p. 3, 2022.
7. S. A. Jahan, “Utilizing predictive analytics and machine learning for enhanced project risk management and resource optimization,” IPHO–J. Adv. Research in Business Management and Accounting, vol. 2, no. 11, pp. 24–31, 2024.
8. A. A. ForouzeshNejad, F. Arabikhan, A. Gegov, R. Jafari, and A. Ichtev, “Data-driven predictive modelling of agile projects using explainable artificial intelligence,” Electronics, vol. 14, no. 13, p. 2609, 2025.
9. J. Xu, Y. Wang, R. Li, Z. Wang, and Q. Zhao, “An effective software risk prediction management analysis using machine learning and data mining methods,” arXiv preprint, arXiv:2406.09463, 2024.
10. E. Kula, E. Greuter, A. van Deursen, and G. Gousios, “Dynamic prediction of delays in software projects using delay patterns and Bayesian modeling,” in Proc. 31st ACM Joint European Software Engineering Conf. and Symp. on the Foundations of Software Engineering (ESEC/FSE), pp. 1012–102, 2023.
11. S. Bauskar, C. Madhavram, and E. P. Galla, “Predictive analytics for project risk management using machine learning,” 2024.
12. S. Dhanka et al., “Padding interpolation, median imputation, RobustScaler, and particle swarm optimization with heterogeneous classifiers: A robust combination for effective heart disease diagnosis,” Frontiers in Medicine, vol. 12, p. 1721740, 2025.
13. T. Emmanuel et al., “A survey on missing data in machine learning,” J. Big Data, vol. 8, no. 1, p. 140, 2021.
14. S. Okada, M. Ohzeki, and S. Taguchi, “Efficient partition of integer optimization problems with one-hot encoding,” Scientific Reports, vol. 9, no. 1, p. 13036, 2019.
15. M. S. Nadeem, M. A. Farooq, and S. Afzal, “A hybrid model using clustering and reinforcement learning for test case prioritization in agile environments,” Pakistan J. Scientific Research, vol. 5, no. 2, pp. 95–112, 2025.
16. Z. Zhang and E. R. Hancock, “A graph-based approach to feature selection,” in Proc. Int. Workshop on Graph-Based Representations in Pattern Recognition, Berlin, Germany: Springer, 2011, pp. 205–214.

17. P. Ghosh et al., “Expert cancer model using supervised algorithms with a LASSO selection approach,” Int. J. Electrical and Computer Engineering, vol. 11, no. 3, pp. 2632-2640, 2021.
18. K. McDonnell et al., “Deep learning in insurance: Accuracy and model interpretability using TabNet,” Expert Systems with Applications, vol. 217, p. 119543, 2023.
19. Z. Zhang and M. Sabuncu, “Generalized cross entropy loss for training deep neural networks with noisy labels,” in Advances in Neural Information Processing Systems, vol. 31, 2018.
20. The project management risk dataset is taken from “Project Management Risk Raw,” accessed in February 2026.