Propuesta de un modelo de Redes Neuronales Artificiales para la predicción del éxito o fracaso de proyectos basados en el enfoque PMBOK

Proposal for an Artificial Neural Network Model to Predict the Success or Failure of Projects Based on the PMBOK Approach

DOI: https://dx.doi.org/10.17981/ingecuc.22.1.2026.05
Artículo de Investigación Científica. Fecha de Recepción: 28/01/2026, Fecha de Aceptación: 25/03/2026.

Javier Antonio Pinedo-Cabarcas https://orcid.org/ 0000-0002-1819-9818
Universidad de Cartagena. Cartagena, (Colombia)
Javierpinedo@unicartagena.edu.co

Gabriel Chanchi-Golondrino, https://orcid.org/ 0000-0002-0257-1988
Universidad de Cartagena. Cartagena, (Colombia)
gchanchig@unicartagena.edu.co

Miguel Ángel García-Bolaños, https://orcid.org/ 0009-0003-1518-4551
Universidad de Cartagena. Cartagena, (Colombia)
mgarciab2@unicartagena.edu.co

To cite this paper
J. Pinedo-Cabarcas, G. Chanchi-Golondrino & M. Garcia-Bolaños "Proposal for an Artificial Neural Network Model to Predict the Success or Failure of Projects Based on the PMBOK Approach," INGE CUC, vol. 22, no. 1, 2026. DOI: *https://dx.doi.org/10.17981/ingecuc.22.1.2026.05*

Resumen

A nivel mundial cada vez más las empresas u organizaciones implementan herramientas metodológicas o estándares de mejores prácticas para la gestión de sus proyectos, como es el caso del PMBOK. No obstante, es importante reconocer que dada la robustez de estos métodos se requiere contar con una buena experticia por parte de los gerentes del proyecto. Por lo anterior, el uso de herramientas predictivas puede contribuir a la toma de decisiones informadas en las diferentes etapas del ciclo de vida del proyecto. Así, en este trabajo se propuso como contribución un nuevo modelo soportado en redes neuronales artificiales para la predicción del éxito o fracaso de un proyecto gestionado mediante PMBOK, a partir de características de un proyecto tales como: los recursos involucrados, el grupo de proyecto y los indicadores del proyecto. Para el desarrollo del proyecto se hizo una adaptación de la metodología CRISP-DM a cuatro fases. Como resultado, el modelo obtuvo un ajuste consistente en las 100 épocas de entrenamiento, teniendo un accuracy por encima del 95% luego de las 60-70 épocas, lo que sugiere una excelente capacidad de ajuste y generalización en la predicción del resultado de un proyecto. Como conclusión, el modelo representa un aporte clave en la gestión de proyectos bajo PMBOK, siendo su factor diferencial la inclusión de los recursos, del atributo grupo del proyecto y de la representación vectorial de características cualitativas del proyecto identificadas en el atributo indicador.

Palabras clave

Gestión de proyectos, PMBOK, CRISP-DM, redes neuronales artificiales, Inteligencia Artificial

Abstract

Globally, companies and organizations are increasingly implementing methodological tools or best practice standards for project management, such as PMBOK; however, it is important to recognize that, given the robustness of these approaches, a high level of expertise is required from project managers. In this context, the use of predictive tools can contribute to informed decision-making across the different stages of the project life cycle. Thus, this study proposes as its main contribution a novel model based on neural networks for predicting the success or failure of a project managed under PMBOK, using project features such as the resources involved, the project group according to PMBOK, and project indicators. For the development of this work, an adaptation of the CRISP-DM methodology into four phases was carried out. As a result, the model achieved consistent fitting over 100 training epochs, reaching an accuracy above 95% after 60–70 epochs, which suggests excellent fitting and generalization capabilities in predicting project outcomes. In conclusion, the model represents a key contribution to project management under PMBOK, with its differential factor being the inclusion of resources, the project group attribute, and the vector representation of qualitative project features identified within the indicator attribute.

Keywords

Project management, PMBOK, CRISP-DM, artificial neural networks, Artificial Intelligence

INTRODUCTION

One of the most widely disseminated approaches to project management is PMBOK, which is considered a reference framework that compiles the best practices of the project management discipline across different application domains (Simonaitis et al., 2023). This framework is composed of distinct process groups and knowledge areas that guide the planning, execution, monitoring and control, and closure of projects across various types of industries, such that it establishes guidelines applicable to multiple domains, thereby facilitating project management within a broad and diverse context (Davidov et al., 2023). In this sense, although this project management approach has been associated with greater success in projects within complex contexts such as data warehouses and information technology, it can also be transferred to non-traditional domains, such as educational planning, providing in all these contexts a clear structure, traceability, and repeatability (García et al., 2024).

The relevance of the PMBOK approach is grounded in its capacity to improve effectiveness and efficiency in project management, such that its systematic use enables the standardization of processes, which is crucial in large-scale projects that require precise coordination of multiple resources and stakeholders (Huda et al., 2019). Thus, among the main advantages of using the PMBOK approach in project management are the following (Bogojevic, 2020; García et al., 2024; Kinsella, 2002; Simonaitis et al., 2023): a comprehensive structure of the project life cycle, facilitating its organization, monitoring, and control; a wide range of tools and techniques for planning, risk management, cost management, and scheduling; the ability to measure progress, analyze delays through baselines, and ensure quality control; proper identification and management of risk responses; and well-defined processes and a common language that enable effective coordination among stakeholders.

Despite the advantages of this approach in project management, its effectiveness depends strongly on the expertise and attitude of both the project manager and the team, since by itself it does not guarantee results (Wells, 2012). In this same sense, the implementation of PMBOK can be complex and overwhelming, which may lead to errors in the application of techniques and to a lack of alignment with project objectives (Davidov et al., 2023). Likewise, another challenge to be highlighted of the PMBOK methodology is its perception as a rigid approach that is not easily adaptable to certain application environments based on innovation or in which adaptability is essential, and therefore, in these contexts, the use of agile methodologies such as Scrum or Kanban is preferred (Henderson et al., 2013). Now, considering the challenge related to the complexity of managing projects, tools based on predictive analytics can help produce more accurate and effective estimates of duration, costs, and potential risks, facilitating project management through the exploitation of statistics and historical data (Santos & Cabral, 2008). In this same sense, within PMBOK-based project management, the areas that benefit the most from the application of artificial intelligence and machine learning models are those related to the estimation of costs, schedules, and resources (Fridgeirsson et al., 2021, 2023; Rosenberger & Tick, 2021). In this same sense, the use of machine learning techniques not only improves the effectiveness of project management, but also promotes more agile and efficient execution (Alfaifi & Aksoy, 2023).

There are several studies in which predictive models have been proposed as support for PMBOK-based project management. Thus, in (Taye & Feleke, 2022), different machine learning models (SVM, decision trees, KNN) are evaluated for failure prediction across the 10 PMBOK Knowledge Areas (PMKAs) at the software project level, finding that the SVM model achieves the best performance with approximately 92% accuracy. In this same sense, (Narbaev et al., 2024) proposed, as a contribution, an XGBoost model for forecasting the total cost of a PMBOK-managed project throughout its life cycle, obtaining better results compared to other machine learning models and improving traditional techniques such as Earned Value Management (EVM), in such a way that the model enables the generation of early warnings of cost overruns. On the other hand, in (Nishat et al., 2025), different machine learning models, such as LSTM and CatBoost, are tuned and evaluated for the prediction of variation orders in construction projects, understood as contractual modifications that affect the project scope, cost, or schedule, with the aim of providing predictive support for change management, risk management, and scope control. Likewise, in (Reznikov, 2025), a logistic regression model was applied to the prediction of relevant events related to environmental parameters in the context of poultry production, suggesting, based on its performance, good feasibility for managing data from projects related to sustainability and environmental regulation. Similarly, in (Xu et al., 2014), decision tree models and support vector machines were applied to predict outcomes related to cost management and risk assessment, significantly improving decision-making across different PMBOK areas. Now, in (Uddin et al., 2022), different machine learning models were evaluated for the prediction of cost overruns in construction projects, finding that the Random Forest model achieved the best performance with 76% accuracy compared to other models such as SVM, logistic regression, KNN, and artificial neural networks. In a similar manner, in (Abuhussain et al., 2025), a study focused on sustainability and quality and resource management was developed through the integration of BIM (Building Information Modeling), PMBOK, and machine learning, obtaining results showing that a neural network-based model achieved an R^2 of 0.95 in energy prediction, leading to energy savings of 13–15% and greenhouse gas emission reductions of up to 25%. In that same sense, (Basaran et al., 2025) proposed a dynamic model based on decision trees for evaluating subcontractor performance on construction sites, achieving high accuracy and strong agreement with real evaluations. On the other hand, (Gabel et al., 2023) proposed the use of neural networks for risk assessment in research and development projects, finding that this model improves outcome prediction and enables a more proactive approach to risk management. Complementarily, in (Alatawi et al., 2023), artificial neural networks were employed for risk prediction in software projects based on historical information, financial variables, and performance metrics, achieving an accuracy of 97.12% in the identification of global and individual risks, thus demonstrating the potential of these models as precise tools for early risk management and mitigation, although their application is limited to the software project domain. Finally, in (Alsugair et al., 2023), the use of an artificial neural network was proposed for the early prediction of the final duration of construction projects based on historical information from 135 projects and variables such as cost and estimated time, achieving an average MAPE error of approximately 12.22% and superior performance compared to classical methods such as Earned Value Management (EVM) and linear regression, thereby evidencing its usefulness as a decision-support tool in early project phases.

The aforementioned works reveal the relevance of developing research based on the use of predictive models as support for decision-making in project management under the PMBOK approach. Thus, this article proposes, as its main contribution, a neural network–based model for the classification of projects from different fields of application into the categories of success and failure, based on attributes related to the group, resources, and indicators. The proposed model was trained using a dataset that compiles a set of projects developed under the PMBOK model, for each of which information is included regarding project resources, the indicators considered, the PMBOK group or phase in which the project is monitored, and the lessons learned. The model was developed using the TensorFlow library in Python and, based on the performance results obtained, it serves as a reference to be considered in the monitoring of projects managed under the PMBOK model, with the aim of supporting early decision-making across the different phases of project management. This is particularly relevant considering that when project scope, time, and budget are not properly managed, it is highly likely that the project will not be completed within the stipulated timeframe, will require a higher budget, or, in the worst case, will fail by not delivering the expected service or product (Chola & Chibomba, 2025; SLAEAT, 2024). Likewise, in the academic context, the proposed model serves as a reference for evaluating projects based on their resources and indicators, becoming an excellent simulation tool for classroom appropriation of widely adopted frameworks such as PMBOK. As a differentiating factor, the use of natural language processing techniques is highlighted for the vectorization of the indicator-related attribute through the combination of the TF-IDF technique with latent semantic indexing (LSI), as well as the consideration of the specific PMBOK group as an explicit model attribute.

The remainder of the article is organized as follows:
Section 2 presents the methodological phases considered for the development of the study. Section 3 describes the results of this research, including the adaptation of the original dataset, the definition of the model architecture, and its training and evaluation, taking into account the metrics of accuracy, precision, recall, and F1-score. Finally, Section 4 presents the conclusions and the future work derived from this research.

METHODOLOGY

For the development of the present research, an adaptation of the CRISP-DM data science methodology into four phases was carried out: Phase 1. Business and data understanding, Phase 2. Data preparation, Phase 3. Modeling, and Phase 4. Model evaluation and deployment, as can be observed in Figure 1. The selection of the CRISP-DM methodology is mainly motivated by the fact that it provides a flexible and well-defined structured model, which favors the clear organization and management of machine learning and data science projects in general across different application contexts, avoiding the exclusion of fundamental steps, while at the same time ensuring that critical steps are not omitted and that iterations between phases can be performed as necessary (Abasova et al., 2021; Martinez-Plumed et al., 2021; Saltz & Krasteva, 2022; Schröer et al., 2021; Shimaoka et al., 2025).

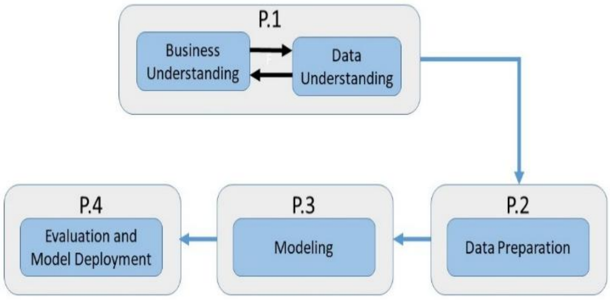


Figure 1. Considered methodology

In Phase 1 of the methodology, a database was constructed from the historical records of projects executed in Colombia under the PMBOK project management model, such that a dataset composed of 186 instances and 13 columns was obtained, as shown in Figure 2. Likewise, it is worth noting that, in order to ensure the replicability of the experimentation, both the dataset and the experimental setup developed were made publicly available on the GitHub platform.[1]

	nombre_proyecto	tipo_proyecto	id_proyecto	fecha_proyecto	area	grupo	recurso_tiempo	recurso_costo	recurso_varios	resultado	descripcion_actividad	leccion_aprendida	indicador
0	API One	Desarrollo de software	1	2020	4	1	1	136000.0	5	exito	Se desarrolla el acta de constitución del proy...	El tiempo destinado para el desarrollo del act...	Los recursos en base a tiempo, costo y número ...
1	API One	Desarrollo de software	1	2020	4	1	1	136000.0	7	exito	Se desarrolla el plan para la dirección del pr...	El presupuesto de costo destinado al desarroll...	Los recursos en base a tiempo, costo y número ...
2	API One	Desarrollo de software	1	2020	4	2	155	136000.0	7	exito	Se dirige y gestiona el trabajo del proyecto	El recurso humano destinado para esta activida...	Los recursos en base a tiempo, costo y número ...
3	API One	Desarrollo de software	1	2020	4	2	1	136000.0	3	exito	Se gestiona el conocimiento del proyecto	El tiempo destinado para gestionar el conocimi...	Los recursos en base a tiempo, costo y número ...
4	API One	Desarrollo de software	1	2020	4	3	155	136000.0	4	exito	Se monitorea y controla el Trabajo del Proyecto	El presupuesto de costo destinado al monitoreo...	Los recursos en base a tiempo, costo y número ...

Figure 2. Constructed dataset

The dataset mainly includes the project name, project type, project identifier, project execution date, the corresponding PMBOK area, the PMBOK group associated with the project phase, the cost resource or project cost, the time resource or number of days corresponding to the development of the project phase, various resources (machinery, human resources, among others), the project outcome (success or failure), the description of the activity carried out in each project, the lessons learned, and the indicator used by each project (time, cost, number of people, etc.). This dataset was constructed in an Excel spreadsheet and subsequently loaded into a Google Colab notebook using the DataFrame structure of the pandas library, in order to facilitate data manipulation for preprocessing, training, and validation of the neural network model.

Now, in Phase 2 of the methodology, five attributes or columns were selected from the dataset shown in Figure 2, namely: "group," "time_resource," "cost_resource," "various_resources," and "indicator," taking into account their conceptual relevance within a project. Likewise, as the target attribute or variable, the variable "outcome" was defined, which, as mentioned, can take two possible values: success or failure. Thus, based on these variables, the implementation and tuning of a binary classifier were pursued, which allows identifying whether a project may succeed or fail based on the five selected attributes. Since the variable "group" is categorical and can take seven possible values according to PMBOK (1. Initiation, 2. Project governance, 3. Planning, 4. Execution, 5. Monitoring and control, 6. Value management, and 7. Closure), a first step involved mapping it into three possible categories: 1. Strategic direction, 2. Operational management, and 3. Value delivery and closure. This mapping was performed considering model simplicity and the limited dimensionality of the dataset. After mapping to three categories, one-hot encoding was applied to the modified "group" attribute, which enables the creation of as many columns as categories in a variable, thereby preventing the model from being affected by spurious ordinal bias (Kunanbayev et al., 2021; Poslavskaya & Korolev, 2023). In this way, three columns derived from the "group" variable were obtained, one for each category. Regarding the variables "time_resource," "cost_resource," and "various_resources," they were kept unchanged, since they are numerical and non-categorical. On the other hand, given that the "indicator" variable is of string type, it was vectorized using the TF-IDF

[1]https://github.com/gabrielchanchi/gestion_proyectos

natural language processing technique combined with latent semantic indexing (LSI), thus obtaining a four-dimensional vector representation for this column. In practice, any desired number of dimensions can be obtained; however, a value lower than five was selected, given that the dataset contains 186 records and it is necessary to maintain an adequate relationship between the number of columns and rows. The LSI technique belongs to the field of information retrieval and is responsible for transforming the classical TF-IDF matrix into a latent concept space using singular value decomposition (SVD), in which only the first k singular values are retained, resulting in a lower-dimensional space that captures the main semantic structure (Lan et al., 2021; Landauer & Dumais, 2008). Thus, in this preprocessing phase, a new dataset with 10 columns was obtained: three derived from the "group" attribute, three corresponding to resources (time, cost, and various), and four resulting from the vectorization of the "indicator" attribute. It is worth noting that, within the adjusted dataset, the class distribution was 54% for "success" and 46% for "failure". Although this does not constitute a perfectly uniform balance, the proportion between both classes is practically even, thereby ensuring a comparable number of instances in each group. In this context, a stratified sampling approach was additionally adopted in order to preserve this distribution throughout the data partitioning process.

On the other hand, in Phase 3 of the methodology, the design of the artificial neural network (ANN) architecture was carried out, such that 10 input neurons and one output neuron were defined, with a sigmoid activation function due to the binary nature of the problem. Likewise, the model includes three hidden layers with 12, 8, and 4 neurons, respectively, each using a ReLU activation function and Dropout layers with a rate of 30% (see Figure 3). The model was trained using the Adam optimizer with a learning rate of 0.001 and the binary cross-entropy loss function. This architecture was implemented by leveraging the capabilities provided by the TensorFlow library in Python. Specifically, the implementation of the architecture in TensorFlow can be mathematically modeled by Equation (1) (Goodfellow et al., 2016).

$$\hat{y} = f\left(\sum_{i=1}^{n} \omega_i x_i + b\right) \quad (1)$$

In Equation (1), xi represents the input to the neuron, ωi denotes the weight associated with each input, b corresponds to the bias term, f is the activation function (such as ReLU, tanh, or sigmoid), and $\hat{y}$ represents the estimated output of the neuron.

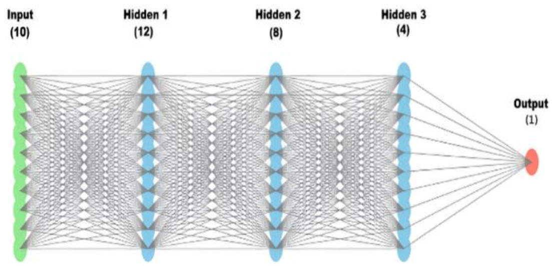
Results and discussion

Regarding the results, the process initially started from the dataset shown in Figure 2 in order to adapt it so that it would be suitable for training the ANN model, as illustrated in Figure 3. Thus, from the string-type "indicator" column, the first four columns of the new dataset were obtained as a result of the vectorization performed using the TF-IDF method in combination with the latent semantic indexing (LSI) technique. The following three attributes ("time_resource," "cost_resource," and "various_resources") being non-ordinal numerical variables, were left unchanged. Finally, based on the "group" attribute of the original dataset, a mapping to three categories was performed and one-hot encoding was applied to obtain the last three attributes (one attribute for each mapped category of the "group" column). This encoding was carried out to prevent the ordinal nature of the original "group" attribute from introducing bias during model training.

	indicador_svd_01	indicador_svd_02	indicador_svd_03	indicador_svd_04	recurso_tiempo	recurso_costo	recurso_varios	resultado	grupo_1	grupo_2	grupo_3
0	0.998234	-0.057316	-0.005837	0.014401	1	136000.0	5	exito	1.0	0.0	0.0
1	0.998234	-0.057316	-0.005837	0.014401	1	136000.0	7	exito	1.0	0.0	0.0
2	0.998234	-0.057316	-0.005837	0.014401	155	136000.0	7	exito	0.0	1.0	0.0
3	0.998234	-0.057316	-0.005837	0.014401	1	136000.0	3	exito	0.0	1.0	0.0
4	0.998234	-0.057316	-0.005837	0.014401	155	136000.0	4	exito	0.0	0.0	1.0

Figure 3. Adapted dataset for ANN model training

Based on the dataset shown in Figure 3, the ANN model was trained, whose architecture and implementation are presented in Figure 4, using the TensorFlow library in Python.




```
model = Sequential()
model.add(Dense(units=12,activation='relu',input_shape=(10,),kernel_regularizer=regularizers.l2(1e-4)))
model.add(Dropout(0.3))
model.add(Dense(units=8,activation='relu',kernel_regularizer=regularizers.l2(1e-4)))
model.add(Dropout(0.3))
model.add(Dense(units=4,activation='relu',kernel_regularizer=regularizers.l2(1e-4)))
model.add(Dense(1, activation='sigmoid'))

model.compile(optimizer=Adam(learning_rate=0.001),loss='binary_crossentropy',metrics=[
        'accuracy',
        Precision(name='precision',thresholds=0.5),
        Recall(name='recall',thresholds=0.5),
        F1Score(name='f1', threshold=0.5),
        AUC(name='auc')
    ])
model.summary()
```

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 12)	132
dropout (Dropout)	(None, 12)	0
dense_1 (Dense)	(None, 8)	104
dropout_1 (Dropout)	(None, 8)	0
dense_2 (Dense)	(None, 4)	36
dense_3 (Dense)	(None, 1)	5

Figure 4. ANN model architecture and implementation

As shown in Figure 4, the ANN model consists of an input layer with 10 neurons, corresponding to the 10 dataset attributes, and a single output neuron due to the binary nature of the classification problem. The architecture includes three hidden layers composed of 12, 8, and 4 neurons, respectively, each employing the ReLU activation function, which favors the modeling of nonlinear relationships in the data. In order to mitigate overfitting and improve the model's generalization capability, Dropout layers with a rate of 30% were incorporated after the first two hidden layers. Additionally, all dense layers employ L2 regularization with a penalty factor of 1×10^{-4}, which contributes to controlling the magnitude of the weights during the training process. Likewise, the model was optimized using the Adam algorithm, with a learning rate of 0.001 and the binary cross-entropy loss function. For performance evaluation, the metrics of accuracy, precision, recall, and F1-score were considered, using a decision threshold of 0.5 for binary classification.

Now, once the model was defined and implemented, its training was carried out considering the training and test sets obtained from the dataset shown in Figure 3. Thus, after training and evaluating the model over 100 epochs, the loss and accuracy results presented in Figure 5 were obtained.

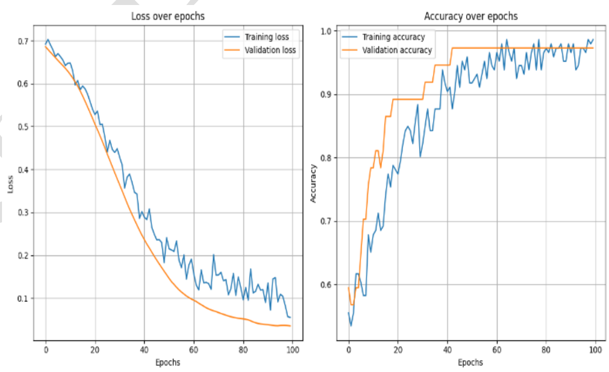


Figure 5. Results obtained from the training and evaluation of the model

The results presented in Figure 5 make it possible to identify that, in terms of the loss metric, both the training and validation sets exhibit a progressive and consistent decrease across the different epochs, which evidences a stable learning process of the ANN model. In particular, the loss shows a pronounced reduction during the first 30–40 epochs, followed by a more gradual decrease, until it stabilizes approximately from epoch 60 onward, without significant increases or divergences between both curves, which suggests the absence of overfitting. Complementarily, the accuracy metric shows a sustained upward trend, reaching values above 90% around epoch 30 and progressively stabilizing from epochs 60–70 with performance above 95% for both training and validation. Thus, considering that the model was trained and evaluated over a total of 100 epochs, these results demonstrate adequate model convergence and a consistent generalization capability in the binary classification process.

In this same sense, based on the evaluation performed with the ANN model on the dataset, the confusion matrix presented in Figure 6 was obtained.

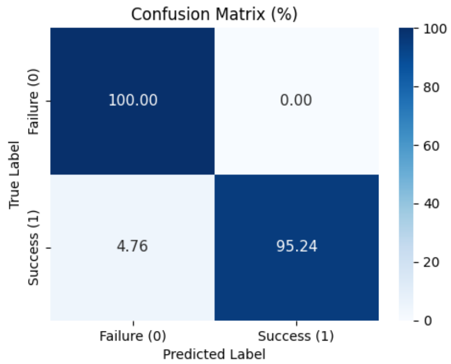


Figure 6. Obtained confusion matrix

The normalized confusion matrix presented in Figure 6 demonstrates a robust and well-balanced performance of the proposed ANN model in the classification of successful (1) and failure (0) projects. In particular, a false positive rate of 0% is observed, indicating that no failure project was erroneously classified as successful, which highlights the reliability of the model in identifying failure scenarios. On the other hand, the false negative rate corresponds to 4.76%, reflecting a small proportion of successful projects that were classified as failure. Complementarily, the model achieves a correct classification rate of 100% for failure projects and 95.24% for successful projects, which suggests a high generalization capability and an adequate balance between sensitivity and specificity. These results confirm that the ANN model exhibits consistent and reliable behavior for project success evaluation, minimizing critical errors and supporting informed decision-making in project management contexts. Likewise, the precision, recall, and F1-score metrics of the trained ANN model were also obtained; these metrics are derived from the confusion matrix and are presented in the bar chart shown in Figure 7.

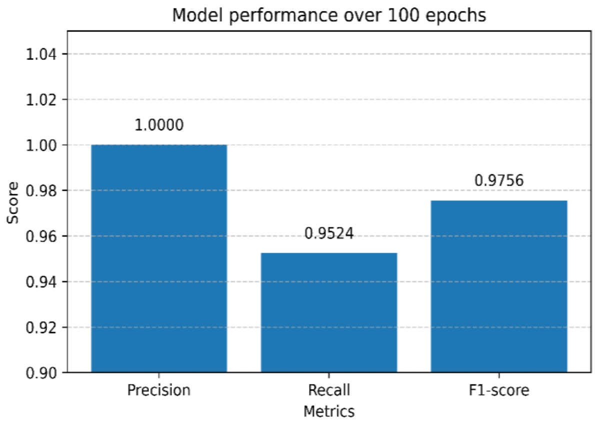


Figure 7. ANN model performance over 100 epochs

Based on the results shown in Figure 7, it can be observed that the precision metric reaches a value of 1.0000, which indicates that all instances classified by the model as successful projects (class 1) effectively correspond to that category, evidencing the absence of false positives. Likewise, the recall metric attains a value of 0.9524, reflecting the model's high capability to correctly identify successful projects, although with a minimal proportion of false negatives. This relationship between precision and sensitivity is adequately summarized by an F1-score of 0.9756, confirming an optimal balance between both metrics. Thus, taken together, these results suggest that the model exhibits balanced behavior in discriminating between successful (1) and failure (0) projects, minimizing critical classification errors and demonstrating robust capability for evaluating project success.

At the discussion level, it is important to highlight that this work proposes, as its main contribution, a novel neural network–based model for predicting the success or failure of a project managed under the PMBOK framework, based on the resources involved, the project group according to PMBOK, and the project indicators. This is achieved using a dataset that compiles the characteristics of different projects from multiple application domains, all developed and managed under the PMBOK approach across diverse contexts. When evaluating the model performance over 100 epochs, it was observed that after approximately 60–70 epochs the model achieves consistent performance on both training and test sets, with accuracy values above 95% and excellent performance in the precision (100%), recall (95.24%), and F1-score (97.54%) metrics across the 100 epochs. Thus, when compared with state-of-the-art works such as (Taye & Feleke, 2022), (Reznikov, 2025) and (Narbaev et al., 2024), which rely on more classical models, the proposed model not only achieves better fitting and performance in the metrics derived from the confusion matrix, but also incorporates, as an added value, the integration of the TF-IDF technique combined with latent semantic indexing (LSI) for the vector representation of the project descriptive attribute referred to as “indicator.” This constitutes a relevant contribution toward improving model fitting capabilities in these application contexts by leveraging the descriptive information of projects.

Now, one of the limitations that needed to be addressed in this project was the limited number of instances or projects available, taking into account the characteristics of neural network models. Therefore, when enriching the dataset attributes through representation techniques based on natural language processing, it was necessary to restrict the number of new columns to be included. In this sense, the latent semantic indexing technique enables customization of the number of components in the generated representation vector, resulting in this case in four components or columns, which allowed the consolidation of a dataset with 10 columns. Likewise, since the “group” attribute was considered and is categorical in nature, it was necessary to map the initial seven categories of this column into three categories, so that when applying the one-hot encoding technique, only three additional columns were generated in the dataset, thus avoiding dimensionality-related issues.

Finally, it is worth noting that this study did not include a comparison with other simple machine learning models, considering that the primary objective was to leverage the capability of neural networks to model complex and nonlinear relationships among variables associated with project resources and their final outcomes. In this regard, within the context of project management, where interactions among factors are often multiple and not readily apparent, neural networks provide greater flexibility to capture high-level patterns without requiring prior assumptions about the form of such relationships, which, together with the favorable results obtained, supports their selection as the main methodological approach.

CONCLUSIONS

Given the level of expertise that may be required for project management under the PMBOK approach, tools based on predictive models constitute a fundamental support for cost and resource estimation, as well as for predicting project risk across different phases of the project life cycle. Thus, this article proposes, as its main contribution, a novel neural network–based model for predicting the success or failure of a project managed under PMBOK, using project characteristics such as the resources involved, the project group according to PMBOK, and the project indicators. Based on the excellent fitting achieved by the model when trained over 100 epochs, this model is intended to serve as a reference for use in project management environments adopting the PMBOK approach. Likewise, the methodology employed for the implementation, training, and evaluation of the model can be considered for the development of predictive models in the field of project management, particularly when incorporating relevant approaches such as the vectorized representation of descriptive project information.

With respect to state-of-the-art works in which classical machine learning models have been used for the estimation of relevant attributes across the different stages of a project life cycle, the proposed model exhibited consistent and competitive results, as well as excellent fitting capability throughout the training epochs, which suggests its strong usefulness in identifying the likelihood of project success or failure according to its specific characteristics. In this same sense, the proposed model presents, as a differentiating factor, not only the consideration of the PMBOK process groups, but also the inclusion of project indicators, which, although non-numerical, were represented in a vectorized form through the use of the TF-IDF technique in combination with latent semantic indexing (LSI). This enables the models to be trained not only on numerical data, but also on relevant information available in the descriptive attributes of the dataset.

At the level of the performance of the implemented, trained, and evaluated model, it is important to note that, in terms of the accuracy metric, after approximately 60–70 epochs this metric achieved performance above 95%. Likewise, regarding the precision, recall, and F1-score metrics, values of 100%, 95.24%, and 97.56%, respectively, were obtained, which suggests that the model exhibits robust performance and that, in addition to project resources and project group, the descriptive information contained in the “indicator” variable provides a significant information gain to the model fitting process. Thus, this work serves as a reference for the implementation of predictive models in the field of project management, taking into account the qualitative information inherent to each project or project phase.

Now, this research demonstrated the relevance of using and leveraging open-source libraries and technologies in the development of the different phases of the adopted methodology. Thus, for dataset cleaning and structuring, the advantages provided by the pandas and NumPy libraries were utilized. For the vector representation of qualitative attributes, as well as for data partitioning into training and test sets and the computation of evaluation metrics, the functions provided by the scikit-learn library were employed. Regarding the structuring, implementation, and training of the ANN model, the capabilities offered by the TensorFlow library were leveraged. Finally, for the visualization of performance curves and the graphical representation of the confusion matrix, the matplotlib and seaborn libraries were used. Collectively, these libraries can be considered suitable tools for the implementation, training, and evaluation of neural network–based models not only in the project management domain, but also in other application domains.

As future work derived from the present research, it is intended to propose a regression model for resource estimation that not only considers the ordinal and non-ordinal numerical attributes inherent to projects, but also incorporates the vector representation of project descriptive attributes through the integration of techniques such as TF-IDF combined with latent semantic indexing (LSI).

CRediT AUTHORSHIP CONTRIBUTION STATEMENT

J. Pinedo-Cabarcas: Conceptualization, Methodology, Investigation, Formal analysis, Writing – Original draft. **G. Chanchi-Golondrino**: Conceptualization, Methodology, Visualization, Writing – Review and editing. **M. Garcia-Bolaños**: Conceptualization, Methodology, Writing – Review and editing, Supervision.

FINANCING

The research project was developed at the Universidad de Cartagena by researchers Javier Antonio Pinedo Cabarcas, Gabriel Chanchi Golondrino, and Miguel Ángel García Bolaños, professors in the engineering faculty, which resulted in the article entitled “Proposal of a neural network–based model for predicting project success or failure under the PMBOK approach using resources and indicators” which was financed with the researchers' own resources and lasted 12 months in 2025.

REFERENCES

Abasova, J., Tanuska, P., & Rydzi, S. (2021). Big Data—Knowledge Discovery in Production Industry Data Storages—Implementation of Best Practices. *Applied Sciences*, *11*(16), 7648. https://doi.org/10.3390/app11167648

Abuhussain, M., Alhamami, A. H., Almazam, K., Humaidan, O., Bashir, F. M., & Dodo, Y. A. (2025). Integrating BIM, Machine Learning, and PMBOK for Green Project Management in Saudi Arabia: A Framework for Energy Efficiency and Environmental Impact Reduction. *Buildings*, *15*(17), 3031. https://doi.org/10.3390/buildings15173031

Alatawi, M. N., Alyahyan, S., Hussain, S., Alshammari, A., Aldaeej, A. A., Alali, I. K., & Alwageed, H. S. (2023). A Data-Driven Artificial Neural Network Approach to Software Project Risk Assessment. *IET Software*, *2023*(1). https://doi.org/10.1049/2023/4324783

Alfaifi, I. J., & Aksoy, P. M. S. (2023). Impact of Machine Learning on IT Project Management. *Journal of Image Processing and Intelligent Remote Sensing*, *41*, 31–38. https://doi.org/10.55529/jipirs.41.31.38

Alsugair, A. M., Al-Gahtani, K. S., Alsanabani, N. M., Alabduljabbar, A. A., & Almohsen, A. S. (2023). Artificial Neural Network Model to Predict Final Construction Contract Duration. *Applied Sciences*, *13*(14), 8078. https://doi.org/10.3390/app13148078

Basaran, Y., Aladag, H., & Isik, Z. (2025). Machine learning-based dynamic model for on-site subcontractor performance management. *Engineering, Construction and Architectural Management*. https://doi.org/10.1108/ECAM-11-2024-1563

Bogojevic, P. (2020). Project Management in Data Warehouse Implementations: A Literature Review. *IEEE Access*, *8*, 225902–225934. https://doi.org/10.1109/ACCESS.2020.3045072

Chola, K., & Chibomba, K. (2025). An Examination on Project Cost Overruns in Project Management: A Study of S3 Project at Kanshanshi Mine. *Journal of Arts, Humanities and Social Science*, *2*(1), 185–193. https://doi.org/10.69739/jahss.v2i1.417

Davidov, P., Ainbinder, I., Ayoubi, N., Avivine, B., & Eliyahu, L. (2023). Implementation of Project Management Knowledge Areas by Project Managers. *Journal of Engineering, Project, and Production Management*, *13*(2), 138–147. https://doi.org/10.32738/JEPPM-2023-0014

Fridgeirsson, T. V., Ingason, H. T., Jonasson, H. I., & Gunnarsdottir, H. (2023). A Qualitative Study on Artificial Intelligence and its Impact on the Project Schedule-, Cost- and Risk Management Knowledge Areas as presented in PMBOK®. *Journal of Research, Innovation and Technologies (JoRIT)*. https://doi.org/10.20944/preprints202309.0012.v1

Fridgeirsson, T. V., Ingason, H. T., Jonasson, H. I., & Jonsdottir, H. (2021). An Authoritative Study on the Near Future Effect of Artificial Intelligence on Project Management Knowledge Areas. *Sustainability*, *13*(4), 2345. https://doi.org/10.3390/su13042345

Gabel, M., Sujka, M., Davis, Z. W., & Keizur, A. E. (2023). Performance of Risk-Based Estimating for Capital Projects.

Transportation Research Record: Journal of the Transportation Research Board, *2677*(1), 1059–1070. https://doi.org/10.1177/03611981221103238
García, H. H. S., Alejandrez, C. V., Vite, R. U. C., & González, F. G. (2024). Metodología Project Management Institute en modelos educativos. Caso de estudio: Academia de Edificación del programa educativo de Ingeniería Civil de la Universidad Tecnológica de la Huasteca Hidalguense. *Región Cientíica*, *3*(1), 2024269.
Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. https://www.deeplearningbook.org
Henderson, L. S., Stackman, R. W., & Koh, C. Y. (2013). Women project managers: the exploration of their job challenges and issue selling behaviors. *International Journal of Managing Projects in Business*, *6*(4), 761–791. https://doi.org/10.1108/IJMPB-06-2012-0033
Huda, M., Soepriyono, S., Siswoyo, S., & Azizah, S. (2019). Implementation of PMBOK 5th Standard to Improve the Performance and Competitiveness of Contractor Companies. *Proceedings of the Proceedings of The 2nd International Conference On Advance And Scientific Innovation, ICASI 2019, 18 July, Banda Aceh, Indonesia*. https://doi.org/10.4108/eai.18-7-2019.2288604
Kinsella, S. M. (2002). Activity-Based Costing: Does it Warrant Inclusion in a Guide to the Project Management Body of Knowledge (PMBOK® Guide)? *Project Management Journal*, *33*(2), 49–56. https://doi.org/10.1177/875697280203300207
Kunanbayev, K., Temirbek, I., & Zollanvari, A. (2021). Complex Encoding. *2021 International Joint Conference on Neural Networks (IJCNN)*, 1–6. https://doi.org/10.1109/IJCNN52387.2021.9534094
Lan, D., Tian, A., Wang, Y., & Li, Y. (2021). An overview of the principle, algorithm improvement and application based on the theory of latent semantic indexing. *Academic Journal of Computing & Information Science*, *4*(5). https://doi.org/10.25236/AJCIS.2021.040510
Landauer, T., & Dumais, S. (2008). Latent semantic analysis. *Scholarpedia*, *3*(11), 4356. https://doi.org/10.4249/scholarpedia.4356
Martinez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernandez-Orallo, J., Kull, M., Lachiche, N., Ramirez-Quintana, M. J., & Flach, P. (2021). CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories. *IEEE Transactions on Knowledge and Data Engineering*, *33*(8), 3048–3061. https://doi.org/10.1109/TKDE.2019.2962680
Narbaev, T., Hazir, Ö., Khamitova, B., & Talgat, S. (2024). A machine learning study to improve the reliability of project cost estimates. *International Journal of Production Research*, *62*(12), 4372–4388. https://doi.org/10.1080/00207543.2023.2262051
Nishat, M. M., Ahsan, A., & Olsson, N. O. E. (2025). Applying Machine Learning for Predictive Analysis in Project-Based Data: Insights into Variation Orders. *Journal of Information Technology in Construction*, *30*, 807–825. https://doi.org/10.36680/j.itcon.2025.033
Poslavskaya, E., & Korolev, A. (2023). *Encoding categorical data: Is there yet anything "hotter" than one-hot encoding?* https://doi.org/https://doi.org/10.48550/arXiv.2312.16930
Reznikov, R. (2025). Enhancing Project Management Success through Artificial Intelligence. *International Journal of Multidisciplinary Research and Growth Evaluation*, *6*(1), 1036–1046. https://doi.org/10.54660/.IJMRGE.2025.6.1.1036-1046
Rosenberger, P., & Tick, J. (2021). Multivariate Optimization of PMBOK, Version 6 Project Process Relevance. *Acta Polytechnica Hungarica*, *18*(11), 9–28. https://doi.org/10.12700/APH.18.11.2021.11.2
Saltz, J. S., & Krasteva, I. (2022). Current approaches for executing big data science projects—a systematic literature review. *PeerJ Computer Science*, *8*, e862. https://doi.org/10.7717/peerj-cs.862
Santos, F. R. S. dos, & Cabral, S. (2008). FMEA and PMBOK Applied to Project Risk Management. *Journal of Information Systems and Technology Management*, *5*(2), 347–364. https://doi.org/10.4301/s1807-17752008000200008
Schröer, C., Kruse, F., & Gómez, J. M. (2021). A Systematic Literature Review on Applying CRISP-DM Process Model. *Procedia Computer Science*, *181*, 526–534. https://doi.org/10.1016/j.procs.2021.01.199
Shimaoka, A. M., Ferreira, R. C., & Goldman, A. (2025). Integrating CRISP-DM and eXtreme Programming in Agile Data Science Projects. *European Journal of Electrical Engineering and Computer Science*, *9*(4), 1–7. https://doi.org/10.24018/ejece.2025.9.4.739
Simonaitis, A., Daukšys, M., & Mockienė, J. (2023). A Comparison of the Project Management Methodologies PRINCE2 and PMBOK in Managing Repetitive Construction Projects. *Buildings*, *13*(7). https://doi.org/10.3390/buildings13071796
SLAEAT, F. (2024). The impact of poor planning on the duration and cost of the projects. *Mesopotamian Journal of Civil Engineering*, *2024*, 60–81. https://doi.org/10.58496/MJCE/2024/009
Taye, G. D., & Feleke, Y. A. (2022). Prediction of failures in the project management knowledge areas using a machine learning approach for software companies. *SN Applied Sciences*, *4*(6), 165. https://doi.org/10.1007/s42452-022-05051-7
Uddin, S., Ong, S., & Lu, H. (2022). Machine learning in project analytics: a data-driven framework and case study. *Scientific Reports*, *12*(1), 15252. https://doi.org/10.1038/s41598-022-19728-x
Wells, H. (2012). How Effective Are Project Management Methodologies? An Explorative Evaluation of Their Benefits

in Practice. *Project Management Journal*, *43*(6), 43–58. https://doi.org/10.1002/pmj.21302
Xu, Z., Yu, J., & Li, H. (2014). Analyzing Integrated Cost-Schedule Risk for Complex Product Systems R&D Projects. *Journal of Applied Mathematics*, *2014*, 1–10. https://doi.org/10.1155/2014/472640