SSRG International Journal of Computer Science and Engineering
ISSN: 2348–8387 / https://doi.org/10.14445/23488387/IJCSE-V10I12P103

Volume 10 Issue 12, 10-16, December 2023

Original Article

Automated Project Selection System in Software Engineering Based on Neural Network Modeling

Amirali Kerimovs

Independent Researcher, Riga, Latvia.

Corresponding Author : amir.academicinquiry@gmail.com

Received: 15 October 2023 Revised: 25 November 2023 Accepted: 10 December 2023 Published: 30 December 2023

Abstract - *The modern software engineering industry is confronted with a constantly growing volume of projects and information. Selecting the most suitable projects for execution is becoming an increasingly complex task. The article examines the methodology of enterprise project management from the perspective of applying the concept of neural network modelling and formulating the critical principles of applying this concept to project management. This paper proposes to develop and investigate an automated project selection system in the field of software engineering, utilizing neural network modelling methods. Our objective is to create an intelligent system capable of analyzing project characteristics and predicting their potential success based on trained neural network models. We suggest conducting an extensive analysis of existing methods and approaches to project selection, followed by the development and training of a neural network model using a collected dataset. It is anticipated that the created system can significantly enhance project selection efficiency and increase the likelihood of successful implementation in the realm of Software Engineering.*

Keywords - *Fuzzy logic, Neural network, Project management, Process approach, Software engineering, Uncertainty.*

1. Introduction

The project management process within a high-tech enterprise is a system comprised of interconnected and mutually coordinated elements: tasks and events [6]. The process-oriented approach views a project as a series of actions aimed at transitioning the management object from one state to another. Project management is further characterized by the presence of risks and uncertainties, along with a range of possible events at each implementation stage. Each event carries a specific probability of occurrence and an associated degree of impact, evaluated based on specific criteria [7].

2. Literature Review

In the context of uncertainty, effective management of high-tech production projects should be conducted by utilizing the latest advancements in decision-making theory and artificial intelligence systems, with a specific focus on the concept of artificial neural networks. Artificial neural networks represent mathematical tools capable of reproducing complex nonlinear functional relationships, incorporating elements of fuzzy logic such as working with fuzzy sets (initially pioneered by Lotfi Zadeh at the University of California in 1965) [9]. The exploration of economic and mathematical models based on fuzzy set theory was pioneered by Professor K. F. Kovalchuk [8], initiating scientific research in this field. Given our topic, the application of these concepts holds significant potential for enhancing project management practices in the high-tech industry.

The purpose of the article is:

- Development of a project implementation process model based on the application of fuzzy logic and the concept of neural networks;
- Construction of an algorithm, which will be expediently used in modelling the process of making management decisions in project implementation.

3. Presentation of the Main Material

The distinct feature of forecasting economic indicators' values over time through the integration of fuzzy logic and Neural networks lie in obtaining forecasts by formulating regression equations in a fuzzy format, utilizing linguistic variables [1].

For instance, considering the linguistic variable "Falling production volumes," it becomes feasible to establish a range of potential values (defined function area): {Insignificant, Significant, Catastrophic, etc.} [2]. This framework enables the analysis of indicators and the generation of forecasts while adhering to a set of predefined rules, including logical statements like "if, then; and, or."

In a neuro-fuzzy system, the analysis of situations and management decision-making relies on a structured knowledge base formed from previously collected project indicator values within a specific timeframe.

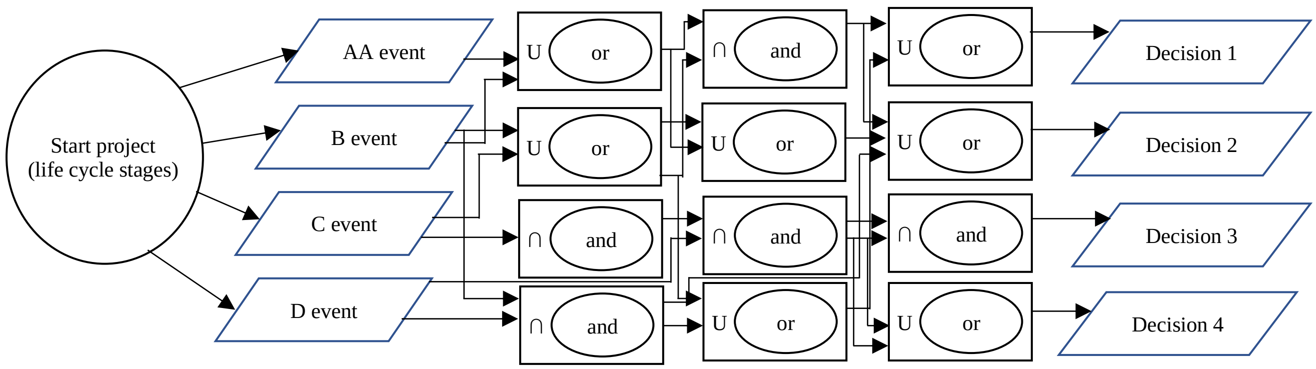

Fig. 1 Schematic representation of the process of making a management decision with certain combinations of events

The fundamental principles for constructing the knowledge base in neuro-fuzzy systems involve replicating the training selections within the knowledge base, formulating rules for various event scenarios, optimizing the rule count, simplifying rules while maintaining their logical essence, and establishing intricate rules [3]. To illustrate, a simplified schematic representation of the decision-making process based on established rules, adapted for ease, occurs within a three-layer neural network (depicted in **Figure 1**). A representative logical formula of one such rule (highlighted in bold), pivotal for the system's functioning, is presented (1). In the context of our project selection theme, these methodologies offer promising avenues for refining the decision-making process and enhancing project outcomes.

$$IF(AUB) \cap (C)U(D)THENSolution1 \tag{1}$$

Where *(AUB)* is a logical operation on the 1st layer of neurons; ∩ *(C)* – is a logical operation on the 2nd layer of neurons; *U (D)* is a logical operation on the 3rd layer of neurons.

The trajectory of impulse propagation relies on neuron characteristics developed during the neural network's learning process based on a pre-existing knowledge base [6, p. 40–50]. From our standpoint, the primary objectives of employing neural network modelling in project management are to leverage predictive capabilities for identifying potential issues during the earliest phases of project execution and conducting a preliminary assessment of their potential impact on the project's ultimate outcomes.

Simultaneously, it remains crucial to anticipate any additional resources that may be required for the project's timely and high-quality completion, taking potential risks into account.

Among the most prevalent parameters utilized for monitoring project implementation progress in management literature are comprehensive indicators. These parameters serve as evaluative tools for assessing the project's execution, both upon its culmination and during intermediate phases of development:

1. Project budget: representing the financial resources essential for executing the entire project or specific stages thereof.
2. Time: an invaluable resource, irreplaceable once lost.
3. Project outcomes: encompassing specific indicator values used as criteria for gauging the attainment of project objectives, which can differ across projects (e.g., a 5% increase in sales, a 10% rise in production profitability, etc.).
4. Quality: indicative of the final product's level of quality, aligning with the project's ultimate goal.
5. Risk: reflecting the probability of unforeseen adverse events occurring throughout project execution or at distinct stages [5].

In the context of our project selection theme, the integration of neural network modelling allows for enhanced foresight into potential challenges, resource allocation, and risk management, ultimately contributing to more effective project management.

The synthesis of these structures can be seamlessly executed when the criterion for compatibility revolves around the method and structure of information transmission within the information system. As each component is intricately linked through an intricate network of information flows with numerous other elements, the resultant intricate network exhibits substantial potential in addressing defined objectives concerning the efficient exploration and optimization of pathways to attain established goals. Moreover, this framework facilitates the evaluation of alternative solutions in terms of their effectiveness and establishes a hierarchy of preferred solutions and potential developments. By adjusting neuron parameters, targeted responses to recurring scenarios in the economic system can be established. Upon practical implementation and establishing an extensive array of interconnected elements within a neural network (as depicted in Fig. 2), a comprehensive decision-making system can be developed, taking into account all potential combinations of economic phenomenon parameters [4]. This aligns seamlessly with our theme of utilizing neural network modelling for project selection and management optimization within the realm of Software Engineering.

Table 1. Matrix foralyzing the project implementation process based on performance evaluation resource usage

	Indicator	Stages of the project									
		Stage 1	**Stage 2**	**Stage 3**	**….**	**….**	**….**	**….**	**….**	**….**	**Stage N**
Software efficiency budget	Plan	100	100	100	100	100	100	100	100	100	100
	Fact *	*150*	*120*	*90*	*70*	*110*	*130*	*80*	*95*	*105*	*97*
	Efficiency in %	-33.33	-16.67	11,11	42.86	-9.09	-23.08	25.00	5.26	-4.76	3.09
	Linguistic meaning	1	1	2	4	1	1	4	2	1	2
Software efficiency time	Plan	100	100	100	100	100	100	100	100	100	100
	Fact *	*60*	*107*	*92*	*68*	*85*	*45*	*88*	*91*	*103*	*72*
	Efficiency in %	66,67	-6.54	8.70	47.06	17.65	122.22	13.64	9.89	-2.91	38.89
	Linguistic meaning	4	1	2	4	3	4	3	2	1	4
Software efficiency results	Plan	100	100	100	100	100	100	100	100	100	100
	Fact *	*75*	*108*	*120*	*170*	*80*	*74*	*125*	*130*	*82*	*66*
	Efficiency in %	33,33	-7.41	-16.67	-41.18	25.00	35,14	-20.00	-23.08	21.95	51,52
	Linguistic meaning	4	1	3	1	4	4	1	1	4	4
Software efficiency quality	Plan	100	100	100	100	100	100	100	100	100	100
	Fact *	*120*	*75*	*88*	*105*	*115*	*108*	*133*	*205*	*64*	*81*
	Efficiency in %	-16.67	33,33	13.64	-4.76	-13.04	-7.41	-24.81	-51.22	56.25	23.46
	Linguistic meaning	1	4	3	1	1	1	1	1	4	4
Software efficiency risk	Plan	100	100	100	100	100	100	100	100	100	100
	Fact *	*63*	*122*	*87*	*67*	*84*	*107*	*135*	*170*	*55*	*44*
	Efficiency in %	58.73	-18.03	14.94	49.25	19.05	-6.54	-25.93	-41.18	81,82	127.27
	Linguistic meaning	4	1	3	4	3	1	1	1	4	4

** Numerical data given in the " Fact" lines are not data taken from a real project; these data are generated by the method of random numbers and are presented in the table for a visual demonstration of the process of applying the method of calculation given in the article.*

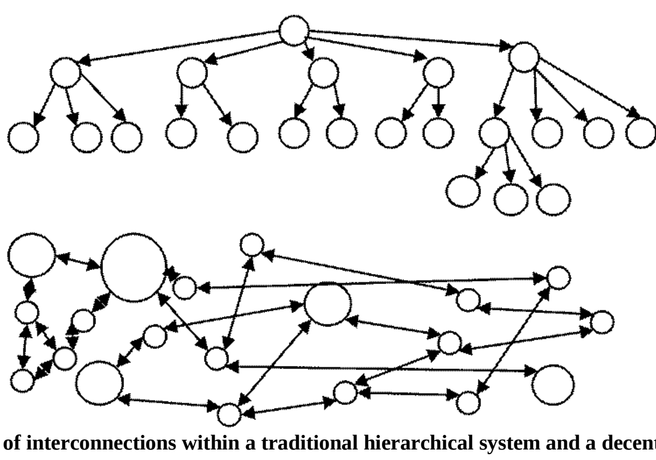

Fig. 2 Visual depiction of interconnections within a traditional hierarchical system and a decentralized neural network

The neural network oprates within two distinct modes: the learning mode and the task execution mode. The learning mode entails the deliberate shaping of predefined responses from the entire system or specific localized segments in response to certain informational cues. During this process, interrelations and absolute parameter values among neurons are established. Conversely, the task execution mode involves the system processing specific information based on parameters established during prior "training" [1].

Furthermore, an additional noteworthy facet concerning neural network modelling in project management is the multifaceted nature of the neural network concept. It encompasses at least two meanings in this context. Firstly, it directly refers to the application of neural network modelling for simulating project execution, utilizing input values, desired outcomes, and constraints. This notion resonates harmoniously with our theme of employing neural network modelling for optimizing project selection and management practices within the realm of Software Engineering.

Our proposed model for evaluating project performance takes into consideration the five parameters outlined earlier. To illustrate the calculation process for our model, input data have been provided in specific rows within the table.

1. Efficiency, as per the classical definition, signifies the ratio of resources expended to achieve a particular outcome in relation to the outcome itself [4]. In accordance with our methodology, project stage efficiency is determined by evaluating the ratio between the anticipated resource allocation and the actual resource expenditure. This assessment is conducted using expert opinions, with the entire planned resource allocation set as the baseline "100." Therefore, if the actual volume, for instance, is "80," it indicates that only 80% of the designated initial resources were utilized. The efficiency calculation formula is presented in equation (2).

$$Eshout. = \left((Eplan./E.!) - 1\right) x 100\% \qquad (2)$$

where: $E_{vyk.}$ - the efficiency of the implementation of the project stage; *It's* $_{a\ plan.}$ – the planned volume of resource expenditure; *It's* $_{a\ fact.}$ – the actual amount of resources spent.

4. Results and their Discussion

In order to further enter the values of efficiency indicators into the neural network model and use the fuzzy logic apparatus, it is suggested to assign linguistic variables to the indicators: if the indicator is less than " 0 " - "negative efficiency " (more is spent than planned) - code "1"; 0-9% - " low efficiency " - code "2"; 10-19% - " average efficiency " - code "3"; 20% and more - " high efficiency " - code "4" (Table 1).

Table 2. Input data for creating a project management process model

№	Project stage	Project performance evaluation parameters				
		x_1	**x_2**	**x_3**	**x_4**	**x_5**
		Efficiency on a budget	**Time efficiency**	**Efficiency by results**	**Efficiency by quality**	**Performance by risk**
1	Stage 1	1	4	4	1	4
2	Stage 2	1	1	1	4	1
3	Stage 3	2	2	3	3	3
4	……..	4	4	1	1	4
5	……..	1	3	4	1	3
6	……..	1	4	4	1	1
7	……..	4	3	1	1	1
8	……..	2	2	1	1	1
9	……..	1	1	4	4	4
10	Stage N	2	4	4	4	4
11	Average effectiveness of the project	2	3	3	2	3

The input array for creating a neural network model should be formed on the basis of linguistic variables from the table. 1, as shown in the table. 2.

The graphic representation of the linguistic values of the evaluation parameters and their comparison at different stages of project execution will look as follows (Fig. 2). Thus, it can be noted that:

$$Y = f(x1, x2, x3, x4, x5) \qquad (3)$$

Where: Y is the total final efficiency of the project; x_1 – efficiency according to the budget; x_2 – time efficiency; x_3 – efficiency according to results; x_4 – efficiency in terms of quality; x_5 – efficiency by risk.

The primary guidelines for constructing three categories of fuzzy rules, linking the error 'e' and its derivative 'e'' with three coefficients K_p, K_i, and K_d [4,5], are as follows:

1. When 'e' is considerably large, increasing K_p and decreasing K_d are recommended to reduce 'e,' alongside excluding the integral effect.
2. If both 'e' and 'e'' are within permissible limits, reducing K_p to a necessary level is advisable to mitigate overregulation and minimize system impact.
3. In the case of 'e' being very small, enhancing K_p and K_i is necessary for system stability. Adjustments to K_d should prevent oscillations within the system. When 'e' is small, an increase in Kd is employed; when 'e'' is large, K_d d is decreased.

The foundational production rules for coefficients K_p, K_i, and K_d are provided in tables 1-3. Fuzzy rules from the knowledge base governing the specification of proportional coefficient K_p, integral coefficient K_i, and differential coefficient K_d take the form:

R_{1p}: If 'e' is *NB*, 'e'' is *NB*, then K_p is *PB*, K_i is *PS*, K_d is *NB*,

....,

R_{49p}: If 'e' is *PB*, 'e'' is *PB*, then K_p is *NB*, K_i is *PB*, K_d is *PB*.

The rule base was established within the FIS editor, integrating a fuzzy inference system through the utilization of the Mamdani algorithm.

Figure 3 displays a closed-loop control system created in the Simulink dynamic modelling package within the MATLAB environment.

This system incorporates a fuzzy adaptation block, a configurable PID controller, and a control object featuring a transfer function (as depicted in Fig. 1). The initial coefficient values were set as follows K_e =1, K_{de} =1, K_{p0} =1, K_{i0} = 1, K_{d0} = 1.

Table 3. Base of production rules of coefficient Kp

	N.B	**NM**	**NS**	**Z**	**PS**	**PM**	**PB**
N.B	PB	PB	PM	PM	PS	Z	Z
NM	PB	PB	PM	PS	PS	Z	NS
NS	PM	PM	PM	PS	Z	NS	NS
Z	PM	PM	PS	Z	NS	NM	NM
PS	PS	PS	Z	NS	NS	NM	NM
PM	PS	Z	NS	NM	NM	NM	N.B
PB	Z	Z	NM	NM	NM	N.B	N.B

Table 4. Base of production rules of coefficient K_i

	N.B	**NM**	**NS**	**Z**	**PS**	**PM**	**PB**
N.B	PS	NS	N.B	N.B	N.B	NM	PS
NM	PS	NS	N.B	NM	NM	NS	Z
NS	Z	NS	NM	NM	NS	NS	Z
Z	Z	NS	NS	NS	NS	NS	Z
PS	Z	Z	Z	Z	Z	Z	Z
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PS	PS	PS	PB

Based on the above, the task for the neural network to be created will be to find the dependence parameters " f" between the input array " (x 1, x 2, x 3, x 4, x 5)" and the output array "Y".

The process of creating a neural network model based on the previously calculated data involves the following steps, considering our project selection theme:

1. Utilize the input variables (x) and the outcome (Y) to formulate a model designed to effectively predict the project's success or failure with a high probability, focusing on resource utilization efficiency. This modelling is based on analyzing the initial 30-40% of project tasks.
2. Establish recommendations concerning acceptable thresholds for efficiency parameter values (denoted as "x") during the early stages. These thresholds indicate the points at which successful project implementation becomes highly probable, given the available resources.

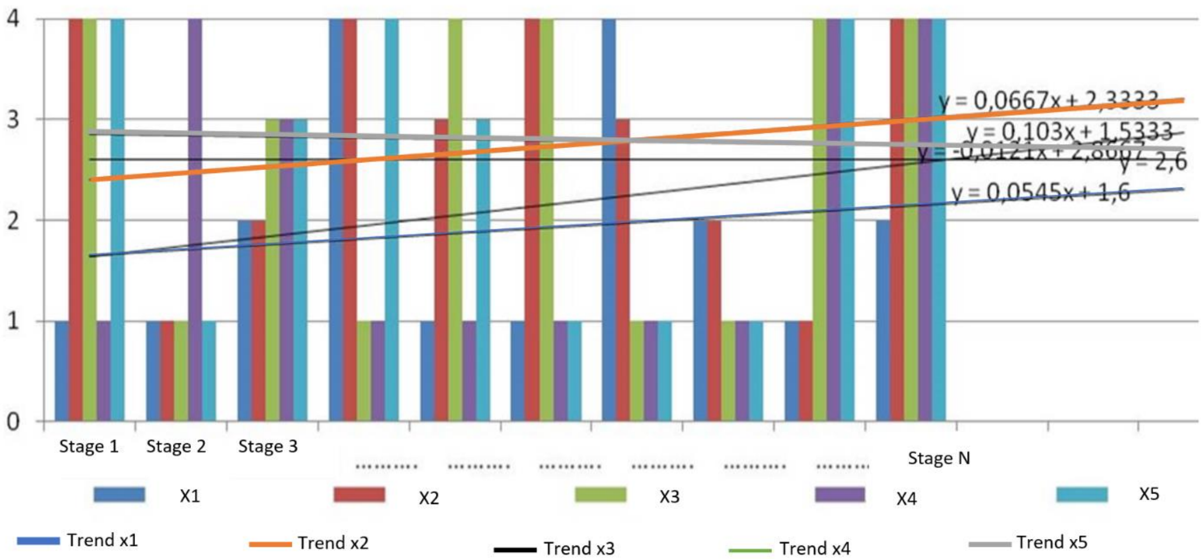

Fig. 3 Graphic display of the linguistic values of the evaluation parameters of the project execution process

3. Based on the insights gained from the preceding steps, assess whether the project should be continued under the current conditions and whether its continuation aligns with the enterprise's goals, ensuring that undue losses are avoided.

This algorithm outlines a systematic approach for developing a neural network model that aids in the decision-making process regarding project selection and continuation, ultimately enhancing the efficiency and success rate of projects in the field of Software Engineering.

5. Conclusion

1. Drawing insights from literature sources, the utilization of information technologies in project management, incorporating fuzzy logic and neural networks, is explored. An innovative approach involves constructing a neural network model for the project implementation process, enhancing the effectiveness assessment methodology. This approach integrates project stages, acknowledges resource constraints, and introduces linguistic parameters for performance evaluation.

2. The culmination of the conducted research has yielded an enhanced methodology suitable for advancing the neural network model of project implementation. This methodology is particularly valuable for analyzing and forecasting the implementation process during its initial stages. The text further provides a schematic representation of how indicators are computed, and an input data array is constructed for the creation of the neural network model.

3. Future research prospects encompass the development of a neural network model tailored for projecting the project implementation process. This advancement hinges on analyzing the correlation between input and output parameter arrays, as outlined in Table 2.

The method, derived from the improved efficacy assessment method found in sources [4], holds potential for integration into the creation of a comprehensive neural network model for project implementation. This endeavour aligns with our theme of utilizing neural network modelling for project selection and management optimization in Software Engineering.

References

[1] Pedro José Bernalte Sánchez, Fausto Pedro García Márquez, and Mayorkinos Papaelias, "Life Cycle Sustainability Assessment of ENDURUNS Project: Autonomous Marine Vehicles," *International Conference on Management Science and Engineering Management (ICMSEM 2023)*, vol. 409, pp. 1-11, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] Illia Gorbatiuk, "Peculiarities of Project Management in the Context of Software Engineering," *Scientific Papers of Berdiansk State Pedagogical University Series Pedagogical Sciences*, no. 3, pp. 267-274, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Kalinina Aksinya Vasilievna, and Petrochenko Marina Vyacheslavovna, "An Integrated Approach to Assessing the Life Cycle of Construction at the Design Stage Using Software Complexes," *Construction: Science and Education*, vol. 12, no. 1, pp. 88-100, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Vera Lyubchenko, "About Some Aspects of the Engineering of Software Systems with Elements of Artificial Intelligence," *Methods and Means of Software Engineering*, no. 3-4, pp. 99-106, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Christine Lucas-Lamouroux et al., "Model-Based System Engineering, An Industrialization Path For Decommissioning Projects by Assystem," *EPJ Nuclear Sciences and Technologies*, vol. 9, no. 24, pp. 1-7, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] G. Mirskikh et al., "Life Cycle of the Software of Engineering Objects," *Energy and Automation*, no. 5, pp. 115-128, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Andrew Quansah et al., "Requirement Engineering Problems Impacting the Quality of Software in Sub-Saharan Africa," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 1, pp. 350-355, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Svetlana's Accident, and Ivanova Olga, "The Choice of Software for the Implementation of Projects Based on Information Modelling Technologies," *Construction and Architecture*, vol. 11, no. 3, pp. 100-113, 2023. [CrossRef] [Publisher Link]

[9] Tatiana Shulga, and Khramov Dmitrii, "Life Cycle Ontology of Software Engineering," *Vestnik of Astrakhan State Technical University Series: Management, Computer Sciences and Informatics*, no. 2, pp. 66-74, 2023. [CrossRef] [Google Scholar] [Publisher Link]