International Journal of Civil Engineering and Technology (IJCIET)
Volume 9, Issue 12, December 2018, pp. 1094–1102, Article ID: IJCIET_09_12_113
Available online at http://www.iaeme.com/ijciet/issues.asp?JType=IJCIET&VType=9&IType=12
ISSN Print: 0976-6308 and ISSN Online: 0976-6316

INTEGRATED SOFTWARE PROJECT RISKS METHOD BASED ON PDF-ANN TECHNIQES

Huda Abdulaali Abdulbaqi, Asmaa Sadiq Abdul Jabar, Zinah S.Abdul Jabar
Computer Science Department, Collage of Science, Mustansiriyah university

ABSTRACT

This paper aims to attain high accuracy of risk assessment during monitoring and evaluation of software risk factors by reconstruction of the pattern for risk recognition and estimation, the research helped construct a functional model for risk assessment. The reconstruction is achieved by integrating Artificial Neural Network, one of today's best modeling techniques with two stage of Probability Density Function (PDF), This brought forward a model for predictions using historical data and taking into account other important common risk factors. The research showed that the employed can be utilized effectively for the identification of risk effects in the entire phases of software development project. We categorized the top risks into human skills, knowledge, the experience of the staff level and difficulties in applying software management and organization presented as software cost risks in the total risk evaluation. Artificial Neural Network ANN represents one of the promising modeling techniques for multi objective software risks such as: cost, scheduling, quality and other business risks.

Key words: Project risk, risk Management, Monitoring software, Artificial Neural Network, PDF

Cite this Article: Huda Abdulaali Abdulbaqi, Asmaa Sadiq Abdul Jabar, Zinah S.Abdul Jabar, Integrated Software Project Risks Method Based on PDF-ANN Techniqes, *International Journal of Civil Engineering and Technology (IJCIET)* 9(12), 2018, pp. 1094–1102.
http://www.iaeme.com/IJCIET/issues.asp?JType=IJCIET&VType=9&IType=12

1. INTRODUCTION

Risk management is referred to as process that identifies, assesses and controls threats to enhance capital and earnings of organizations. Risks may originate from various sources like erroneous strategic management, accidents and natural crises. In this regard, the development of risk management life cycle is a process that brings about enhanced efficiency and flexibility of operations1. Prior literature made a list of the software risk factors by rank based on their importance and number of occurrence in terms of data source. They then presented such factors and risk management methods to respondents and found that all software projects risks were significant in the perspective of software project managers. Moreover, the ranking based on risks importance is made in light of analysis, planning, maintenance, design and implementation [2,3]. In order to analyze the risks economically, collecting the risk factors in

groups will be the primary target. A scientific way of approaching risks is to classify them based on risk attributes. Development of risk management software can be classified into scheduling risks and quality risks. Also, it can be grouped into performance risks, cost risks support risks and schedule risks. These risks are gotten through studies and experiences in projects [2]. More importantly, the top ten software risk [4]. Some authors made used of the risks to develop an application for risks identification during software development and for predicting project successes or failure in the form of Artificial Neural Networks[5,6] While, other researchers develop some techniques such as Singular Value Decomposition SVD[7]. he present work combines Artificial Neural Networks with Singular Value Decomposition SVD method for risk control and these encapsulate processes, methods and tools of risk management prior to the risks becoming significant problems. The rest of the paper is organized as follows: Risk Management Methods in next section 2, next section 3 explains the Selection of Risk Factors, section 4 explains in detail about Artificial Neural Networks (ANNs), next section 5 Risk Factors Evaluation, then in section 6 Experiment Results and finally Conclusion in Section 7.

2. RISK MANAGEMENT METHODS

Projects generally have risks, high-tech projects have compounded ones owing to their unique differences. If ignored, risks can lead to below at par product outcome, unsatisfied customer expectations, and anxious and conflicted staff during project completion, and ultimately these will minimize productivity. The general organization perspectives, conceptually, the risk occurs when the organization meets an uncertainties from limited costs and capacity in its activities for opportunities. The effective risk management must couple the risk strategies with the project initiate to lowest the stress and cost that produced by risk issues [8,9].

More specifically, risk analysis indicates the risk occurrence details (when, where and why), using direct stakeholders' feedback regarding the potential and impact of risk elements. It investigates the way project results and objectives might change owing to the risk effects. In addition, risk analysis is a risk management component used in projects, IT, security issues and processes wherein risks can be analyzed by qualitative or quantitative means[10,11]

The quantitative risk analysis simulates each critical risk effect and the overall costs are examined. This was done by correcting individual sub-market diagonal block. Similarly Elzamly in 2014 present a new method for risk assessment using quantitative evaluation and mining method in order to compare the risk of software project life cycle [4]. Hojjati and Noudehi in2015 applied the Monti Carlo simulation in assessing risks. They evaluated project risks in Information Technology field and utilized the Primavera Risk Analysis software in order to analyze and quantify the risks management factors [10]. Paraschivescu in 2016 integrated quality and risk management concepts. He sheds the light on new perspectives and dimensions [10]. Gandhi in 2014 applied the Artificial Neural Network (ANN) in risk management. He predict the level of risks in software life cycle project. The method observe a prior risks in project implementation to ensure the successful of project [3]. Also, Elzamly in 2016 proposed an identification of software risks control method [4]. The study classify and rank the risk factors based on the risk factor importance due to possibility of data occurred [4]. Andreas in 2017 analyze a survey data using risk assessment practice effectiveness. The study developed risk context of product development (PD) for forecasting the project outcomes [4]. The study observed the relationship between the risk management factors and the successful project.

Also, Kulkarni present a review in using ANNs. They suggest that the ANN's is useful when the input representation information was inadequate. The research present the fact that all the investigators apply feed forward propagation network type. They found that ANN's

development needs a hybrid modeling in association with other machine learning tools such as genetic programming and support vector machines were much useful5. Salman in 2018 proposed Singular Value Decomposition (SVD) correlated with the traditional risk factor calculations to estimate the software maintenance projects[7].

3. SELECTION OF RISK FACTORS

Risk management refers to a process of identifying risks and calculating their impact in order to develop strategies to keep them under control. The steps taken for risk management process are as follows;

The first step is risk identification, where the entire potential risks affecting the project development are determined and included in a check list[6]. This is followed by risk analysis[12], where the risk occurrence in terms of when, where and why are identified and understood through obtaining stakeholders' feedback on their probability and impact[13].The third step is planning and in this stage, a strategy is created to mitigate the risk impact and proper steps are taken for risk management[11] The next step is monitoring to steer clear of or to mitigate risks and to ultimately,

keep it under control [14] The last step is control and in this step, the actions that are planned in the prior steps are carried out with the occurrence of a risk[12].

In this regard, Elzamly in 2016 listed the top 50 software risk factors in software development lifecycle that have been commonly cited in the literature review [15]. Several authors have also made such a list, where their contents differ from one author to the other to some extent. However, there are some common software risk factors that are listed in majority of lists and they need to be addressed and controlled. More specifically, the list contains the 10 topmost critical risks in software project from the first to the tenth rank, with risk status and a resolution plan for each risk. Hood and Rashid present a new classification to the risk management in order to specify the relations between the different factors of risk. They organized a tree risk factors structured as probabilistic calculations. The analysis provide a good results to the qualitative and quantitative of risk failure. Also it facilitate the risk management processes [2]. For that this classification is used in this paper to develop an effective Software Life Cycle as shown in Figure 1.

Table I presents the software risk factors cited and used in the field of software development.

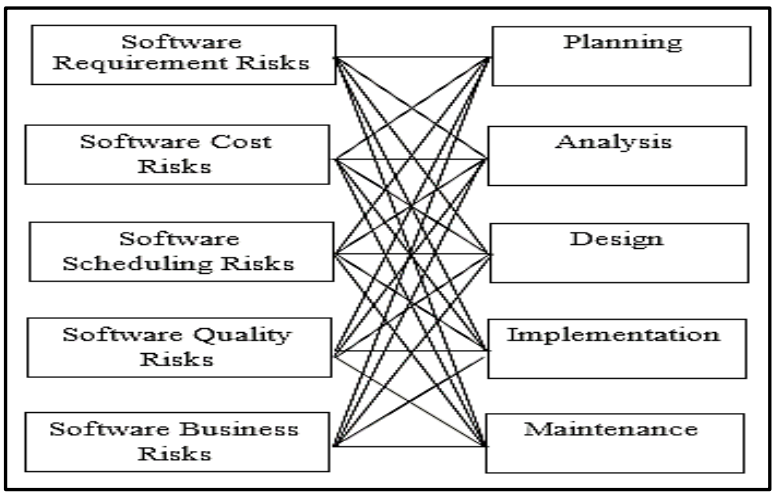


Figure 1 The project scheme

4. ARTIFICIAL NEURAL NETWORKS (ANNS)

A neural network refers to a parallel distributed processor comprising of processing units that form what is akin to a biological nervous system consisting of the brain, within which the information processing takes place [16,17]. Added to this, ANN has many layers of computing elements referred to as nodes, with each node receiving input signal from others like it or from external inputs, and following signal processing, the node outputs a transformed signal to other nodes, making up the final outcome. The first layer, also known as the input layer is the receptor of external information, whereas the last layer, also known as the output layer, is where the network generates the solution. The layers in between are also significant to ANNs identification of the data's complex patterns, with the transfer function transforming input to output signals[18]. ANNs are primarily used to learn how tasks are done on the basis of data provided for training, and to generalize reasonable outputs for inputs that it has not handled before[19]. The present method will be used to investigate the risk effect based on the presented risk factors in all Software Project Life Cycle.

5. EVALUATIONOF RISK FACTOR

The present method developed a questionnaire that comprised a list of questions related to chosen 34 risk factors. The selected factors represent the risks in software maintenance adopted from Loez and Salmeron [7]. The chosen risk factor questions prepared based on two types of questions, positive and negative [20, 21, 22]. The numerical value of these questions pave the way to numerical evaluation. The study samples comprising 250 persons worked in IT organization in Iraq. The methodology for evaluating the questionnaire results as in below:

Step 1: for each questionnaire, the boundary condition is represented as:

$$Q = \begin{cases} Positive\ Questios & \begin{matrix} if\ yes=1 \\ if\ no=0 \end{matrix} \\ Negative\ Questions & \begin{matrix} if\ yes=0 \\ if\ no=1 \end{matrix} \end{cases}$$

The value of each question (Q) that collected data will be used to determine the commonly occurring risks as in the next step.

$$RF = \sum_{i=1}^{n} (Q_i) \qquad (1)$$

where RF is the risk factor query answers for n questions.

Step 2: apply the values of each query questions (Q) in formal of Probability Density Function (PDF) in order to find the density of probability. The general form of PDF is:

$$\Pr[a \le X \le b] = \int_a^b f_X(x)dx \qquad (2)$$

where Pr is the Probability Density Function, X is continuous random variable which define the functionf_(X) (x) in (a→b) domain. The researchers applied the functions on all queries in all the selected companies and the answers density representation provided by the PDF as shown in figure 2.

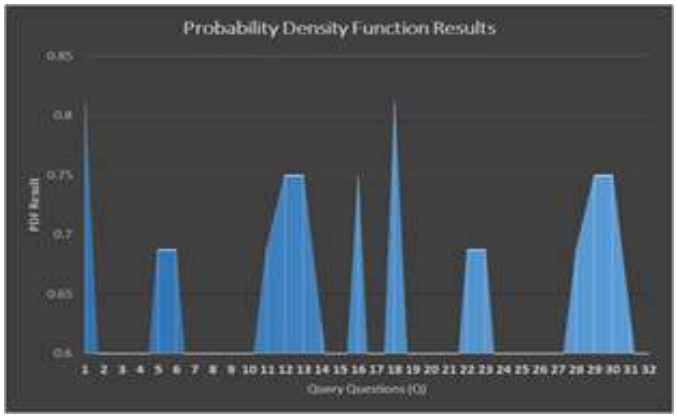


Figure 2. PDF result

The figure observes that not all the query questions are useful to create a pattern to identify the risks effect.

Step 3: the results of all queries can be applied again in the Probability Density Function (PDF) in order to find the second density of probability in all queries. The final formula of the present method is:

$$Pr_F = \sum_1^n (\int_a^b f_X(x)dx \int_c^d f_X(y)dy) \quad (3)$$

6. EXPERIMENT RESULTS

This paper integrated the ANN and PDF methods, with the latter presenting the former's pattern. Based on the results, the software project risks were important in the perspective of the project managers, which is why all controls were used often. The risks were ranked on importance in light of analysis, planning, maintenance, design and implementation. More specifically, the top software risk factors in the lifecycle of software development were obtained and the responses were aggregated to provide the outcome. The surveys were filled based on the selected software risk factors that presented in table 1. The study trained the neural network employing scaled conjugate gradient back propagation algorithm. The study conducted a comparative analysis with two groups; first, with the ANN results and the second with ANN-PDF results. The results of training step observed as shown in Figure 3. the result achieved represent the level of machine learning, the MATLAB training figure observed the risk factors input representation which account 32 factors, and with two layers steps the pattern produced in on risk factor which represent the major risk in the project.

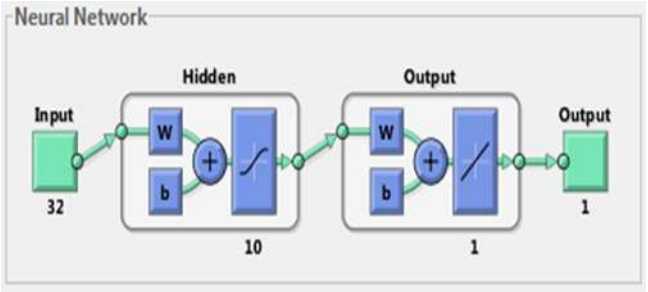
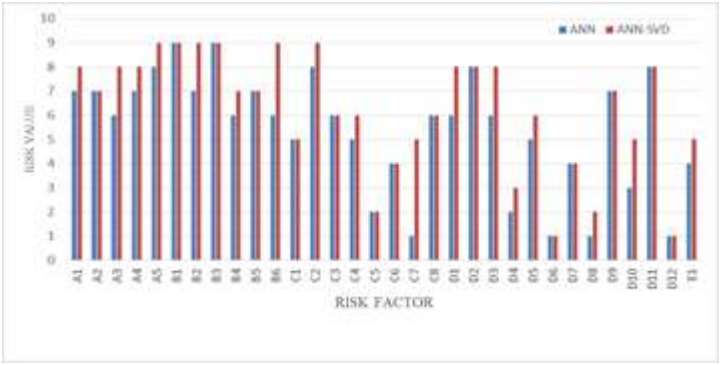


Figure 3 Neural network with 1 hidden layer and 10 hidden nodes

Figure 4 presents that the plotted results noted the same responses of the risk factor effects. The blue bars in the figure depict the ANN results and the red bars are ANN-PDF results.



Figures 4 Risk factors Comparative analysis

The presented bar graph reveals the risk identification of the method used and the valid results indicate the top IT software risk factors presented in table 1.

Figure 5 Identifying the Software Requirement Needs

Figure 6: S Identifying the software Cost Risks

Figure 7: Identifying the Software Scheduling Risks

Figure 8: Identifying the Software Quality Risks

The above figures illustrates the results of the identification of the software improvements needs based on the human skills, knowledge and the experience level which considered the top risk factors noted. With regards to the risk factor classification, on the basis of the five categories, the results found that software cost risks represent the first priority risk area, while the software requirement take the second priority.

7. CONCLUSIONS

This study applied the PDF data analysis method coupled with ANN to provide support for risk management processes. On the basis of the results, the PDF method displayed total variance in common questionnaire items of each software risk factor to measure the weight of the factor accurately. The PDF outcomes were juxtaposed on the ANN input representation enhance the training and testing results and the comparison showed that the method can be utilized effectively for the identification of risk effects in the entire phases of the project. The employed method categorized the top risks into human skills, knowledge and the experience staffing level and difficulties in applying software management and organization representing software cost risks of the total risk classification.

ACKNOWLEDGEMENT

The author thankful Department of Computer Science, Collage of Science, Mustansiriyah University, for supporting this work.

REFERENCES

[1] Al-balqa SKA, Al-balqa TJK. Risk Factors in Software Development Phases. 2014;(January).

[2] Hoodat H, Rashidi H. Classification and Analysis of Risks in Software Engineering. 2009:446-452.

[3] Economics E, Kong H. Integrating Safety , Environmental and Quality Risks for Project Management Using a FMEA Method. 2010;(January).

[4] Elzamly A. Evaluation of Quantitative and Mining Techniques for Reducing Software Maintenance Risks. 2014;8(111):5533-5542.

[5] Info A. Artificial Neural Networks Management : A Review for Construction. 2017;2:70-88.

[6] Gandhi A, Naik A, Thakkar K, Gahirwal M. Risk Management in Software Development using Artificial Neural Networks. Int J Comput Appl. 2014;93(19):1-7.

[7] Salman HA. G lobal J ournal of E ngineering S cience and R esearch M anagement. 2018;5(5).

[8] Okema JE. Risk and Uncertainty Management of Projects : Challenges of Construction Industry. (1990).

[9] Angelini E, Roli A. A neural network approach for credit risk evaluation. 2008;48:733-755. doi:10.1016/j.qref.2007.04.001

[10] Hojjati SN. The use of Monte Carlo simulation in quantitative risk assessment of IT projects. 2015;2621:2616-2621.

[11] Galway NUI. Towards an Integrated Risk Management Framework for Quality , Environmental and Health & Safety Management Systems. 2011.

[12] Informatik Q. Risk Management for IT and Software Projects. :1-16.

[13] Fredekind RE, Cuny EJ, Nadershahi NA. Quality Assurance and Risk Management : A Survey of Dental Schools and Program Management. 66(4).

[14] López C, Salmeron JL. Monitoring software maintenance project risks. 2012;5:363-368. doi:10.1016/j.protcy.2012.09.040

[15] Elzamly A, Hussin B, Salleh N. Top Fifty Software Risk Factors and the Best Thirty Risk Management Techniques in Software Development Lifecycle for Successful Software Projects. 2016;9(6):11-32.

[16] Berzina L, Zujevs A, Sudars R, Berzina L, Zujevs A, Sudars R. Neural network approach in risk assessment of phosphorus loss. 2008:320-326.

[17] Schwarz PJ. Implementation of artificial intelligence into risk management decision-making processes in construction projects.

[18] Pacelli V, Azzollini M. An Artificial Neural Network Approach for Credit Risk Management. 2011;2011(May):103-112. doi:10.4236/jilsa.2011.32012

[19] Zernikow B, Holtmannspoetter K, Michel E, et al. Artificial neural network for risk assessment in preterm neonates. 1998.

[20] Carr MJ, Konda SL. Taxonomy-Based Risk Identification. 1993;(June).

[21] Webster PB, Oliveira M De, Anquetil N, Av B, Norte W. A Risk Taxonomy Proposal for Software Maintenance.

[22] Neves SM, Silva CES da. Risk management applied to software development projects in incubated technology-based companies: literature review, classification, and analysis. Gest Prod, São Carlos. 2016;23(4):798-814. doi:10.1590/0104-530x472-15

[23] Kostelich E, Kuhl D. Introduction to SVD and Applications. Practice. 2008;(1):1-5.

Table 1 Top software risk factors in software project lifecycle based on researchers

Software Requirement Risks	A1	Lack of analysis for change of requirements
	A2	Poor definition of requirements
	A3	Ambiguity of requirements
	A4	Inadequate of requirements
	A5	Invalid requirements
Software Cost Risks	B1	Lack of good estimation in projects
	B2	Lack of reassessment of management cycle
	B3	Unrealistic schedule
	B4	Complexity of architecture
	B5	Human errors
	B6	Lack of monitoring
Software Scheduling Risks	C1	Inadequate knowledge about Techniques
	C2	Human errors
	C3	Lack of employment of manager experience
	C4	Lack of good estimation in projects
	C5	Lack of enough skill
	C6	Lack of accurate system domain Definition
	C7	Lack of goals specification
	C8	Lack of a good guideline
Software Quality Risks	D1	Lack of enough skill
	D2	Lack of good estimation in projects
	D3	Human errors
	D4	Lack of employment of manager experience
	D5	Lack of project standard
	D6	Poor definition of requirements
	D7	Inadequate knowledge about techniques
	D8	Lack of accurate system domain definition
	D9	Inadequate knowledge about programming language
	D10	Lack of reassessment
	D11	Inadequate knowledge about tools
	D12	Lack of analysis for change of requirement
Software Business Risks	E1	Failure in management because of change in different people