A SURVEY ON SOFTWARE PROJECT MANAGEMENT USING MACHINE LEARNING

Jasmin Nizar*, K. Anuradha

Abstract

The area of software project management (SPM), which is connected to the study of software engineering (SE), is significant. SPM's growing importance in graduate and postgraduate academic programs has, as a result, attracted more scholars in latest years. Many studies have been conducted by researchers employing machine learning techniques to examine it from various angles. The main objective of this work is to evaluate the current state of machine learning-based SPM. An extensive overview of numerous studies on the use of machine learning in software project management is provided in this article. Dr. K. Anuradha

Keywords: Software Project Management (SPM), Machine Learning (ML), Software Engineering (SE).

I INTRODUCTION

One of the academic domains in evaluating the success of any project has proven to be project management. Conflicting requirements, fluctuating deadlines, and personality clashes are commonly characteristics of the development and management of large and dynamic systems. Software project management (SPM) is a task that prioritizes people [1]. Businesses face challenges while handling software projects, including achieving successful projects and improving software project performance. Lack of data, qualified staff, and appropriate project management techniques usually result in riskier project breakdowns. Appropriate resource management and quality attainment are required to lower errors. The prediction models are created using machine learning methods such as the K-nearest Neighbor Algorithm (KNN), Artificial Neural Networks (ANN), Support Vector Machine (SMV), and others.

The application of machine learning in various project management phases has been highlighted in the past [2]. The researchers have put in a lot of effort to investigate different aspects of SPM. A review of the study literature reveals the potential to identify the hidden causes of developments in the chosen industry. It can be done by compiling, categorizing, and assessing the most recent studies on SPM that have been conducted using machine learning.

The main goal of this study is to use a systematic methodology to give a review of SPM utilizing machine learning. The rationale for employing automated projects, the issue of project management assessment, and the advancement of machine learning approaches are discussed using literature findings. The outcomes of the trial would be assessed. Despite the fact that the project's literature covers both achievements and setbacks.

The review of project requirements, which also leads to costs, time expenses, unfulfilled goals, or even project cancellations, is a natural, unplanned project risk of negative effects on the dependability of web applications [3]. The primary factors contributing to problems during the software development process are numerous extensions, deletion, and change. Consequently, we want to use the potential of machine learning in this research [4].

The following is how the document is set up. Section 1 provides a quick overview of approaches and methods for

[1]Department of Computer Science,
Karpagam Academy of Higher Education , Coimbatore, IndiaTamil Nadu
*Corresponding Author

SPM with machine learning is covered. Section 2 provides an explanation of the principles underlying software project evaluation and computer training technologies. Section 3 introduces the methodology, which includes a systematic literature review (SLR). The paper's results are displayed in Section 4 of the text. The final portion includes the research's findings as well as a recommendation for additional study.

II. PRELIMINARY STUDY

Some software project management principles, software estimating approaches, cost estimation strategies utilizing SPM, risk management, and machine learning techniques are presented in the preliminary study.

A. SOFTWARE PROJECT MANAGEMENT

Software project management entails overseeing the software life cycle, starting with the gathering of software requirements, followed by design, testing, and maintenance, all carried out promptly and in accordance with established project management methodologies [5]. The life cycle of each project is divided into portions in traditional software project management, allowing for the development of each project to be finished in stages. Software project management has a specific task that must be completed, along with a procedure that is designed to do so. As follows [6]:

- Project Planning and Tracking
- Project Risk Management
- Estimation Management
- Scope Management
- Configuration Management
- Project Communication Management
- Configuration Management
- Project Resource Management

B. SOFTWARE ESTIMATION METHODS

One of the most crucial tasks in managing a software project is estimation. Software estimation procedures include a variety of techniques for calculating the work, cost, and duration of software project development. The estimation of effort and duration is a crucial responsibility in software project management (SPM). The forecast needs to be accurate and trustworthy. The effort estimates at this early stage of a software project must be precise. This will lower the significant risks involved in software product development. However, it was shown that Machine learning-based software project estimation can offer a more precise effort estimation.

C. COST ESTIMATION APPROACHES IN SPM

Basic economic resources are used to find cost estimates and develop software projects using a method called cost estimation. The cost of developing software is estimated using cost estimation models, which are mathematical formulae. For the purpose of estimating effort, expense, and schedule for software projects, Boehm proposed the COCOMO model in 1981. Delivered source instructions' effort in man-months versus project size in a scatter plot reveals that there are three separate groups in the COCOMO data for effort in man-months versus product size. The three classifications were given the names organic, semidetached, and embedded by Boehm. Equation (1) and (2) shows the link between effort size and cost [6]:

$$MM = Ai \times (KDSI) Bi \qquad (1)$$

$$TDEV = Ci \times (MM) Di \qquad (2)$$

where MM represents for effort in Man-Months, TDEV is for development time in months, and KDSI stands for thousands of delivered source instructions. Ai, Bi, Ci, and Di are the constants that vary for each of the three modes.

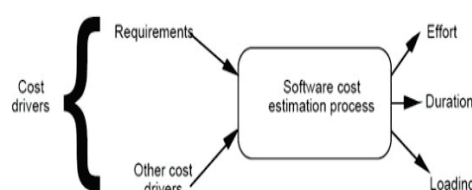Figure 1 depicts the traditional approach to software project cost estimation [6]:

Figure 1: The classical view of software project cost estimation

D. RISK MANAGEMENT

The process of risk management entails identifying, addressing, and resolving problems that might compromise the cost, timeline, or functional success of the software product. The two primary steps in risk management are significant risk assessment and risk control. Risk assessment includes the identification and evaluation of potential hazards that can obstruct work or the final product.

Any number of hazards can be managed in a software project. The risks can be divided into various categories in order to systematically identify significant hazards that may have an impact on a software project. The decision of which risks from each class are appropriate for the software project is made by the Project Manager (PM) [7].

E. MACHINE LEARNING

Artificial intelligence is used in machine learning, which enables computers to automatically learn from their experiences and get better without explicit programming [8]. The basic goal of machine learning is to enable computers to autonomously learn without the involvement of humans. Due to the learning-oriented nature of the methodologies, machine learning models have demonstrated great accuracy [9]. Supervised learning, unsupervised learning, semi-supervised learning, analytical learning, and reinforcement learning are the five main classifications of machine learning techniques [10].

The relationship between project circumstances and software process models from previous software projects can be automatically modelled using machine learning. The process model selection problem can thus be effectively solved by using machine learning approaches, according to data [11]. Machine learning techniques have proven to be useful in a variety of application fields. The management of software projects is proving to be a fruitful sector where numerous jobs, such as software creation and maintenance,

may be developed as learning issues and improved upon in terms of learning algorithms.

III. RESEARCH METHODOLOGY

The paper's primary goal is to provide a comprehensive evaluation of articles on the use of machine learning in software project management. Figure 2 depicts the complete process and steps for conducting this review job.
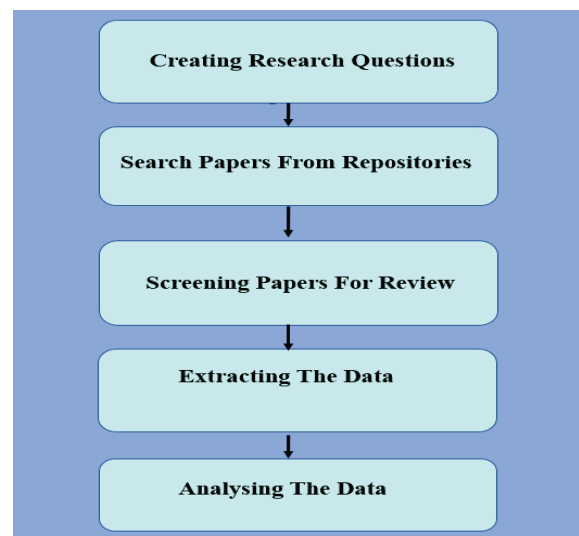

Figure 2: The proposed methodology of SPM using ML

The initial part of the research or study was to formulate the research questions to investigate software project management using SPM, as shown in figure 2 above. The second phase involved looking through the papers in the relevant repositories. The third step involves selecting papers for evaluation. The studies are chosen, and the data is extracted in the fourth stage. The analysis of the data for discoveries and review outcomes comes in the last step.

IV. RESEARCH QUESTIONS

The development of research questions is crucial in order to properly conduct a review of the literature. Our studies' primary goal is to find ways to use machine learning to potentially disclose SPM's hidden components. We explore the many research questions RQs to obtain a thorough

visualization of this subject. Below are some of the questions:

RQ1. What kind of studies in software project management employ machine learning methods?

RQ2. What are the topics based on the use of machine learning techniques to software project management is covered in the literature review?

RQ3. Which major branch of research has expertise in applying machine learning methods to software project management?

RQ4. How can different machine learning estimation algorithms illustrate the current degree of precision in software project management?

RQ5. What aspects of the management of software projects still require improvement?

RQ6. How are challenges in software project management handled utilising different approaches when machine learning is used?

V. SEARCH STRATEGY

A variety of digital libraries, including the IEEE Digital Library, ACM, Google Scholar, SpringerLink, and Science Direct are available for the search. The digital repositories can be used for this research's paper search. Both manual and automatic search methods were used by us.

VI. REVIEW AND SURVEY ARTICLES

The review and in-depth analysis of research publications explain the state of the art in machine learning technologies used in software project management and the algorithms that are used to predict. The study is broken down into various topics and applications. Based on machine learning (ML) techniques used in software development and risk management, the research has been divided into main groups. Table 1 summarizes a few carefully chosen review articles on software project management that use machine

SNo	Paper Title	Outcomes/applications	Ref.
1.	Software Project Management Using Machine Learning Technique—A Review	Machine learning techniques; ANN, Decision tree; software project estimation; Regression algorithm, By using the ML approaches in SPM, it has been concluded that we can measure the progress/ performance of a project in terms of threat, advantages and recommendation.	Mahdi, M.N.; Mohamed Zabil, M.H.; Ahmad, A.R.; Appl. Sci. 2021, 11, 5183. https://doi.org/10.3390/app11115183
2.	Teamwork-performance prediction by using soft skills and technological savvy skills	The regression and support vector machine algorithms, the random forest classifier method give a higher accuracy rate. The study demonstrated that well-rounded Technological Savvy Skills are a critical component in project teamwork. The study also identified soft skills that are adaptive to team processes.	Hoi Jun Lin Jia You (2021), ScienceDirect, Current Research in Behavioral Sciences, 3 September 2021
3.	Effects of co-curricular activities on student's academic performance by machine learning	The authors experimented with voting perceptrons, logistic regression, multilayer perceptrons, and the random forest classifier method. Among these, the authors obtained an accuracy of 99.5294% in logistic regression. The authors achieved 96.845% accuracy using the random forest method. To be successful in their professional lives, the writers must possess other abilities such as interpersonal skills, communication skills, leadership skills, management skills, and teamwork abilities.	Shaikh Rezwan Rahman, Md. Asfiul Islam, Computers & Electrical Engineering Volume 89, January 2021
4.	Enhancing prediction of student success: Automated machine learning approach	AutoML to increase the prediction accuracy of student performance using data features available prior to the students starting their new academic program, i.e. pre-start data.	HassanZeineddineUdoBraendle AssaadFarah , Computers & Electrical Engineering Volume 89, January 2021, 10690
5	Understanding the soft side of software projects: An empirical study on the interactive effects of social skills and political skills on complexity – performance relationship	Use of variance based structural equation modeling, the findings illuminate theoretical and practical contributions in better understanding complexities in software-projects performance. The results of the research introduced a new perspective to look at the project performance or reason of project failures i.e. Behavioral attributes of project managers.	Umer Zaman a,*, Zulaikha Jabbar b, Shahid Nawaz, International Journal of Project Management Volume 37, Issue 3, April 2019, Pages 444-460
6	Enhancing the Student Perception on Software Project Management in Computer Science	Two empirical studies were conducted. The first one, based on Ground Theory(GT) research. So that obtained findings was done in some improvements in a course redesign. The second study, the student's satisfaction level regarding the new course design was analyzed. The introduction of the PBL methodology and the sessions involving companies have greatly influenced the students' attitude toward the course	Antonia Mas , Antoni-Lluis Mesquida , and Ricardo Colomo-Palacios, 0018-9359c 2020 IEEE.
7	Deep-Learning-Based Agile Teaching Framework of Software Development courses in computer science education	This paper proposes to use Deep Auto-Encoder to map explicit qualitative and quantitative features into the abstract representation of skill status, and then construct a Recurrent Neural Network model to reveal the nonlinear relationship between students' skill status and its improvement stimulated by the project units.	Jian Yang, 1877-0509 © 2019 The Authors. Published by Elsevier Ltd. 10.1016/j.procs.2019.06.021
8	Use of machine learning techniques for educational proposes: a decision support system for forecasting students' grades	This paper aims to fill the gap between empirical prediction of student performance and the existing regression techniques and the random forest classifier method give a higher accuracy rate.	S. B. Kotsiantis, 12 May 2011 © Springer Science+Business Media B.V. 2011

Table1: Summarization of some selected review papers

VII. ANALYSIS & DISCUSSION

The most recent developments in software project management using ML technology are covered in this paper. The goal of this study is to identify current themes in this area of research. The research reveals the surface levels of the

literature's substance, including the drivers driving the advancement of ML-based software project management, the difficulties in doing so successfully, and suggestions for how to deal with these issues.

The benefits of making use of the ML platform for software project management are obvious. Figure 3 illustrates this section, which highlights some of the advantages of SPM utilizing ML in the literature:
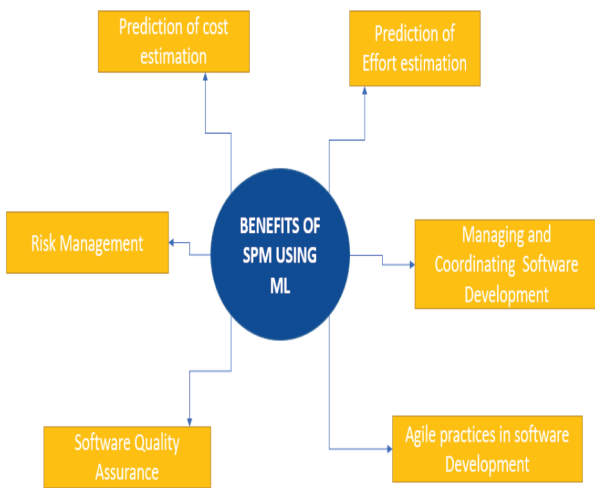

Figure 3: Benefits of SPM using ML

The framework for cost estimates is used to verify predictive failure models. The following are the key conclusions drawn from the review papers: (1) New techniques outperform commonly used ones. (2) In software projects with an error group percentage below the Wagner proposed threshold number, a fault prediction strategy is effective [12]. Fault prediction is the main emphasis of the ensemble techniques. The execution of anticipated monitoring costs and uniform fault elimination costs serve as the single input.

Risk assessment and a number of cost criteria are used to estimate the software effort [13]. In software project development, predictable effort is calculated. Risk

management is the practice of engaging in all necessary risk management behaviors. A basic component in the project preparation phase, risk evaluation is vital in determining how well a software development project is progressing [14]. Software quality assurance and agile practices include communication concerns, issues with forming team or group relationships cultural obstacles, and difficulties controlling and organizing distributed software projects. In conclusion, it is still challenging to design applications that need for tight communication in a distributed setting [15].

VIII. CONCLUSIONS

The literature review on the use of ML techniques in software project management has been completed in this study. ANN, Fuzzy Logics, classification, and regression algorithms are the primary Machine Learning techniques utilised for effort calculation. These precise measurements are the cornerstones of software development techniques. The primary aim of this article is to provide justifications based on the findings. Different software effort estimation projects can provide distinct patterns. The performance of projects and their failure are explained by a review of the literature on software project management. Regarding how progress may be measured and what accurately depicts development on a software project, there are numerous points of contention. These tactics and guidelines will address the problems that arise during the development of software employing ML approaches.

REFFERENCE

[1] Muhammad Shoaib Farooq, Uzma Omer, Rabia Tehseen, and Ifat Un Nisah, Software Project Management Education: A Systematic Review, 2021 VFAST Transactions on Software Engineering Volume 9, Number 3, July-September, 2021 DOI: 10.21015/vtse. v9i3.702

[2] Ishaan Buch, Kirtan Pokiya, Project Management using Machine Learning, November 2020, publication/ 348327272_Project_Management_using_ Machine_Learning.

[3] Wang, J.; Li, J.; Wang, Q.; Zhang, H.; Wang, H. A simulation approach for impact analysis of requirement volatility considering dependency change. In Proceedings of the International Working Conference on Requirements Engineering: Foundation for Software Quality, Essen, Germany, 19–22 March 2012;

[4] Sikiru Adewale, Amarachi Mbakwe Machine Learning Project Paper,December 2020 DOI:10.13140/RG.2.2.36291.53287

[4] Ahmed BaniMustafa, Predicting Software Effort Estimation Using Machine Learning Techniques, 2018 8th International Conference on Computer Science and Information Technology (CSIT) ISBN: 978-1-5386-4152

[5] Tameem Abdulbaset Abdulwahid Abdo Hezam, Software project management, March2021, DOI:10.6084/m9. figshare.14945535.v1

[6] Ishaan Buch, Kirtan Pokiya, Project Management using Machine Learning, November 2020, publication/ 348327272_Project_Management_using_ Machine_Learning.

[7] Dinesh Bhagwan Hanchate, Rajan Bichkar, The machine learning in software project management: A journey. Part II january 2016,, Applied Discrete Mathematics and Heuristic Algorithms,(4):29-58

[8] Machine Learning-2020 Technology, Ieee Paper, Ieee Project

[9] P. Sharma and J. Singh, "Machine Learning Based Effort Estimation Using Standardization," in 2018 International Conference on Computing, Power and Communication Technologies (GUCON), 2018, pp. 716-720.

[10] Subhendu Kumar Pani, Anil Kumar Mishra, Machine Learning Applications in Software Engineering: Recent Advances and Future Research Directions, DOI: 10.17577/IJERTCONV8IS01015

[11] QinbaoSongabXiaoyanZhub GuangtaoWangb HeliSunbHeJiangc ChenhaoXueb BaowenXua WeiSongd, A machine learning based software process model recommendation method, Journal of Systems and Software Volume 118, August 2016, Pages 85-100

[12] S. Wagner, "A literature survey of the quality economics of defect detection techniques," in Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering, 2006, pp. 194-203.

[13] A. B. Nassif, D. Ho, and L. F. Capretz, "Towards an early software estimation using log-linear regression and a multilayer perceptron model," Journal of Systems and Software, vol. 86, pp. 144-160, 2013.

[14] K. Heldman and W. Heldman, CompTIA Project+ Study Guide Authorized Courseware: Exam PK0-003: John Wiley & Sons, 2010.

[15] T. Menzies, O. Mizuno, Y. Takagi, and T. Kikuno, "Explanation vs performance in data mining: A case study with predicting runaway projects," Journal of Software Engineering and Applications, vol. 2, p. 221, 2009.