International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048
Volume 7, Issue 12, pp 67-72, December-2023
https://doi.org/10.47001/IRJIET/2023.712009

Survey of Cost Estimating Software Development Using Machine Learning

[1]Nedaa Thamer Qassem, [2]Ibrahim Ahmed Saleh

[1]Student, Department of Software, College of Computer & Math., University of Mosul, Iraq
[2]Professor, Department of Software, College of Computer & Math., University of Mosul, Iraq

Abstract **- The estimate of software project costs and efforts is a critical step. Meeting a plethora of diverse criteria, such as resource allocation, cost estimate, effort estimation, time estimation, and the shifting expectations of software customers, is a component of software estimating. Software Engineering Models assist project managers in estimating the cost, delivery time, and personnel that were crucial for software development. SDCE (Software Development Cost Estimation) has long been an exciting and developing area. There are several intelligent models used to predict or estimate the cost of software. The aim of this research is to highlight the algorithms that are used for cost estimation. However, there are no models that work in different circumstances that each practitioner or researcher chooses there are models that use algorithms and those that don't. This makes it easier to determine how much it will cost to build software. Comparing machine learning approaches to traditional software estimating, the ability to anticipate program expense with a high rate of accuracy is possible.**

Keywords: software Engineering, cost estimation, software development, machine learning.

I. INTRODUCTION

In modern economic conditions, the development and implementation of new technologies is especially important for successful competition of companies. An important element in making investment decisions on technological projects is the evaluation of their effectiveness. Since the market for the purchase / sale of new technologies exists and functions, it becomes necessary to determine the cost of development. The object of the transaction may be technology at different stages of its development creation and/or implementation. The main criterion for evaluating technologies is the "net value" indicator, determined for different stages of implementation and implementation of intellectual value, different scales of its application, competitive environment and different investment risks [1]. Successful investment and asset management requires not only an understanding of what "value" is, but also knowledge of the factors that influence it.

The creation of new technologies is a step-by-step process: from its development to bringing it to a commercially successful implementation. Technology cost estimation is necessary to analyze the profitability of current and future technology projects, and hence the feasibility of investments. An important feature of modern technologies is their information capacity, which, in particular, is ensured by the widespread use of software. Since the 60s of the twentieth century, when the software market appeared, i.e. software has evolved into a software product and has become a special kind of commodity (information commodity), software development cost estimation remains one of the most difficult issues in software engineering, which is puzzled by financiers, appraisers, analysts, programmers, development engineers, leaders of innovative enterprises and research institutes. Due to the fact that the preliminary estimation of the development cost includes many elements of uncertainty, enterprises use a wide range of methods in practice - from the most elementary to the most sophisticated. What is a software product? Here are definitions from various normative documents.

A software product is a software tool intended for delivery, transfer, sale to a user. Accordingly, a software tool is an object consisting of programs, procedures, rules, and, if provided, their accompanying documentation and data related to the functioning of the information processing system "Interstate standard GOST 28806–90. Software quality terms and definitions" [2].

A software product is a publication of the text of a program or programs in an executable code format or in a programming language. It is an autonomous, alienable work. "Interstate Standard GOST 7.83–2001. SIBID "System of standards for information, librarianship and publishing, Electronic publications, Main types and output data [3].

A software product is a collection of computer programs, processes, and perhaps associated data and documentation "GOST RISO / IEC 12207:2010" digital technologies Software and system engineering. Methods of the software life cycle [4].

Let's highlight the main features of the software product. This is, firstly, a complex of programs presented in various

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 12, pp 67-72, December-2023

https://doi.org/10.47001/IRJIET/2023.712009

forms, and, secondly, an alienable work, and alienation is ensured by the presence of the attached documentation and data. Consideration of a software product as an economic object involves the study of economic foundations, as well as existing approaches and methods for estimating the cost of developing a software product [5].

There are several common methods used to estimate the cost of software, including the COCOMO model. Which was used in the eighties to estimate the cost, despite its advantages, but it does not contribute to solving the problem of accuracy in estimation, as well as its dependence on historical data that may not be available at all times and takes time for estimation [5] Other models used are neural networks, regression-based techniques and others that give more accurate results than traditional methods [7] .Although there are techniques for calculating the cost accurately, but it differs according to the circumstances. There is a need for techniques to avoid doubt in the estimation process [8].

The purpose of this paper is to analyze the advantages and disadvantages of various software cost estimation techniques. By exploring these techniques, researchers and practitioners can gain insights into the appropriate domains for their deployment and the specific factors that make each model well-suited for a particular area. This paper will thoroughly examine and discuss these matters.

II. COST ESTIMATION

Estimating the cost is one of the most difficult phases of project management for the project manager, as the cost can be estimated using different tools and models. One of the obstacles facing the project manager is improving the efficiency of estimating the cost of software. Poor estimation can lead to software failure. It is possible to use historical data to benefit from it in the development of forecast models [9]. The cost estimation process is one of the most important elements of project management, as the project manager faces challenges in delivering the software or product on time within the budget and according to the specifications. The wrong estimation can lead to failure or delay [1]. It is the process of estimating the cost and time spent by developers to develop a software project before work. The project manager uses appropriate methods and tools to estimate the cost to control the progress of work and avoid problems that may appear with progress such as lack of resources that lead to delay [10]

The topic of software cost estimating involves a number of challenges since it necessitates careful observation and repeated testing before determining that a certain approach is appropriate for the estimation needs. One way to estimate the amount of work and money that will go into a particular software project is to use software cost estimation methodologies. Therefore, the software cost estimating procedure makes it feasible to estimate the number of workers needed and the time needed to finish the project in the allotted amount of time. (Figure 1).

Figure 1: Process of cost estimation

Cost estimation is a prediction of the effort required to develop a software product. A successful software project that achieves requirements within a predetermined time and budget [11]. The process of estimating project effort is one of the basic stages that are determined at the beginning of the project. To calculate the development cost, it is planned in advance, and the effort, cost, and budget of the project are determined. Accurately estimating the cost of software development contributes to the success of the project [8]

III. RELATED WORK

There are numerous papers and studied on software cost estimation approaches have been published to date. These have discussed several approaches to cost estimation, some of which are shown in figure 2.

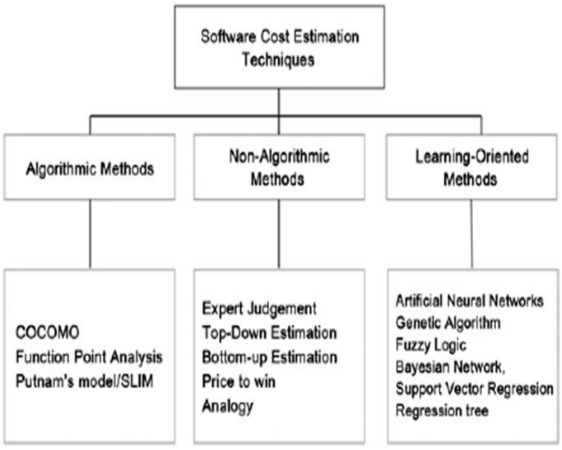

Figure 2: Software cost estimation techniques

In 2010 Reddy and his team [12] presented they concluded that Radial Basis neural networks(RBF) for developing software effort estimation model based on COCOMO dataset consist of 63 projects which randomly select 53 project for training and 10 project for testing . the model gave better results for criteria when they compared

ISSN (online): 2581-3048
Volume 7, Issue 12, pp 67-72, December-2023
https://doi.org/10.47001/IRJIET/2023.712009

them with other models based on the scales (MARE, VARE, MEAN BRE, PRED) they proved in their result that Radial Basis Neural Network is better than GRNN or Intermediate COCOMO out of 63 projects, where they used 53 random projects and used them as training data.

In 2014, Dave and Kamlesh [13] introduced an innovative software development effort estimation model. This novel approach combined the traditional COCOMO model with Artificial Neural Networks (ANN) to create a more accurate estimation tool. Their model underwent enhancement through the integration of neural networks, refining its predictive capabilities. To validate their model, the researchers utilized three comprehensive datasets for training and testing. Employing the back propagation feed forward technique; they fine-tuned the neural network by iteratively processing the training data samples. To assess the model's performance, they compared its estimates with actual effort values, finding that the proposed model produced estimates much closer to the real effort than the traditional COCOMO model. Results indicated that the Mean Magnitude of Relative Error (MMRE) for their model was significantly lower compared to other models like COCOMO, DANN, BPNN, and RBNN. This experiment showcased the superior predictive power of their model over its counterparts, marking a substantial advancement in software development effort estimation. in the same year Hidmi, Omar, and Betul Erdogdu Sakar [14] presented model for estimate software effort objectively some machine learning models on different data types. The researchers used two models k-nearest neighbor) and support vector machine each method was applied to different types of data to predict the effort to develop a software project. Where the results showed that when using a single method, accuracy is equal to (85%) When using the (SVM) method on two categories of a data set (desharnais). But when the classifiers were merged, they obtained accurate results (91.35%) using a desharnais dataset. But when they used data from (Maxwell) Obtained accurate results (85.48%).

In 2018, Saljoughinejad and Vahid [15] presented a novel hybrid approach aimed at enhancing the accuracy of cost estimation within the COCOMO model. This approach incorporated three distinct artificial optimization algorithms: Particle Swarm Optimization (PSO), Invasive Weed Optimization (IWO), and Genetic Algorithm (GA). The objective was to evaluate the performance of this proposed model, with MMRE and PRED scores both achieving a value of 0.25. Experimental outcomes obtained from real-world software projects demonstrated that the hybrid (IWO-PSO) model significantly improved the precision of cost estimation results."

In this research, Arslan (2019) [16] conducted an assessment of 13 machine learning algorithms using two distinct datasets. The evaluation was based on multiple criteria, including R^2, MAE, RMAE, RAE, and RRSE. The primary objective of this study was to create a predictive model for estimating effort by utilizing dataset attributes and comparing them to actual effort levels, while employing various evaluation metrics. Specifically, a higher R^2 value indicated better performance, whereas lower values were favored for the remaining criteria. The initial experiment featured the application of the Random Forest algorithm, which yielded superior results when working with the 'Usp05-ft' dataset. Additionally, three other models, namely REP Tree, Additive Regression, and K star, demonstrated better outcomes when employed with the 'Usp05' dataset. Conversely, the Zero R model exhibited poorer results when applied to the first dataset. Interestingly, certain methods, including Multilayer Perceptron, IBk, and Linear Regression, did not perform well when applied to the second dataset."

In 2020 Ilham [1] and other researchers suggested approach using Random Forest Regression and some machine learning algorithms for guessing cost and effort estimation for project where they implemented their proposed method of guessing using a NASA93 dataset. The researchers found the implementation results showed that (RFR) is the method that has the best results (54%) to estimate the program effort using “COCOMO II” the reason is that the size (MMRE) is small compared to Bee Colony Method MMRE (115%) and SVR MMRE equal (73%) . In the same year Khazaiepoor et al. [17] introduced hybrid approach method for software development effort consist of three stages, In the initial phase, features are chosen using a combination of a multilayer perceptron neural network and the genetic algorithm (GA). Using multiple linear regression techniques, the impact factors are linked to each selected feature in the second step, serving as impact coefficients for each feature, while in the last stages, an imperialist competitive algorithm optimizes the feature weights. The experimental findings demonstrate that the model produces respectable results on both Maxwell and Albrecht data. However, the results produced for the COCOMO dataset are comparatively weaker. This is likely due to the restricted diversity of projects in the Maxwell data sets. As is the case with other models, the model's drawback is its reliance on the data set.

In the 2021 the Abdulmajeed et al [18]. Introduced three techniques to predict the cost of software development from these technologies K-Nearest Neighbors Algorithm (KNN), convolution neural network CNN, and Emergency Nutrition Network ENN. They used NASA data where they compared the results of the algorithms (KNN, CNN, ENN) where the results showed accuracy in guessing and predicting the cost of

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 7, Issue 12, pp 67-72, December-2023

https://doi.org/10.47001/IRJIET/2023.712009

the software was high when using the (KNN) technology where the model used had lower values (MMRE, RMSE, BRE).The resulting accuracy in the guess reached (90.238%) when using (KNN) therefore the error rate is very low compared to the accuracy in the prediction is high.

The program's effort guess was predicted by researchers Farah and Laheeb [19] in 2022 when they presented an algorithm Long Short Team Memory (LSTM). This method was compared to other machine learning methods that were utilized for earlier projects utilizing the same data sets. Three criteria were used to evaluate the results: RMSE, MAE, and R _squared. Based on the findings, it was discovered that the LSTM method performed better than machine learning algorithms when compared to datasets from Kitchenhand with 145 projects and China with 499 projects. Where the outcomes of applying the (LSTM) algorithm were displayed, when using a dataset from China, the algorithm produced better results than when using the same dataset from Kitchenham. The metrics, MAE, R _squared, and RMSE findings for the China dataset are 0.016.

In 2023 researchers' Şengüneş and Nursel 20] suggested model for prediction about insufficient software project effort estimation in automation phase. They depended to Artificial neural network (ANN) for estimated avoid shortening the process of guessing the required effort. The dataset for-training is collected from characteristics of the project of (101) real projects, the researchers applied Bayesian optimization to improve hyper parameters. The value of prediction accuracy (PRED) were (83%) for training, (89%) verification and (73%) for examination. Apply ANN model on the problems of guessing in a realistic project they get accuracy was quite good compared with other studies in project effort estimation.

Table 1: Summary of the literature used in software cost estimation

Authors	Technique used	Accuracy	Weaken
Reddy , et al [12]	RBNN	MARE=7.13 VARE=3.27 Mean BRE=0.17 MMRE=17.29 Pred(40)=90.48	Don't take important for development of software engineering
Dave and Kamlesh [13]	COCOM ANN	MMRE=8.7896	It requires a large number of calculations
Hidmi, Omar et al [14]	KNN SVM	91.35% accuracy when using desharnais 85.48% accuracy when using maxwell dataset	Difficult to model and this method don't gave exactly values for classification because it not placed above and below the classifying hyperplane
Ramin Saljoughinejad1 and Vahid Khatibi2 [15]	PSO, GA IWO	MMRE=0.21 PRED(0.25)=0.75	The algorithm are many hybrid parameters and not needs supervisors
Arslan, Farrukh [16]	REP tree Random forest M5P ZeroR Decision table Additive- regression Input- mapped- classifier Linear- regression Multilayer- perceptron SMOreg IBK KStar Gussian- processes	R^2=0.8441 MAE=2.5025 RMAE=4.8546 RAE=41.7323 RRSE=55.6778	This methods are given many computations for find the accuracy
(Ilham) et al [1]	Random Forest Regression	MMRE =54%	This method gave good accuracy but it more slow and not suitable for real time.
Mahdi Khazaiepoora et al[17]	perceptron neural network genetic algorithm multiple linear regression Imperialist Competitive Algorithm	MMRE=0.34 MDMRE=0.18 PRED=0.58	They frequently over fit, and they need a tremendous amount of processing power.
Abdulmajeed et al [18]	KNN CNN ENN	MMRE=0.101 RMSE=0.547 BRE=0.205 Accuracy= 90.238	Large deferent among the three algorithm and the searchers not Shows why CNN is the best

ISSN (online): 2581-3048
Volume 7, Issue 12, pp 67-72, December-2023
https://doi.org/10.47001/IRJIET/2023.712009

Farah B [19]	LSTM	MAE=0.016 RMSE=0.019 R-squared=0.972	Low level issues are difficult to recognize, which results in underestimating; less information may obscure crucial project characteristics
Şengüneş and others [20]	ANN	MMER= 30% PRED(25)=70%	For this algorithm the researcher not reach in optimal solution and need a lot of epoch.

IV. CONCLUSION

In this paper, introduced an overview of the software cost, effort and size estimation techniques based on all the techniques different artificial intelligence algorithm approaches. The paper tabulated all the techniques based on their type, strengths, weaknesses, amount of data and validation methods used by them. The paper finds the machine learning algorithm for cost estimation is best model is exceed other models.

REFERENCES

[1] Ilham C. S., Riyanarto S., Sholiq, Implementation of Random Forest Regression for COCOMO II Effort Estimation. *IEEE reg*, 2020.

[2] I.Attarzadeh, A. Mehranzadeh, and A. Barati, "Proposing an Enhanced Artificial Neural Network Prediction Model to Improve the Accuracy in Software Effort Estimation," *in 2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks, Phuket, Thailand,* Jul. 2012, pp. 167–172, doi: 10.1109/CICSyN.2012.39.

[3] S.-J. Huang and N.-H. Chiu, "Applying fuzzy neural network to estimate software development effort," *Appl Intell*, vol. 30, no. 2, pp. 73–83, Apr. 2009, doi: 10.1007/s10489- 007-0097-4.

[4] A.B. Nassif, L. F. Capretz, and D. Ho, "Estimating Software Effort Based on Use Case Point Model Using Sugeno Fuzzy Inference System," *in 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, Boca Raton, FL, USA*, Nov. 2011, pp. 393–398, doi: 10.1109/ICTAI.2011.64.

[5] Mahdi, M.N.; Mohamed Zabil, M.H.; Ahmad, A.R.; Ismail, R.; Yusoff, Y.; Cheng, L.K.; Azmi, M.S.B.M.; Natiq, H.; Happala Naidu, H. Software Project Management Using Machine Learning Technique—A Review. *Appl. Sci.* 2021.

[6] Shweta.KR, Dr. S.Duraisamy, Dr. T.Latha Maheswari, Software Cost and Effort Estimation using Ensemble Duck Traveler Optimization Algorithm (eDTO) in Earlier Stage, *Turkish Journal of Computer and Mathematics Education* Vol.12 No.13 (2021), 3300-3311, 4 June 2021.

[7] Junaid Rashid, Muhammad Wasif Nisar, Toqeer Mahmood, Amjad Rehman, Syed Yasser Arafat, A Study of Software Development Cost Estimation Techniques and Models, *Mehran University Research Journal of Engineering and Technology*, Vol. 39, No. 2, 413- 431, April 2020.

[8] Noor Azura Zakaria, Amelia Ritahani Ismail, Afrujaan Yakath AliNur Hidayah Mohd Khalid, Nadzurah Zainal Abidin, Software Project Estimation with Machine Learning, Vol. 12, No. 6, 2021.

[9] A G, P.V.; K, A.K.; Varadarajan, V. Estimating Software Development Efforts Using a Random Forest-Based Stacked Ensemble Approach. *Electronics 2021*.

[10] Sakineh Asghari Agcheh Dizaj, Farhad Soleimanian Gharehchopogh, A New Approach to Software Cost Estimation by Improving Genetic Algorithm with Bat Algorithm, 18 September 2018.

[11] A.Saberi Nejad, R. Tavoli, A Method for Estimating the Cost of Software Using Principle Components Analysis and Data Mining, 20 December 2017.

[12] Reddy, P. V. G. D., et al. "Software effort estimation using radial basis and generalized regression neural networks." *arXiv preprint arXiv:*1005.4021 (2010).

[13] Dave, Vachik S., and Kamlesh Dutta. "Neural network based models for software effort estimation: a review." *Artificial Intelligence Review* 42.2 (2014): 295-307.

[14] Hidmi, Omar, and BetulErdogdu Sakar. "Software development effort estimation using ensemble machine learning." *Int. J. Comput. Commun. Instrum. Eng* 4.1 (2017): 143-147.

[15] Saljoughinejad, Ramin, and Vahid Khatibi. "A new optimized hybrid model based on COCOMO to increase the accuracy of software cost estimation." *Journal of Advances in Computer Engineering and Technology* 4.1 (2018): 27-40.

[16] Arslan, Farrukh. "A review of machine learning models for software cost estimation." *Review of Computer Engineering Research* 6.2 (2019): 64-75.

[17] Khazaiepoor, Mahdi, Amid Khatibi Bardsiri, and Farshid Keynia. "A hybrid approach for software development effort estimation using neural networks, genetic algorithm, multiple linear regression and imperialist competitive algorithm." *International Journal of Nonlinear Analysis and Applications* 11.1 (2020): 207-224.

[18] Abdulmajeed, Ashraf Abdulmunim, Marwa Adeeb Al-Jawaherry, and Tawfeeq Mokdad Tawfeeq. "Predict the required cost to develop Software Engineering projects

ISSN (online): 2581-3048
Volume 7, Issue 12, pp 67-72, December-2023
https://doi.org/10.47001/IRJIET/2023.712009

by Using Machine Learning." *Journal of Physics: Conference Series.* Vol. 1897. No. 1. IOP Publishing, 2021.

[19] Ahmad, Farah B., and Laheeb M. Ibrahim. "Software Development Effort Estimation Techniques Using Long Short Term Memory." *2022 International Conference on Computer Science and Software Engineering (CSASE). IEEE,* 2022.

[20] Şengüneş, Burcu, and Nursel Öztürk. "An Artificial Neural Network Model for Project Effort Estimation." *Systems* 11.2 (2023): 91.

Citation of this Article:

Nedaa Thamer Qassem, Ibrahim Ahmed Saleh, "Survey of Cost Estimating Software Development Using Machine Learning" Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 7, Issue 12, pp 67-72, December 2023. Article DOI https://doi.org/10.47001/IRJIET/2023.712009

****

