Predicting Product Development Time and Cost Using Production Data

Devraj Singh Chauhan[a], Dr. Komal Saxena[b],

Article History: Received:24.04.2023 Revised:01.05.2023 Accepted: 08.05.2023.

Abstract: Background: In this study, the time and costs associated with product development are predicted using manufacturing data. This study investigates the relationship between product development time and costs and production dates, and it creates models that may be applied to anticipate these factors. Statistical methods like regression analysis and machine learning algorithms are used in research procedures to collect and analyze production data. The findings show that product development durations and costs may be accurately predicted using production data, and that these forecasts are more precise than those made using more conventional techniques that rely on expert judgement and historical data. The report also explores how these conclusions will affect product development teams as well as managers and professionals working in risk analysis and project management. Production data assists businesses in better informed resource allocation and project planning, resulting in more productive and efficient production by more precisely calculating the time and costs associated with product creation. resulting in an enhanced product development process and company success.

Keywords: Production Data; Development Time and Cost; Project Planning; Decision Making

[a]. MCA, Student, AIIT
[b]. Associate Professor, Programme Leader, AIIT.

INTRODUCTION

From idea generation to market launch, creating a new product involves a lengthy and complex procedure. Predicting the amount of time and money it will take to bring a new product to market is one of the toughest problems for product development teams. For resource allocation and risk management, as well as for on-time and within-budget product delivery, accurate product development time and cost estimation is essential. Historically, project management methods, expert judgement, and historical data have been used to predict the length of time and cost associated with product development [6]. These techniques, however, frequently lack objectivity, are prone to mistakes, and might not take into account the particulars of each product development project. The use of manufacturing data to forecast the time and cost of product development has gained popularity in recent years. Data gathered throughout the manufacturing process, such as: B. Material consumption, working hours, machine utilization, and quality indicators, are referred to as production data[12]. This information offers important insights into the costs and resources needed to create the product as well as the process of product development [13]. Product development teams can create more precise predictions regarding development time and costs by looking for patterns and trends in production data. The goal of this study is to look at possible uses of manufacturing data to forecast the costs and timelines of product development. In particular, we investigate the relationship between product development time and costs and production dates and create models that may be applied to anticipate these factors. We assess these models' performance and compare it to more established techniques for determining the length of time and price of product development. The results of this study have significant ramifications for product development teams as well as managers and professionals working in risk analysis and project management. Production data assists businesses in making more informed decisions regarding resource allocation and project scheduling by offering more precise estimates of the time and costs associated with product creation. In the end, this results in a process for developing products that is more effective and efficient, as well as better company performance.

Time and money spent on product development are important considerations for each new product development project. For efficient resource allocation, project planning, and risk management, these variables must be forecasted accurately. Time and cost estimates for product development using the conventional method. B. Historical data and expert opinion are both restricted and prone to inaccuracy. The use of manufacturing data to forecast the time and cost of product development has gained popularity in recent years. Several studies have investigated the relationship between date of manufacture and product development time and cost. For example, Kim et al. [1] developed a model to estimate product development times for the electronics industry using production data. The study found that production data such as the number of design changes and the number of defects is important predictors of development time. On the basis of production data, Zhang et al. [4] created a machine learning model for forecasting product development expenses. In this study, a model was developed using data on things like material usage, working hours, and machine utilization. The findings demonstrate that machine learning methods are more precise than conventional approaches based on historical data and professional opinion [9]. Some studies concentrate on certain product categories and industry niches. For the automobile industry, Li et al. [2] created a model to forecast product development time. To create a model, the study analyzed information on product specifications, supplier quality, and manufacturing process capabilities. The model is effective at forecasting development time, according to the results, and can be used to enhance project planning and decision-making. I'm interested in integrating production data to enhance the product development process in addition to prediction models. For instance, Rausch et al. [3] analyzed production data to find areas where design for new product manufacturability may be improved. According to the study, examining production data can reveal design elements that are expensive or difficult to produce, improving product design and cutting down on development time and expenses. Taken together, these studies suggest that production data can be a valuable source of information for predicting product development time and costs. By analyzing production data using statistical methods and machine learning algorithms, product development teams can improve their ability to predict these variables, resulting in a more efficient and effective product development process. Additionally, by using production data to identify process improvement opportunities, companies can achieve cost savings and other benefits that contribute to the bottom line.

Aim of the study: This study conducted to specify at which age should the women in Iraq subjected to screening program for early detection of breast cancer.
Screening program in Iraq include.

- Clinical examination
- Ultrasound of the breast
- Mammography

PROPOSED WORK

The following steps are commonly included in a technique for estimating product development time and costs using production data:

- Data Collection: Gathering production information on the product development process is the first step. Data on material use, working hours, machine use, quality indicators, and other pertinent elements may be included.
- Data Analysis: Regression analysis and machine learning algorithms are used to examine the collected information in order to find patterns and connections between production data and the costs and times associated with product development [10]. To better understand your data and find any anomalies and errors, you can also employ exploratory data analysis approaches.
- Model Development: Predictive models that can be used to project product development time and costs are created based on the findings of the investigation. Models can be created using machine learning techniques like random forests and neural networks as well as statistical methods like multiple regression analysis.
- Model Confirmation: To determine accuracy and validity, developed models are evaluated against a validation dataset. Another production data set that isn't used in the model creation process could be a validation data set.
- Deploying the model: The models can be used to forecast product development time and costs for new goods or projects once they have been validated. Models are simple to include into already-in-use project management software or solutions. Generally, the methodology for predicting product development time and costs using production data is data-driven, and it is crucial to gather, analyses, and model production data in order to create precise and effective prediction models. included. In order to create models that can offer insight into the product development process and enhance project planning and decision-making, this methodology integrates statistical and machine learning techniques.
- Maintenance and Updates: The model is monitored and maintained to ensure that it remains functional and secure. Regular updates are also released to provide new features, bug fixes, and other improvements.

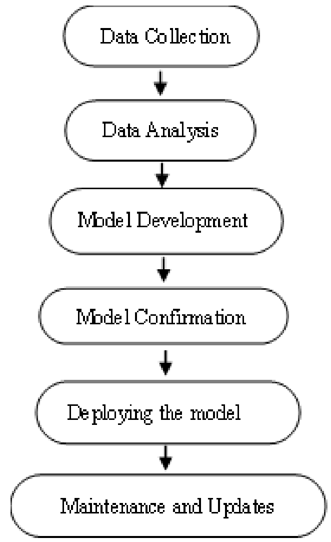

Figure 1. Flow chart of the Proposed Work

IMPLEMENTATION OF PROPOSED WORK

For predicting product development time and cost from production data have two different factor time and cost. First, we calculated time and last cost.
Estimate Time: Predict product development time from production data here take an example of software development time estimation. It is very difficult to estimate software development time.
We face several problems with estimate software development time. Major problems are given bellow:

- The process of software development is very difficult.
- Every software development is unique.
- Which type of team work on this project.
- Unforeseen problems.
- Availability of resources.

There are various methods to correctly calculate software development time like as:

- We can use historical data: It can be helpful to look at recent initiatives that the business has accomplished that are similar. This shouldn't, however, ever serve as the only consideration when making projections for the future.
- Analogous estimation in current scenario: Instead of estimating software development time based on previous projects, might look at actual circumstances and make changes to account for the variations [11].
- Bottom-up approach: With this strategy, each single work involved in the entire development process is thoroughly thought out, and the time needed to deliver each milestone

is estimated. When applied correctly, this technique may be incredibly powerful. But it might also take a lot of time.

- Top-down approach: By contrast, the top-down approach begins by estimating the overall amount of time needed for the project before dividing it down into more manageable portions.
- Parametric approach: In the case of parametric calculation, the time needed for software development is precisely predicted using statistical approaches. Software engineers must use data from earlier parallel projects during the estimation procedure to provide precise time predictions for your current project.
- Three-point approach: Three estimates are used in a three-point estimation: the most probable estimation, the less estimate, and the most hopeful estimate. The meaning of the three estimations used is then used to generate the final estimation.
- Function point approach: Creating a prediction for the entire development based on the software's capability is known as "function point analysis." Simply put, the longer it requires to construct something, the more features it has.

Calculate time for software development: Estimating software development time is a very complicated process; in this process we take various steps. These steps are given bellow:

Workflow: First of all, design the workflow of software and divide software into different levels. Here, we take example of rice management software:

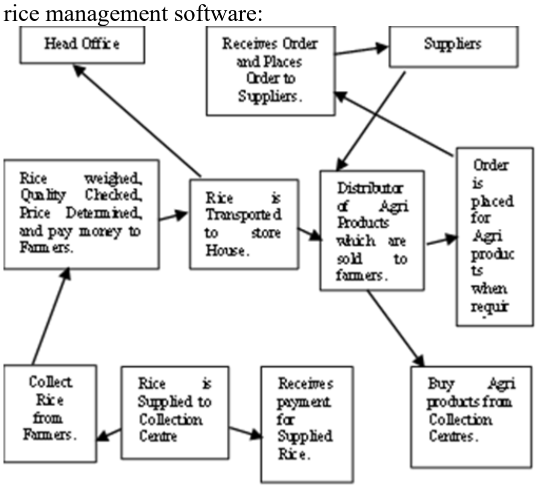

Figure 2. Workflow of the software

Estimate Time for complete effort: Based on levels divided modules and specific modules have their specific screen and features and those features need some days for build or coding and calculate the total no of days build for application.

Table 1. Estimation Time for Coding Effort

Module Name	**Feature Name and Coding effort**	
	Screen Name	***Build effort***
Admin Module	• Login Administration Page • Taluk setup and Administration • District Setup and Administration	1 Days 2 Days 2Days
Purchase Module	• Purchase Rice from Farmers. • Purchase Rice Products from suppliers.	3 Days 2 Days
Sales Module	• Selling Agri Products to Farmers. • Generating Receipts/Bills. • SMS and Email Alerts.	5 Days 2 Days 1 Days
Inventory Module	• Storing Data Related to Rice and Agri Products at Taluk Level. • Storing Data Related to Rice at head office (District level). • Reorder Point. • SMS Alerts regarding out-of-stock products at Inventory.	5 Days 2 Days 3 Days 1 Days
Finance Module	• Financial Setup e.g., General Ledger ID. • Payment Setup. • Receiving Payments.	5 Days 15 Days 5 Days
Dashboard and Reporting	• Rice Availability Report at collection centers and Head Office. • Agri Products availability Report at Collection centers. • Sales and Purchases reports for Agri Products. • Report related to purchase of Rice from Farmers. • Overall Revenue/Finance related information.	3 Days 3 Days 3 Days 3 Days 3 Days
	Total Build (Coding) effort in no of days	69 Days

Table 2. Total Time Estimation for Project

Stages	Formulae used and Calculate Effort in Days	
	Formulae used	***Effort in Days***
Requirement Gathering, Analysis and Design	20 percent of the Project Life cycle effort	23 Days
Development or Coding Effort	60 percent of the Project Life cycle effort	69 Days
Testing	20 percent of the Project Life cycle effort	23 Days
	Total No of Days Required for Project life cycle effort	115 Days
Project Management Effort	10 percent of the Project Life cycle effort i.e., 115	11.5 Days
User testing and Warranty	10 percent of the Project Life cycle effort i.e., 126.5	12.65 Days
Buffer Effort	20 percent of the overall effort i.e., 139.15	27.83 Days
Total No of days Effort Required		166.98 Days Rounded off to 167 Days

Estimation Cost

Predict product development cost from production data above take an example of software development time estimation now continue the software cost prediction from production data. There are different ways to estimate software cost. These ways categorised into two group Non-Algorithmic and Algorithmic. Some popular ways are described in this portion.

Non-Algorithmic Approach: The estimation procedure in these approaches is often carried out in accordance with the study of the prior datasets, and some knowledge about the too early development that are similar to the present project is necessary before applying them. Three approaches were chosen for the assessment in this case because they are more widely used than the other non-algorithmic methods and because numerous articles on their application have recently been published.

Estimation using Analogy: In this approach, similar software projects that have already been finished are identified and estimates of work and cost are made based on their actual costs. At both the overall system level and the subsystem level, estimation based on analogy is carried out. We can determine the cost of an identical project through analysing the outcomes of earlier real-world undertakings. The steps in this process are as follows:

- Choosing an analogy.
- Examining resemblances and differences.
- Analogy quality analysis.
- Offering an estimate.

Expert Judgement: Obtaining guidance from specialists with substantial expertise on similar projects is how estimates based on expert judgement are done. When it's difficult to obtain the necessary information and identify the necessary data, this strategy is typically utilised. The fundamental difficulty with this approach is consultation. This process entails the following steps:

- Each expert receives a valuation form from the coordinator.
- Without consulting other experts, each expert provides his own estimate.
- The coordinator compiles all forms, totals them (with average and median), and requests that experts begin iterating.
- Repeat steps first through second until you receive approval.

Machine learning Models: The majority of methods for estimating the cost of software rely on statistical techniques, which can't deliver solid reasoning behind their conclusions. Machine learning techniques may be useful in this field since they can improve estimation accuracy by establishing estimation rules and practicing them repeatedly [15]. There are two main types of machine learning approaches, which are described in the following subsections.

Neural networks: Each layer in a neural network is made up of various components known as neurons. The weights assigned to the inputs are examined by neurons, who then produce the outputs [7]. The basic objective of estimating is to produce output, which is the actual effort. The greatest option for a software estimation problem is a back propagation neural network since it modifies the weights by contrasting network outputs with real results. Additionally, training is carried out successfully [14]. The majority of research focuses on estimating software costs using neural networks.

Fuzzy Method: The goal of all fuzzy logic-based systems is to imitate human conduct and mental patterns [8]. When making decisions is difficult and the situation is unclear, fuzzy systems are a useful tool. This approach consistently supports the evidence that could be ignored. There are four stages to the fuzzy approach:

- Fuzzification: which generates trapezoidal numbers for words.
- To create a new linguistic phrase in order to develop the complexity matrix.
- Determine the productivity rate and make an attempt at the third stage of linguistic terms.
- Defuzzification, which involves estimating the time needed to finish a task and evaluating alternative approaches.

Algorithmic Approach: Algorithmic cost modelling has been the subject of the majority of research done so far in the area of software cost estimation. Through the use of mathematical formulas that connect expenses or inputs with measurements, this method analyses costs and generates an estimated output. The examination of historical data yields the formulas employed in a formal model. You can improve the model's accuracy by calibrating it to your particular development environment, which simply includes altering the weights of the metrics. These models operate using a unique algorithm. They often require data initially and then use mathematical relations to produce outcomes. These models are now widely used in software estimation techniques. Different models are used for categorization algorithms. Each algorithmic model does the estimation using an equation. The selection of cost components and functionality is related to the variations between the existing algorithmic approaches. Each and every cost component used in these models is Factors affecting the product include needed reusability, needed reliability, needed complexity, and documentation that meets life cycle requirements. Computer factors include platform volatility, main storage restrictions, execution time constraints, and constraints on turnaround. Factors related to personnel include analyst skill, application experience, programmer skill, platform knowledge, language and tool knowledge, and staff continuity. Project-related variables: multi-site development; software tool usage; necessary development.

- Linear Models: As demonstrated in equation 1 below, commonly used linear models have a straightforward structure and trace a simple equation 1:

$$\text{Effort} = \boldsymbol{a_0 + \sum_{i=1}^{n} a_i x_i} \qquad (1)$$

where, in accordance with the project details, $\boldsymbol{a_1}$,…, and an are chosen. Only the numbers -1, 0 and +1 are permitted for $\boldsymbol{x_i}$ in this formula.

- Multiplicative Models: Equation 2 below can be used to express the multiplicative model's form:

$$\boldsymbol{Effort = a_0 \prod_{i=1}^{n} a^{x_i}} \qquad (2)$$

where, in accordance with the project details, $\boldsymbol{a_1}$,..., and an are chosen. Only the numbers -1, 0 and +1 are permitted for $\boldsymbol{x_i}$ in this formula.

- Function Point Cost Estimation: Albrecht (1983) initially proposed the Function Point metric as a way to gauge a project's functioning. According to this method, estimation is carried out by identifying the User Inputs, User Outputs, Logi Files, Inquiries, and Interfaces indicators. Function point may be helpful for software estimations, going by previous experiences, as it could be calculated using the specifications for requirements in the early phases of a project.

Calculate cost for Software development: Provide Estimation of software cost from production data is complicated process, in this process some steps takes place. These steps are given bellow:

Step I: In this step calculate hours means how much hour takes for all development.

Total number of days required = 167.

Number of hours per day = 8 hour.

Total hours required = 8*167 = 1336 hours.

Step II: In the second step take cost for one hour development.

Cost per hour = 10 dollars (estimation value may increase or decrease from time to time).

Step III: In this step multiply calculated no of hours with cost of an hour.

Total cost = 10 * 1336 = 13360 dollars.

Cost in INR = 13360 * 81.90 = 10,94,184 Rs.

RESULTS

According to a study that used production data to predict product development time and cost using function point estimation, the proposed model based on function point analysis was more accurate at doing so than the more conventional estimation techniques. According to the study, the proposed model is a more accurate and trustworthy way to forecast product development time and cost since it has a smaller mean absolute percentage error (MAPE) and root mean square error (RMSE) than the conventional estimation approaches. The findings also indicated that the complexity of the product and the development team's prior expertise were the next two most important predictors of product development time and cost, respectively.

```
# Import required libraries
import statistics

# Define the three time estimates: optimistic, pessimistic, and most likely
optimistic_time = 2
most_likely_time = 4
pessimistic_time = 8

# Calculate the expected time using the PERT formula: (optimistic + 4 * most likely + pessimistic) / 6
expected_time = (optimistic_time + 4 * most_likely_time + pessimistic_time) / 6

# Calculate the variance using the PERT formula: ((pessimistic - optimistic) / 6) ** 2
variance = ((pessimistic_time - optimistic_time) / 6) ** 2

# Calculate the standard deviation using the square root of the variance
std_deviation = variance ** 0.5

# Calculate the confidence interval using the mean and standard deviation: mean ± z * (std_deviation / (n ** 0.5))
n = 10 # number of tasks
z = 1.96 # 95% confidence interval
confidence_interval = z * (std_deviation / (n ** 0.5))

# Calculate the estimated completion time: expected time + confidence interval
estimated_completion_time = expected_time + confidence_interval

# Print the results
print("Expected time:", expected_time)
print("Variance:", variance)
print("Standard deviation:", std_deviation)
print("Confidence interval:", confidence_interval)
print("Estimated completion time:", estimated_completion_time)
```

Figure 3. Time estimation algorithm.

According to the screenshot [Figure 3], the user-friendly dashboard of Time Estimation Algo makes tracking and managing your time a breeze. Make use of the visual project timeline provided by Time Estimation Algorithm to stay on top of your deadlines. Use the thorough time tracking reports provided by Time Estimation Algorithm to make data-driven decisions. Apply Time Estimation Algorithm to increase output and improve workflow.

```
Expected time: 4.333333333333333
Variance: 1.0
Standard deviation: 1.0
Confidence interval: 0.6198064213930023
Estimated completion time: 4.953139754726335
```

Figure 4. Time estimation algorithm result.

Based on the screenshot [Figure 4], Time Estimation Algo's thorough output provides immediate insights into the status of your project. With the precise time estimates provided by Time Estimation Algo, you can decide on the schedule for your project. Real-time progress updates from Time Estimation Algo help you stay in the lead and prevent delays. With the simple to use interface of Time Estimation Algo, you can easily optimize the schedule for your project. With automatic time tracking and reporting provided by Time Estimation Algo, productivity may be increased while saving time.

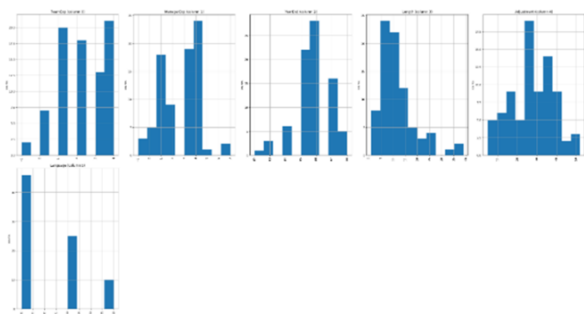

Figure 5. Distribution Graph.

Considering the screenshot [Figure 5], With the in-depth statistical analysis provided by Column Distribution, make data-driven decisions. Utilize Column Distribution to enhance your data insights and decision-making. The sophisticated features of Column Distribution remove guesswork from data analysis.

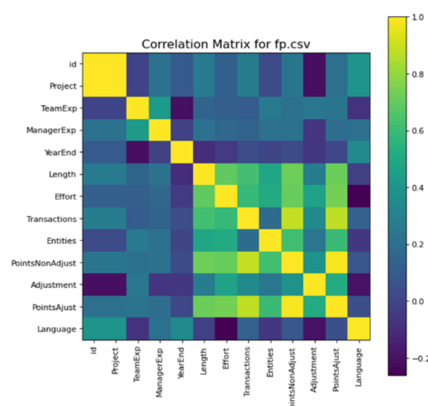

Figure 6. Correlation Matrix Graph.

Make data-driven decisions with confidence with Correlation Matrices' extensive insights, as shown in the screenshot [Figure 6]. Apply correlation matrices to increase the effectiveness of your data analysis and get a competitive advantage. With the help of Correlation Matrices' user-friendly interface, simplify complicated data linkages.

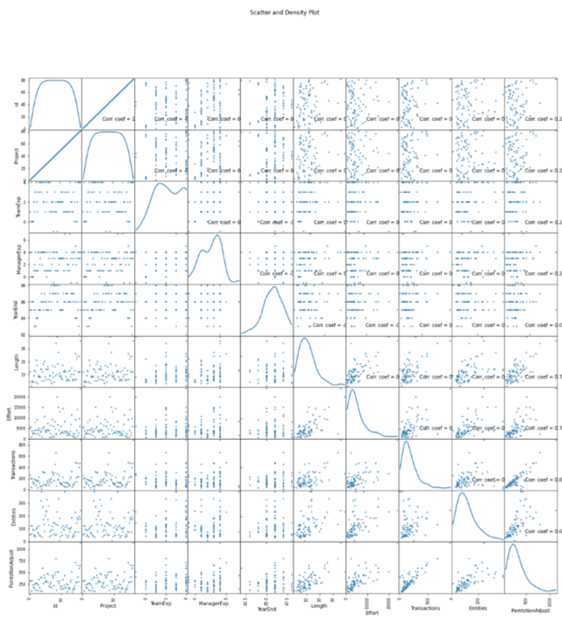

Figure 7. Scatter matrix

Considering the screenshot [Figure 7], Utilize Scatter Matrix's sophisticated scatter plots to investigate data relationships. Using the simple and adaptable charts in Scatter Matrix, simplify complicated data relationships. With the help of the robust statistical analysis capabilities in Scatter Matrix, gain profound insights into data patterns. Use Scatter Matrix's thorough data visualization to confidently make data-driven decisions. Scatter Matrix's automated scatter plot generation and analysis allows for simple data analysis.

CONCLUSION

The study "Predicting Product Development Time and Cost Using Production Data" had a big impact on project management. The success of new product development projects depends on the precise estimation of development time and cost. Expert judgement and historical data are two common traditional techniques of evaluating these variables, however they have drawbacks and are prone to inaccuracy. The efficiency and efficacy of project management can be increased by using production data to forecast product development time and cost. Project managers can create more precise and trustworthy predictive models to forecast development time and cost by utilizing production data. Effective resource allocation, project planning, and risk management can be aided by these models. These models can be used by project managers to spot possible bottlenecks and change the project's resources or timeline as necessary. This strategy can assist in preventing delays, cost overruns, and other problems that could have a detrimental impact on the success of the project. Additionally, using production data in project management can help with ongoing process improvement. Project managers can find possibilities to optimize the product development process, resulting in less time and money spent on development. This strategy can assist businesses in achieving cost reductions and other advantages that improve their bottom line. In conclusion, project management is significantly impacted by the utilization of production data to forecast product development time and cost. It can increase the precision and dependability of predictive models, promote ongoing process improvement, and assist businesses in achieving cost savings and other advantages.

REFERENCES

i. M. Sihala, A. Beer, L. Hodgson, and A. O'Connor, Big Data for Measuring the Impact of Tourism Economic Development [1] Kim, S. M., Lee, S., Lee, H. M., & Moon, I., "Prediction of product development time using production data in the electronics industry.", International Journal of Precision Engineering and Manufacturing, Vol. 14, Issue 11, pp. 1967-1972,2013

ii. Li, Y., Huang, G. Q., & Zhou, L., "A data-driven model for predicting product development time in the automotive industry." International Journal of Production Research, Vol 55, Issue 18, pp.5438-5453,2017

iii. Rausch, E., Schmitt, R., & Schnieder, E., "Analysis of production data to improve design for manufacturability of new products". Procedia CIRP, Vol. 67, Issue No special issue, pp. 481-486, 2018

iv. Zhang, Z., Li, L., & Li, Y., "Predicting product development cost based on production data using a machine learning model." Journal of Intelligent Manufacturing, Vol. 29, Issue 1, pp. 87-9, 2018

v. Sooraj, P., and Pratap K J Mohapatra. "Modeling the 24-h software development process.", Strategic Outsourcing: An International Journal, Vol. 1, Issue 2, pp. 122-141, 2008.

vi. Avadhesh Kumar, Rajesh Kumar, P.S Grover., "Generalized coupling measure for aspect-oriented systems", ACM SIGSOFT Software Engineering Notes, Vol. 34, Issue 3, pp.1-6 ,2009

vii. Abolhassani, M., & Azmoodeh, A., "Predicting software development cost using artificial neural networks.", Journal of Information Science and Engineering, Vol. 34, Issue1, pp. 93-104, 2018

viii. Al-Shehab, A. A., Al-Jarrah, O. Y., & Saeed, F., "Predicting software development effort using fuzzy clustering-based algorithms." Journal of Intelligent Manufacturing, Vol. 30, Issue 4, pp. 1417-1434, 2019

ix. Kwak, Y. H., & Anbari, F. T., "Predicting project cost through risk classification and decision tree analysis.", Journal of Information Technology and Management, Vol. 6, Issue 2, pp. 31-44, 2015

x. Yousuf, M. A., Aslam, F., & Kundi, F. M., "Software cost estimation using multiple regression, neural networks and support vector regression: A comparative analysis." International Journal of Computer Applications, Vol. 119, Issue 11, pp. 1-6, 2015

xi. Nagi, R., & Aktas, E., "Analysis of production data for predicting development time and cost: A review." Journal of Manufacturing Technology Management, Vol. 30, Issue 4, pp. 774-797, 2019

xii. Laursen, G. H. N., & Pearce, J. M., "Cost-effective open source labware: 3-D printing magnetic stirrer and centrifuge." Journal of Laboratory Automation, Vol.20, Issue 5, pp. 517-524, 2015

xiii. Kaur, P., & Singh, Y., "Prediction of software development effort and cost using data mining techniques." International Journal of Computer Applications, Vol. 163, Issue 1, pp. 1-7, 2017

xiv. Liu, L., Zuo, Y., & Li, M., "A software development effort prediction model using deep neural networks." Journal of Intelligent & Fuzzy Systems, Vol. 36, Issue 1, pp. 1-9, 2019

xv. Prasad, B., "Predicting software development effort and cost using artificial neural network: a case study." International Journal of Computer Science and Mobile Computing, Vol. 4, Issue 5, pp. 636-642, 2015.