Predictive Approach towards Software Effort Estimation using Evolutionary Support Vector Machine

Tahira Mahboob, Sabheen Gull, Sidrish Ehsan
Department of software Engineering
Fatima Jinnah Women University
Rawalpindi, Pakistan

Bushra Sikandar
Department of Computer Science
Fatima Jinnah Women University
Rawalpindi, Pakistan

***Abstract*—The project effort measurement is one of the most important estimates done in project management domain. This measure is done in advance using some traditional methods like Function Point analysis, Use case analysis, PERT analysis, Analogous, Poker, etc. Classical models have limitations that they are burdensome to implement, especially when there are LOC (lines of code) or objects' count required in measurement. Sometimes historical information regarding a project is also considered to estimate the projects' effort. But these estimates are then needed to be adjusted. The idea proposed in this research is to determine what factors regarding a project are directly related to the effort estimation. Other than that a model is proposed to predict the effort using minimum number of parameters in software project development.**

Keywords*—Correlation coefficient; Decision tree; Effort Estimation; Evolutionary Support Vector Machine; Software project management

I. Introduction

We need software project cost estimation and project effort estimation to get an idea of the required amount of work to be done and the related amount to be spent on that particular work during the course of work of software product [3]. Effort estimation means that we are going to calculate or forecast the required exertion by the manpower involved in the project in person hours or person months. There are various techniques to estimate effort like Function Point analysis (FP), Use case analysis, PERT analysis, Analogous, Poker, etc. [17][18]. The classical ways of calculating effort are still in practice and providing services in software project management. Although these methods have limitations like FP is related to the size of project and has some long calculations, need to specify adjustment factor which if wrong than it leads to false FP analysis. Similarly all other techniques have their own merits and demerits.

Machine learning is making advancements because of its algorithms used for making predictions. There are number of algorithms which are handy in prediction and forecasting. In calculating effort by any of the above mentioned method, we require large datasets with maximum information regarding a project. What we need now is fast calculation with accurate results even if we have less information. This research paper makes use of machine learning to predict the effort if we know certain parameters. What parameters can help in prediction is determined by linear correlation and decision tree.

II. Literature Review

In a company, its data is the most important entity to be considered as a source of information and a source of production of new products by forming a firm connection of data with management and expertise. The company's competitive position is furnished by the launch of new products. Most of the companies fail to develop new successful products due to inability to develop fine schedule and development plans in New Product Development (NPD). CPM (Critical Path Method) and PERT (Program Evaluation and Review Technique) are outmoded approaches in project scheduling. For non-linear data, a system with combination of Neural Networks and Fuzzy Neural Networks (intelligent systems) are to be used. [1]

Cost estimation of a project is done to complete the project within specified budget or before that. Two types of models of cost estimation are there; one is algorithmic and the other is non-algorithmic. Algorithmic types have defined formulae for cost estimation calculation while non-algorithmic types have no defined formulae for this purpose. Then there are evaluation techniques to determine the difference between estimated and actual cost like root mean square error (RMSE) [2]

We need software project cost estimation and project effort estimation to get an idea of the required amount of work to be done and the related amount to be spent on that particular work during the course of work of software product. Schedule, effort and quality are the three corners of a "magic triangle" and to maintain a balance between these three aspects is a tough job yet essential job in software projects. Time management has its drawbacks in all cases whether it is accurately estimated, underestimated or overestimated. [3]

Human Resource is an asset of software development firm. Due to scarcity of skilful developers, software managers find it challenging to schedule the time and cost for the limited developers resource to fill in the total time and cost of the whole project. "Who must do what, and when?" is the question to be answered while scheduling tasks within specified time for developers. Scheduling is a hard-to-do thing for software projects as 1) software is not material hence its progress is difficult to monitor. 2) There are only rough estimates of each development life cycle. 3) For those activities, which run parallel, when interfaced with each other their other components are also changed to support this sudden interface change, hence taking more time than estimated. [4]

The development of techniques to schedule software projects is a challenging task. The challenge is not only to schedule project but also the human resource according to project. The existing models are designed such that to make project scheduling more accurate, human resource allocation is made to suffer. Event based schedule (EBS) and Ant Colony Optimisation (ACO) algorithm combination is an innovative plus flexible approach for scheduling software projects. In EBS, events are the times when employees start or leave the project or when resources are released. [5]

There are several evolutionary algorithms to solve project scheduling problems but each algorithm has its own way of functioning, how they perform depends on the way they are designed. The design of proposed evolutionary algorithm is made better by combining it with normalisation techniques and it improves practical effectiveness. The project scheduling problem becomes problematic whenever the project is large scaled, employees has dedication to a task up to specific limit, there is a large space of allocation of tasks for employees. So it is a need of the time and management to automate the process of project scheduling so that each employee gets the optimised workload. [6]

The dimension of human resource working on a software project is more crucial as compared to the technical dimensions in that particular project. Each individual human plays the key role in the software development life cycle deliverables delivery. The performance of a project, its success and failure cases depend on the personality of the individuals involved in it. “Belbin Team Roles Assessment Tool” is a tool to assess the personality type of individuals and in this work, it is used to support the argument of human personality impact in one way or the other on information technology projects. [7]

In-house software, outsource software and off-the-shelf software are the various types of software. Whenever outsource software projects are not being monitored and managed by appropriate ways, there are possibilities that the threats and suspicions associated with project will be handled based on individual knowledge of the work and personal ideas. Risk management in software engineering is a domain to handle risks effectively and managed in a timely fashion. This indicates that project manager and other team members must train themselves to handle the risks associated with each stage during the software development life cycle. [8]

Performance evaluation of component based software is actually the performance evaluation of individual component‘s performance. There are specific models of development process of component based software which are specialised to best utilise the plus points of component based systems i.e. reuse of components and division of labour. The performance of component based systems is hard to find because in a running environment, components work the way they are deployed in a system according to the system and developers of components have no idea about the usage profile of particular component, or how they will be used. [9]

Decision Support maker involve multiple actives and variable to analyse the performance of the project. Project Performance Measurement System (PPMS) involve variable and manager to tackle with volume of data to evaluate project performance. Decision Support categorised performance of the project by taken current state and decision maker view and if deviation is detected then management team will analyse reason per project manager performance interest. Performance of the project is basic support for the decision maker. MACBETH tool is used for analysing performance of the subsequent data of (PPMS) with respect as per project manager. [10]

The Selection of Project Manager is a critical task that contains multiple criteria that must consider such as past project performance, suitability and nature of project and qualification of candidate. Project Performance are often measure by the cost, time and technical description. Decision Make Support System (DMSS) is based on the pervious performance for the selection of the candidate as project manager by using the ranking method of the past project. Different Ranking methods are presented for the ranking previous project that should fulfill specific as well as multiple requirements for ranking. DMSS analyse tool enable ranking the candidate on base of the previous project success DMSS also consider History of Candidate, Antiquity of Managing Project and outcome Performance of the Project. [11]

In I.T based business Projects Knowledge management involves understanding of three areas such as knowledge of Technical, Organisational and business value Solution. Empirical Study shows that business esteem is better accomplished in IT-empowered project change with active knowledge management in three domain areas moreover empirical study also shows that knowledge management explain 38% of the project value performance. It has been proposed that knowledge management have significant impact on project performance. Performance against other essential targets, e.g. budget and schedule has positive impact of knowledge management by to managing project. [12]

Paper presented a Hyper-Cube framework for Ant Colony Optimisation ACO to solve the Software Project Schedule Problem by using System Max-min algorithm. Hyper-Cube Framework improves the performance of the Algorithm. Max-min Algorithm Results Compared with Genetic and ACO Algorithm and it is observed that Max-min Algorithm give better results for small instances and attain low cost and duration of the project. [13]

Scheduling of Software Projects is Critical task in the process of software development. Software Project Schedules SPS involves human resource and people intensive activity. In SPS two goals arises such as reduction of duration and cost. Paper analyses eight multi object algorithm scalability for large project in the competitive software industry. It has been analysed that PAES Algorithm shows best scalability among eight multi object algorithm. [14]

Schedules in Critical chain project management not necessary always meet dates for completion of task. This paper proposes critical chain schedule optimal approach. The Problem related to critical chain schedule is taken in to account that involve duration uncertainty and resource constraints. CCPSP (Critical Chain Project Scheduling Problem) is expressed and then DE algorithm is applied as per the characteristic of the CCSPSP to find out solution of the problem new strategies of differential evolution (DE) algorithm follow to obtain fast coverage and global exploration ability. Critical chain method Implementation for large scale project is difficult and complex task. It is observed that modified algorithm DE achieve global coverage effectively for CCPSP. [15]

Considering project complexity, this paper throws light on the how project success is related with risk management while correlating the soft and hard skills. For this purpose, a hypothesis was tested where 263 projects were involved where interviews were conducted from project managers and risk managers and analysed. A structural model was proposed that correlated the soft and hard skills of risk management with project success. It was concluded that soft side has 10.7% effect on the project success and hence, cannot be neglected while it also supports the hard side in addition to the correlation that is 25.3% effect on the hard side. [16]

For Estimation of effort, schedules, cost, and size of software projects use case method is use. Use case method depends on the use case diagrams for estimation. Estimation of Software Project helps to determine cost, effort size and schedule of software projects. For this purpose, use case method use widely use case method have also limitation that it may affect estimation accuracy. Some techniques have been address to solve the problem related to the estimation, Techniques such as neural networks and fuzzy logics can be used for better accuracy of the estimation to improve use case points methods. [17]

Use case point (UCP) technique is use for the effort estimation of the software project. UCP technique gives a better and accurate estimation of effort but still have certain limitation with it due to which UCP is not use accepted by the Software Industries. To Enhance the Predication and accuracy for the effort estimation Random Forest Technique is use to overcome constraint of the UCP technique. RF is Machine Learning Technique that utilises approach of Use Case Point method. RF combines results obtain by different model which give more accurate effort estimation. Results shows that RF Technique gives more accurate predication as compare to other techniques such as SGC (Stochastic Gradient Boosting), LLR (Log-Linear Regression), MLP (Multi-Layer Perceptron), RBFN (Radial Basis Function Network) [18]

Cost and Performance are important factor for reducing the exploration time of target system of the design space at high level of abstraction in Software and Hardware code-signs. Estimation methods, such as S-Graphs are in software/hardware code-sign with POLIS at different abstraction level. S-Graph Level method provide better accuracy in software synthesis for optimisation and CFSM-level provide better accuracy for generation of schedule and automatic portioning into software and hardware. Experimental results show that S-Graph method when compared to assembly level have accuracy within (-20% to +20) and for maximum time estimation CFSM-level accuracy range is (-10% and +25%) [19]

Cost estimation for software projects are difficult task in early lifecycle of software development process. Deterministic methods are used for the software project evaluations. A Comparative analysis has been done between COCOMO and Fuzzy Logic for cost estimation of software. Fuzzy Logic is a new approach which is used to investigate Software Cost Estimation (SCE). Multiple Membership functions such as Trapezoidal, Triangular, Generalised Bell, Sigmoidal and Gaussian has been to analyse the Fuzzy Logics (FL) for the SCE. FL shows a better performance in comparison to the COCOMO model when tested for dataset of software projects. [20]

III. Experimental Setup

SAMPLE: A sample of 81 projects is taken from NASA Repository [21]. The actual dataset contains 12 parameters with varying types. They are mentioned in Table 1.

TABLE I. All Dataset Parameters with Types

Parameters	Data type
Project	Numeric
TeamExp	Nominal
ManagerExp	Nominal
YearEnd	Numeric
Length	Numeric
Effort	Numeric
Transactions	Numeric
Entities	Numeric
PointsAdjust	Numeric
Envergure	Numeric
PointsNonAjust	Numeric
Language	Class/Categorical {1,2,3}

The parameters mentioned in Table 1 are the ones in actual dataset. All of them will not be required in our work as the actual dataset was used in Analogous technique of effort estimation [22]. The parameters “TeamExp” and “ManagerExp” are numerical but they have nominal values from 0 to 4 and 0 to 7 respectively. The dataset contains 81 instances but 4 of them have incomplete information so 77 instances are in workable form.

In our research “PointsAdjust” and “PointsNonAdjust” are not used as they are specific to analogous method of effort calculation.

The parameters involved to calculate effort are given in Table 2

TABLE II. DATASET PARAMETERS

Parameter	Explanation
Team Experience	Experience of the working team
Manager Experience	Experience of the manager in charge
Length	Length of project from start till completion
Transactions	Queries handled per module
Year End	The year in which the project ended
Entities	Modules in project
Languages	Language of development. It is assumed that the categories {1, 2, 3} represents three types of languages as 1 being the most easy to 3 being most difficult.
Envergure	Scope of project

Tables 3 to 5 depicts the dataset categories and population spread in certain classes of data

TABLE III. TEAM EXPERIENCE POLYNOMIAL DATA SUMMARY

Index	Nominal value	Absolute count	Fraction
1	4	21	0.25925925925925924
2	1	20	0.24691358024691357
3	2	18	0.2222222222222222
4	3	13	0.16049382716049382
5	0	7	0.08641975308641975
6	?	2	0.024691358024691357

TABLE IV. MANAGER EXPERIENCE POLYNOMIAL DATA SUMMARY

Index	Nominal Value	Count	Fraction
1	4	24	0.2962962962962963
2	3	19	0.2345679012345679
3	1	18	0.2222222222222222
4	2	9	0.1111111111111111
5	0	5	0.06172839506172839
6	?	3	0.037037037037037035
7	7	2	0.024691358024691357
8	5	1	0.012345679012345678

TABLE V. SUMMARY OF INTEGRAL DATA PARAMETERS

	Min Value	Maximum value	Average	Deviation
Year End	83	88	85.790	1.148
Length	1	39	11.716	7.400
Effort	546	23940	5046.309	4418.767
Transactions	9	886	179.901	143.315
Entities	7	387	122.33	84.882
Envergure	5	52	27.630	10.592
Languages	1	3	1.556	0.707

The Dataset has pre-calculated efforts required in each project. Following figures (Figures 1 to 8) show the effort plot across each parameter. Effort is considered to be a dependent variable (y-axis) and other parameters as independent variable (x-axis). The scatter plot is chosen for this demonstration because it will give a rough idea about correlations and dependencies.

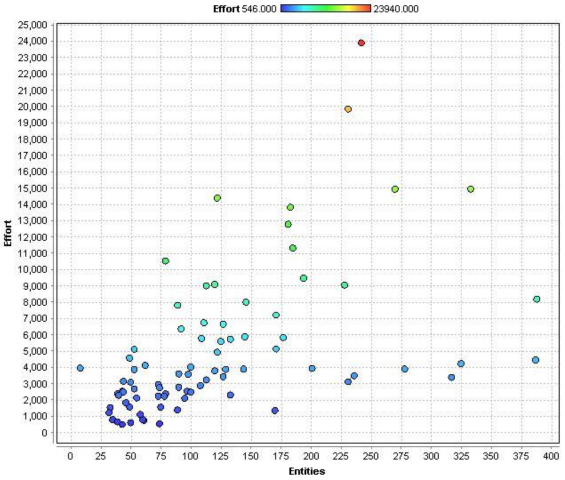


Fig. 1. Effort across Entities scatter plot

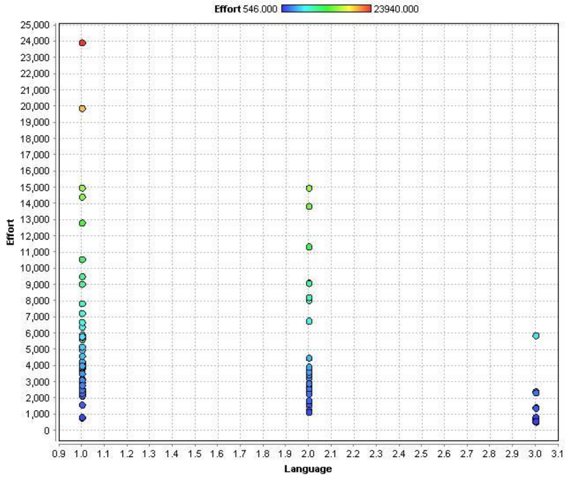


Fig. 2. Effort across Language scatter plot

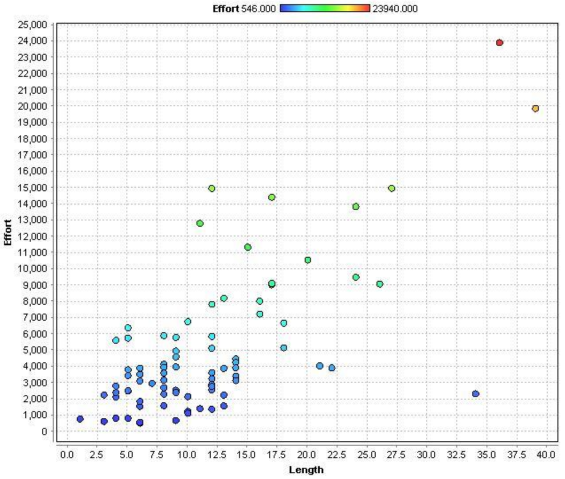


Fig. 3. Effort across Length scatter plot

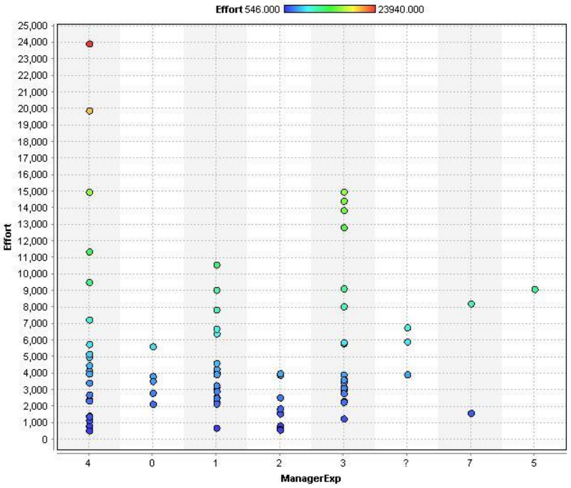


Fig. 4. Effort across Manager Experience scatter plot

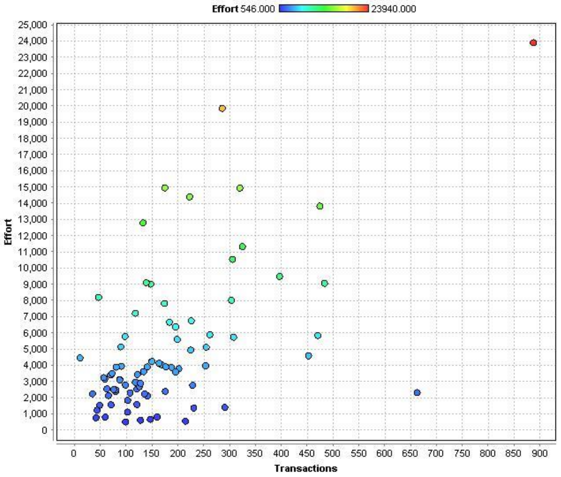


Fig. 5. Effort across Transactions scatter plot

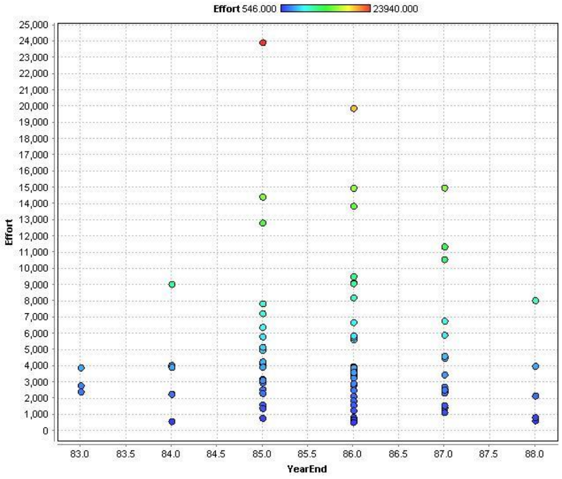


Fig. 6. Effort across Year End scatter plot

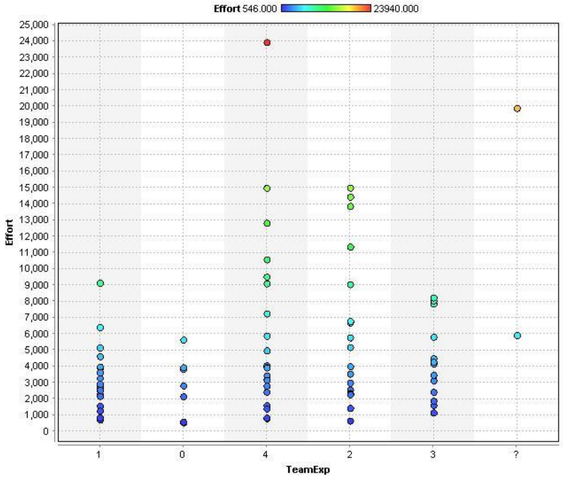


Fig. 7. Effort across Team Experience scatter plot

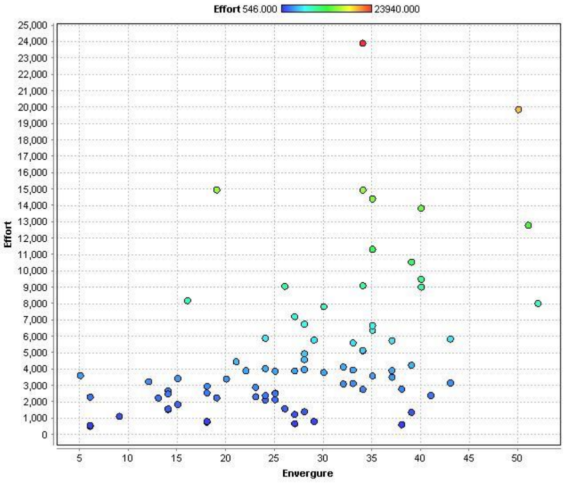


Fig. 8. Effort across Envergure scatter plot

IV. METHODOLOGY

The methodology so followed is that the data is first tested for correlation. For that a correlation matrix is created to get an idea about the relation of each attribute with other attributes. The threshold value is set to 0.5. All correlation values less than 0.5 are discarded i.e. those attributes are not correlated. The correlation matrix can be viewed in Figure 9.

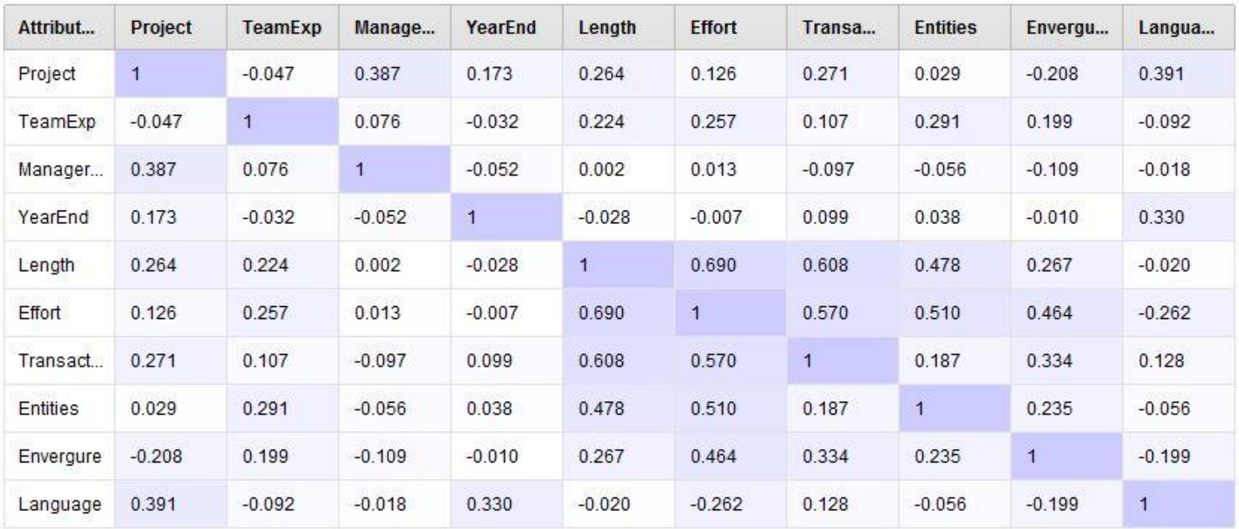
TABLE VI. SHORTLISTED ATTRIBUTES

Correlation of selected attributes	
Effort and Length	0.690
Transaction and Length	0.608
Transaction and Effort	0.570
Entities and Effort	0.510

The decision tree is made using Effort as a key label. The decision tree is used to shortlist the attributes, the matching attributes are then finally shortlisted. The tree can be viewed in Figure 10 and the shortlisted attributes can be seen in Table 6.

From the tree we can see that the attributes playing role in decision making are the same as the output of correlation matrix i.e. Entities, Transactions and Length. Predictions will be made using these three parameters as they are correlated with Effort.

After parameter selection several prediction models where developed and their performance was measured to calculate the correlation of predicted effort and actual effort. These models were preferred over other machine learning models because they were able to handle numerical data and as per previous tests to determine the parameters for prediction, all of them are numerical in nature (Table 7). The parameter "Effort" itself is a numerical attribute in this dataset.



Attribut...	Project	TeamExp	Manage...	YearEnd	Length	Effort	Transa...	Entities	Envergu...	Langua...
Project	1	-0.047	0.387	0.173	0.264	0.126	0.271	0.029	-0.208	0.391
TeamExp	-0.047	1	0.076	-0.032	0.224	0.257	0.107	0.291	0.199	-0.092
Manager...	0.387	0.076	1	-0.052	0.002	0.013	-0.097	-0.056	-0.109	-0.018
YearEnd	0.173	-0.032	-0.052	1	-0.028	-0.007	0.099	0.038	-0.010	0.330
Length	0.264	0.224	0.002	-0.028	1	0.690	0.608	0.478	0.267	-0.020
Effort	0.126	0.257	0.013	-0.007	0.690	1	0.570	0.510	0.464	-0.262
Transact...	0.271	0.107	-0.097	0.099	0.608	0.570	1	0.187	0.334	0.128
Entities	0.029	0.291	-0.056	0.038	0.478	0.510	0.187	1	0.235	-0.056
Envergure	-0.208	0.199	-0.109	-0.010	0.267	0.464	0.334	0.235	1	-0.199
Language	0.391	-0.092	-0.018	0.330	-0.020	-0.262	0.128	-0.056	-0.199	1

Fig. 9. Correlation Matrix

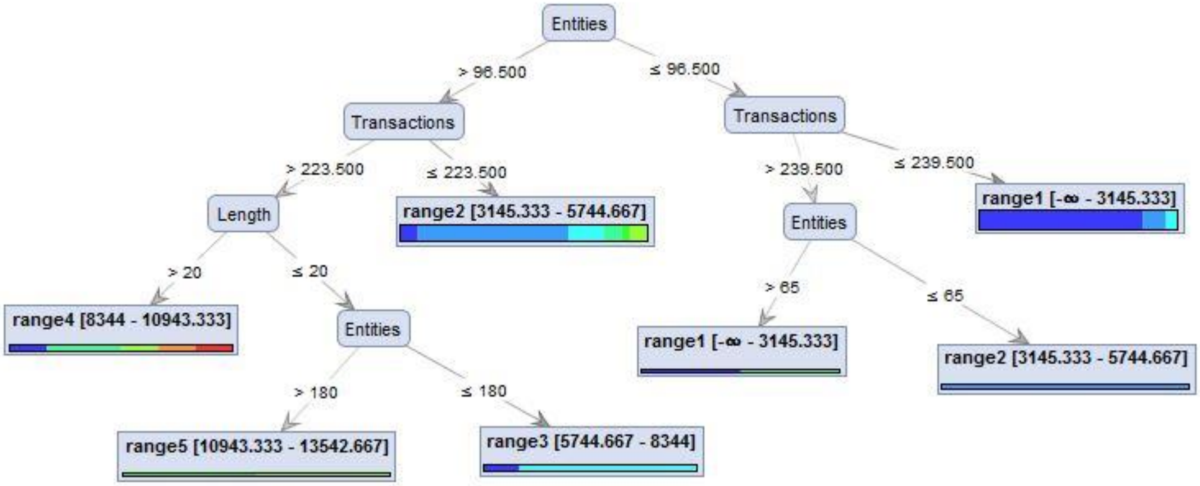


Fig. 10. Decision Tree

TABLE VII. OTHER PREDICTION MODELS CORRELATIONS

Comparison with other Prediction methods					
Prediction Models	**R**	**R^2**	**Spearman Rho**	**Kendall Tau**	
Gradient Boosted	0.850	0.722	0.835	0.673	
Deep learning	0.892	0.796	0.800	0.610	
Support Vector Machine	0.912	0.832	0.817	0.638	
Linear Regression	0.913	0.833	0.813	0.629	
Vector Linear Regression		0.913	0.833	0.813	0.629
Generalised Linear Model		0.914	0.836	0.813	0.629
Neural Network (learning rate 0.3, training cycles 500)	0.927	0.860	0.806	0.619	
Polynomial Regression	0.927	0.859	0.830	0.648	
Evolutionary Support Vector Machine	0.965	0.930	0.938	0.800	

1) Gradient Boosted:

Boosting is to add prediction models in ensemble serially. The Gradient Boosting technique adds new models in current model to increase the accuracy of the target Prediction variable. New base learners are established which are meant to be highly correlated with negative gradient of loss function. [23]

2) Deep Learning:

Deep learning is a type of neural networks [24]. Depending upon the need to make weights more accurate, NN might require long chains of computational phases. Each stage may transform the cumulative stimulation of network. Here is when deep learning comes in action by giving due credit to large number of stages as per requirement. [24]

3) Support Vector Machine:

Support vector machine is a linear classifier that divides data into two classes by hyper plane [25]. The training data in SVM is actually vectors in space. Those training points that are close to hyper plane are support vectors [26].

4) Linear Regression:

Regression is a statistical method to analysis a relationship between variables. Linear regression is so called because it gives a straight line as a result between depending variables. In linear regression, predictor x is kept fixed, regressor y is than analysed. Any change in y demonstrates the external factors which determine its behaviour represented as ε. [27]

5) Vector Linear Regression:

Vector regression is efficient regression model. It includes the concept of support vectors i.e. training points near the separating boundary. Vector regression adjusts ε of the loss function in linear regression given training set and target variable to predict. [28]

6) Generalised Linear Model:

A general assumption in linear model is that data being modelled will be on continuous measure. Hence linear models cannot handle binary data or data that represents count. To overcome that limitation generalised linear model is used. [29]

7) Neural Networks:

Neural Networks have connected pre-processors called neurons. Input neurons get activated when they receive sensation from environment. Remaining neurons need weights from already activated neurons for activation. For learning process, the main task for Neural Networks is to find weights that will give best desired results. [24]

8) Polynomial Regression:

Polynomial regression is a type of linear regression in which there might be a case that either the predicting variables are non-linear or the relationship between the predictors and regressor is curvilinear. They are also linear models but just have higher powers f polynomial involved [30].

9) Evolutionary Support Vector Machine:

Using the above three parameters, evolutionary support vector machine model is developed to predict the value of "Effort" parameter. Effort is considered as a dependent variable whereas Entities, Transactions and Length are Independent variable. The developed model can be viewed in Figure 11.

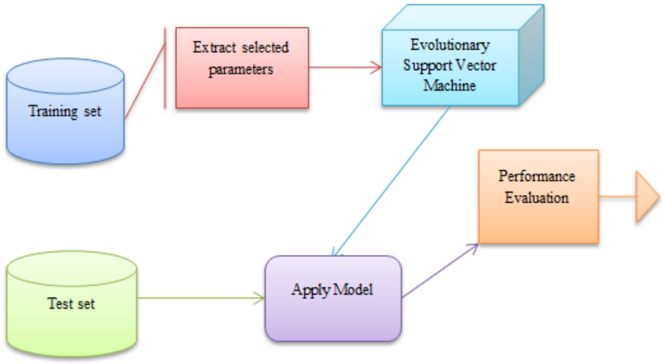

Training set
Extract selected parameters
Evolutionary Support Vector Machine
Performance Evaluation
Test set
Apply Model

Fig. 11. Evolutionary Support Vector Machine model for prediction

Support Vector machine is a supervised learning technique. In support vector machine, training examples which are closest to the hyper plane are called support vectors. There is hyper plane which separates out the classes in datasets such that they have maximum distance in between them SVM is not only used for classification but for regression also, where regression is how we separate out the data based on correlation. The input here will be elements of set X where X is a training set with x_1, x_2, x_n being the input sample parameters which in this case are the shortlisted parameters.

The out of SVM is a set of weights for input parameters needed for prediction. These weights adjust the value of elements of X and make predictions more accurate.

Equation for hyper plane is

$$w^t x + b >= 0 \text{ for } d_i = +1 \tag{1}$$

$$w^t x + b < 0 \text{ for } d_i = -1 \tag{2}$$

Where w is a weight vector, x is input, b is bias and d being the distance between the plane and vector point (data point).

In evolutionary support vector machine there are evolutionary strategies for fast optimisation. Old SVM are not able to optimise the function if it encounters the negative or non-positive kernel function. Hence a better approach would be to introduce *evolution strategies* (ES). Using this method, weights can be directly optimised. Evolutionary support vector machines simply use ES. In it there is real esteemed vectors$'\alpha'$, Gaussian distributed random variables having standard deviation $C/10$ and this random variable is used for transformation. Initial vectors are random with $0 \leq \alpha \leq C$

The training set comprises of 81 observations from the dataset, the proposed model is trained for those values. The test set checked across the proposed model comprised of 20 observations.

V. Observations

N_c is number of elements in same order and N_d is number of elements in different order.

If the value of Kindall coefficient is greater than 0.5 than it implies high correlation between variables.

These correlation values are calculated for the actual Effort and the Predicted effort.

From the results of prediction we can deduce that in predicting the effort required in a project in person hours, the crucial parameters needed are the transactions performed within a module of the developed software, the length in moths of each project and the entities in development model.

VI. Conclusion and Future Work

This paper is written to find a solution to estimate software project effort using minimum information from a project history of the same organisation. To find out the best parameters for prediction two methods were used, namely, (1) correlation matrix, and (2) decision tree. Two methodologies were considered just to verify the results of each test. Both tests generated the same results and three parameters were selected for prediction. Several prediction models were built and trained for selected parameters. The evaluation results revealed that Evolutionary Support Vector Machine gives the best prediction results. The final result is that if we know only the number of entities in a project, the transaction of that project and the length of project in months, we can predict the effort required for that project using Evolutionary support vector machine.

In the future other algorithms, such as Bagging and Ensemble algorithms can be implemented for predicting effort in projects. Results would be observed against each in order to identify better performance keeping in view the accuracy and prediction estimates.

The value of correlation if greater than +0.70 than it indicated high positive correlation which in our case is 0.965.

The formula for Spearman Rho is

$$\rho = 1 - \frac{\partial \sum d_i^2}{n(n^2-1)} \tag{3}$$

P is spearman roh coefficient, d is the difference between x and y values and n is the number of elements in a dataset.

Is value of p is greater than 0.5 than it is highly correlated.

The formula for Kindall correlation is

$$\tau = \frac{n_c - n_d}{\frac{1}{2}n(n-1)} \tag{4}$$

References

[1] M. Relicha and W. Muszynskib, "The Use of Intelligent Systems for Planning and Scheduling of Product Development Projects", Procedia Computer Science, vol. 35, pp. 1586-1595, 2014.

[2] S. Kumari and,. Pushkar, "Performance Analysis of the Software Cost Estimation Methods: A Review", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, no. 7, pp. 229-238, 2013.

[3] J. G. Borade and V. R. Khalkar, "Software Project Effort and Cost Estimation Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, no. 8, pp. 730-739, 2013.

[4] F. Padberg, "Scheduling software projects to minimize the development time and cost with a given staff", Proceedings Eighth Asia-Pacific Software Engineering Conference IEEE, pp. 187-194, 2001.

[5] W. Chen and J. Zhang, "Ant Colony Optimization for Software Project Scheduling and Staffing with an Event-Based Scheduler", IEEE Transactions on Software Engineering, vol. 39, no. 1, pp. 1-17, 2013.

[6] L. L. Minku, D. Sudholt and X. Yao, "Improved Evolutionary Algorithm Design for the Project Scheduling Problem Based on Runtime Analysis", IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, vol. 40, no. 1, pp. 83-102, 2014.

[7] N. N. Vit6 Ferreira and J. J. Langerman, "The Correlation Between Personality Type and Individual Performance on an ICT Project", in The 9th International Conference on Computer Science & Education (ICCSE 2014), Vancouver, Canada, 2014, pp. 426-430.

[8] J. H. Yahaya, N. Fazila Hamzah and A. Deraman, "Evaluating Vendor's Performance in Outsource Software Development Risks Using Analytic Hierarchy Process Technique", in 2014 8th Malaysian Software Engineering Conference (MySEC), Malaysian, 2014, pp. 61-66.

[9] H. Koziolek, "Performance evaluation of component-based software systems: A survey", Performance evaluation, vol. 67, no. 8, pp. 634-658, 2010.

[10] G. Marques, D. Gourc and M. Lauras, "Multi-criteria performance analysis for decision making in project management", International Journal of Project Management, vol. 29, no. 8, pp. 1057–1069, 2011.

[11] Y. Hadad, B. Keren and Z. Laslo, "A decision-making support system module for project manager selection according to past performance", International Journal of Project Management, vol. 31, no. 4, pp. 532-541, 2013.

[12] Y. Hadad, B. Keren and Z. Laslo, "How knowledge management impacts performance in projects: An empirical study", International Journal of Project Management, vol. 32, no. 4, pp. 590-602, 2014.

[13] B. Crawford, R. Soto, F. Johnson, E. Monfroy and F. Paredes, "A Max–Min Ant System algorithm to solve the Software Project Scheduling Problem", Expert Systems with Applications, vol. 41, no. 15, pp. 6634–6645, 2014.

[14] F. Luna, D. L. González-Álvarez, F. Chicano and M. A. Vega-Rodríguez, "The software project scheduling problem: A scalability analysis of multi-objective metaheuristics", Applied Soft Computing, vol. 15, pp. 136–148, 2014.

[15] W. Peng and M. Huang, "A critical chain project scheduling method based on a differential evolution algorithm", International Journal of Production Research, vol. 52, no. 13, pp. 3940-3949, 2014.

[16] M. Monteiro de Carvalho and R. Rabechini Junior, "Impact of risk management on project performance: the importance of soft skills", International Journal of Production Research, vol. 53, no. 2, pp. 321-340, 2015.

[17] A. Bou Nassif, L. Fernando Capretz and D. Ho, "Enhancing Use Case Points Estimation Method Using Soft Computing Techniques", Journal of Global Research in Computer Science, vol. 1, no. 4, pp. 12-21, 2016.

[18] S. Mouli Satapathy, B. Prasanna Acharya and S. Kumar Rath, "Early stage software effort estimation using random forest technique based on use case points", IET Software, vol. 10, no. 1, pp. 10-17, 2016.

[19] K. Suzuki and A. Sangiovanni-Vincentelli, "Efficient Software Performance Estimation Methods for HardwareEoftware Codesign", in 33rd Design Automation Conference Proceedings, Las Vegas, NV, 1996, pp. 605-610.

[20] I. Maleki, L. Ebrahimi, S. Jodati and I. Ramesh, "ANALYSIS OF SOFTWARE COST ESTIMATION USING FUZZY LOGIC", International Journal in Foundations of Computer Science & Technology (IJFCST), vol. 4, no. 3, pp. 27-41, 2014.

[21] M. Shepperd, "Promise Software Engineering Repository", The PROMISE Repository of Software Engineering Databases. School of Information Technology and Engineering, University of Ottawa, Canada, 2005. [Online]. Available: http://promise.site.uottawa.ca/SERepository/datasets/desharnais.arff.

[22] J. Desharnais, Analyse statistique de la productivitie des projets informatique a partie de la technique des point des fonction, 1st ed. 1988.

[23] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial", 2013. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnbot.2013.00021/full.

[24] J. Schmidhuber, "Deep learning in neural networks: An overview", Neural Networks, vol. 61, pp. 85-117, 2015.

[25] S. Tong and D. Koller, "Support Vector Machine Active Learning with Applications to Text Classification", Journal of Machine Learning Research, pp. 45-66, 2001.

[26] M. Adankon and M. Cheriet, "Support Vector Machine", Encyclopedia of Biometrics, pp. 1504-1511, 2015.

[27] D. Montgomery, E. Peck and G. Vining, Introduction to Linear Regression Analysis, 1st ed. .

[28] B. Gu, V. S. Sheng, Z. Wang, D. Ho, S. Osman and S. Li, "Incremental learning for ν-Support Vector Regression", Medical Biophysics Publications, 2015.

[29] D. Bates, Generalized linear models, 1st ed. 2010.

[30] Polynomial Regression Models, 1st ed. .

[31] I. Mierswa, "Evolutionary Learning with Kernels: A Generic Solution for Large Margin Problems", Seattle, Washington, USA, 2006.