Journal of Computer Science 8 (2): 195-199, 2012
ISSN 1549-3636

Detection of Aberrant Data Points for an effective Effort Estimation using an Enhanced Algorithm with Adaptive Features

[1]Malathi S. and [2]S. Sridhar
[1]Department of CSE,
Sathyabama University Chennai, Tamilnadu, India
[2]Department of CSE and IT,
Sathyabama University Chennai, Tamilnadu, India

Abstract: Problem statement: The spiraling growth of IT industry has witnessed an unprecedented change in the software development paradigm, from algorithmic models to machine learning techniques. At present, there are no standard methods to predict the accuracy of software cost estimation, which is an important goal of the software community. **Approach:** This study proposes a simple and systematic algorithmic procedure for analogy based software cost prediction to detect the aberrant data points. The algorithm is analyzed and correlated with the Desharnais and NASA datasets containing all adaptive features with numerical and categorical variables. **Results:** The interpreted curves using the above datasets depict a discernible anomaly for the dataset having more categorical variables, thereby indicating the erroneous data points. **Conclusion:** The elimination of aberrant data points using the new algorithmic method improves the accuracy of software cost estimation using historical data sets.

Key words: algorithmic method, cost prediction, categorical variables, adaptive features, aberrant datapoints

INTRODUCTION

Software cost estimation plays a critical role to predict the effort and evaluate the feasibility of the project based on the costs involved. It is invariably essential that the technique used to estimate the cost should produce accurate results for decision making. Numerous methods have been proposed for the effort estimation, which fall into one of the three broad categories viz., expert judgment, algorithmic models and machine learning (Mendes *et al*., 2003).

This study presents an enhanced algorithm that predicts the software cost using the Analogy-X method by identifying the most appropriate and stable set of project sets which aid in the prediction of the effort involved. The enhanced algorithm successfully produces accurate results based on the concept of analogy.

Related work: Over the past three decades, intensive research in the area of effort prediction has provided several approaches for cost estimation. However, algorithmic models for effort estimation, based on statistic and regression analysis, constitute the major proportion of the research. Although several new approaches have been proposed, software effort estimation heavily rely on the local factors such as Lines Of Code (LOC), Function Points etc.

Though most research projects dealing with effort estimation have adapted the traditional algorithmic models, there has been a significant development in the exploration of machine learning or non-algorithmic models. Estimation of effort can be carried out in an efficient and accurate manner by collecting relavent software data terms.For the collection of such data,agile methodology (Omar *et al*., 2011) can be employed which is an accurate, incremental and an iterative one. Li *et al*. (2009a) have proved that the use of neural nets for the prediction of software reliability outperform the traditional statistical systems.

Azzeh *et al*., (2011) have improved the performance of analogy at the early stage of identification process by using fuzzy numbers. Huang *et al*., (2007) developed a fuzzy neural network by applying artificial neural networks to fuzzy inference processes. Le-Do *et al*., (2010), have proposed a scheme for filtering the Inconsistent Software Project Data for Analogy-based effort Estimation.

Another sphere of software cost estimation is the Case Based Reasoning (CBR) which is currently a popular alternative procedure for algorithmic and

Corresponding Author: Malathi, S., Department of CSE, Sathyabama University Chennai, Tamilnadu, India

machine learning methodologies. Mukhopadhyay *et al*., (1992) discuss about the early study using a hybrid case based reasoning and rule based system. Data-intensive analogy based software effort prediction gained popularity in the late 1990's by Shepperd and Schofield. Recently, Kocaguneli *et al*. (2011) proposed a method to improve Analogy based software estimation. Empirical experiments using the tools such as ESTOR and ANGEL (Keung, 2008) show that the estimation by analogy is a viable alternative to predict accuracy and flexibility.

MATERIALS AND METHODS

Analogy: Analogy based effort estimation method longs to machine learning category. The basic idea of analogy prediction (Jorgensen and Shepperd, 2007) is shown in Fig.1.

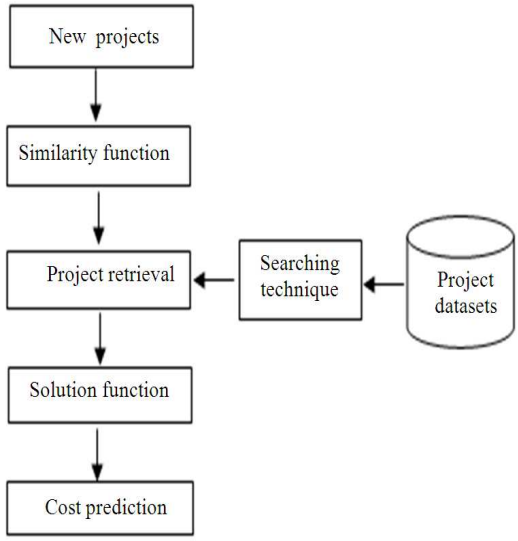

Fig. 1: Analogy Based cost Estimation Flow chart

"Projects that are similar with respect to project and product features such as size and complexity will be similar with respect to project effort".The steps involved are:

- Select the historic projects and find the cost drivers.
- Find the similarities between the new and the target projects.
- Recognize the historic projects that are analogous to the target.
- Set the effort of the historic project to estimate the effort of the target project.

The prediction of effort using analogies is based on the completed set of projects that are similar to the new one. Li *et al*., (2007) comment that there are many methods for finding the number of analogies, but tools such as ANGEL compute a similarity measure using the project and product features between a new project and projects in the historical database. However, the major drawback with this method is that the tool will provide estimation even if the dataset is absolutely irrelevant for case-based estimation.

Analogy-X: To overcome this drawback, a new research has identified an approach called Analogy-X that uses Mantel's correlation and randomization tests to verify the basic hypothesis of finding the statistical basis for analogy. Initially, a similarity matrix is constructed for effort as well as for project factors based on which the Mantel's correlation is calculated. The correlation value is then used to decide the relevance of the datasets. The dataset whose correlation value is significantly different from zero is found to be appropriate.

Mantel's Randomization test is subsequently used to test whether the value is significantly different from zero. A stepwise feature selection and sensitivity analysis, based on Jack-Knife method is used to provide the confidence limits on values of R_M that are significantly different from zero.

In effect, the benefits of using the proposed algorithm include:

- An algorithmic basis for estimation of analogy
- Ability to accurately discriminate the significant and insignificant predictive relationships
- Detection of aberrant data point using sensitivity analysis

Proposed algorithmic method: In order to determine the software cost, although several statistical packages are available, the lack of complete tool for estimating the project cost by the concept of analogy for handling the categorical dataset has inspired us to develop an enhanced algorithm that serves as the base for the development of a software tool for cost estimation.

Our study consists of two enhanced algorithms which are depicted in Fig.2. Algorithm 1 is used to find out whether the analogy based estimation is an appropriate method for the dataset. Algorithm 2 is proposed to detect the aberrant data points in the historical datasets by taking all possible combinations of project adaptive features

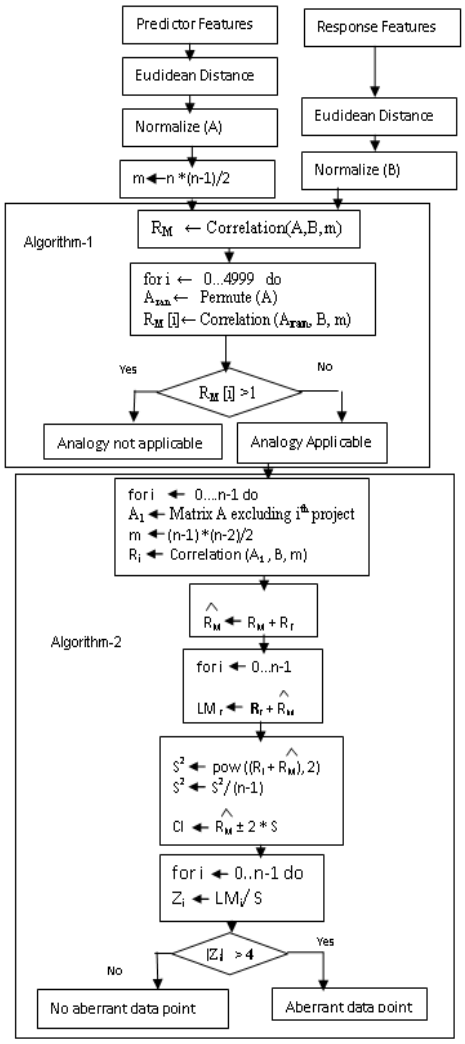

Fig. 2: Proposed algorithmic flow chart

Algorithm 1 considers predictor distance matrix and the response distance matrix as input parameters. To test a hypothesis, there should be a relationship between these two matrices and Algorithm 1 identifies the association between the corresponding elements in two distance matrices. The significance of the correlation is determined by the permutation procedure in which the original value of the test is compared with the distribution of elements found by randomly reallocating the order of the elements in one of the distance matrices .

However, it is highly impractical to consider all the possible permutations of distance matrix elements for the number of cases that is invariably large. However, it is essential that the significance level generated from permutations has to be close to the level obtained with all the possible permutations. There should be a minimum of 1000 randomizations for estimating a significant level of about 0.05 and at least 5,000 randomizations are realistic minimum for achieving a significance level of about 0.01.

When R_M <1, Algorithm 2 is used to construct the estimator $\hat{R}_M$ and the confidence intervals. A new project feature has been added to obtain an increased $\hat{R}_M$. It is assumed that the new feature has more significant contribution to assess the similarity, if the new value of $\hat{R}_M$ is larger than the upper 95% confidence limit. The abnormal data points are obtained by omitting one project at a time from the dataset and calculating the R_i which is the correlation for the dataset excluding project i. The LM_i which supports the sensitivity analysis for analogy is calculated by finding the difference between the overall $\hat{R}_M$ and R_i indicating the impact of the specific case R_i on the overall R. The z-test provides a mechanism to formally verify whether the value of R_i is an aberrant one.

RESULTS

The datasets used in this study is the Desharnais dataset (Keung *et al*., 2008) and NASA 93 (Li *et al*., 2009b). The dataset has 9 independent variables and 1 dependant variable. Actual Effort in person hours is used as the 10th variable for the matrix B.

The NASA 93 dataset comprises of 93 complete projects, having 17 independent variables of which 15 are categorical variables. This dataset is in COCOMO 81 format collected from NASA centers Published in PRedictOR Models in Software Engineering (PROMISE). In this format, the DevEffort variable is considered for the generation of matrix B.

On applying the Desharnais dataset to the described algorithm, the following set of results shown in Table 1 and 2 were obtained.

Table 1: Results produced using algorithm 1on desharnais

Results of algorithm 1 on desharnais dataset		
R_M	:	0.1345
Value returned	:	true
Hence, analogy is applicable for desharnais dataset		

Table 2: Results produced using algorithm 2 on desharnais

Results of Algorithm 2 on Desharnais Dataset		
$\hat{R}_M$	:	-0.1348000
S^2	:	0.0000590
UCI	:	-.0119463
LCI	:	-0.1501610

Table 3: Results produced using algorithm 1 on NASA 93

Results of algorithm 1 on NASA 93 dataset		
R_M	:	-0.12104
Value returned	:	true
Hence, Analogy is applicable for NASA 93 dataset		

Table 4: Results produced using algorithm 2 on NASA 93

Results of algorithm 2 on NASA 93 dataset		
$\hat{R}_M$	:	-0.120340
S^2	:	0.0000350
UCI	:	-.0108448
LCI	:	-0.132231

The results obtained by applying the algorithms to the NASA 93 dataset are tabulated in Table 3 and 4.

It is found that the Desharnais 77 dataset and NASA 93 are applicable for Software effort estimation by Analogy. The value of z_i is usually taken as an indicator to identify the abnormal cases. If the value of $|z_i|>4$, then it is said to indicate an abnormal data point.

In our experiments, it is found that:

z_{76} = 8.4121 (Desharnais Dataset)

z_{89} = 9.102127 (NASA 93 Dataset)

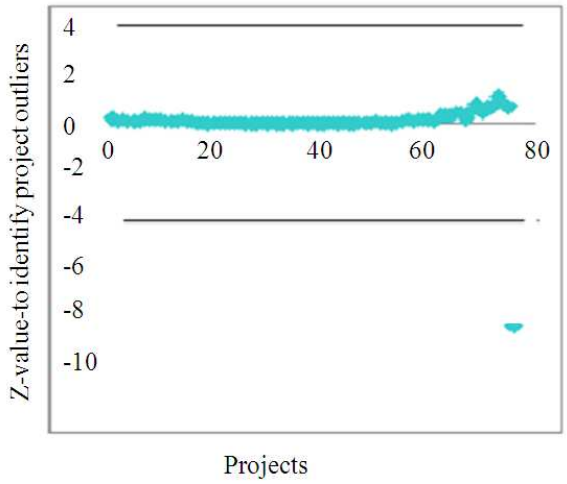

Fig. 3: Results of desharnai's dataset

Figure 3 illustrates the outlier case present in the Desharnai's dataset, while Fig. 4 indicates the same for NASA 93 dataset. From Fig. 3 and 4, it is predicted that if the dataset contains more categorical variables, it is easy to identify the aberrant data points. The theoretical and graphical results simulated shows that the particular case is an aberrant project set and hence can be removed. The process is again repeated excluding the aberrant case for predicting the software cost estimation.

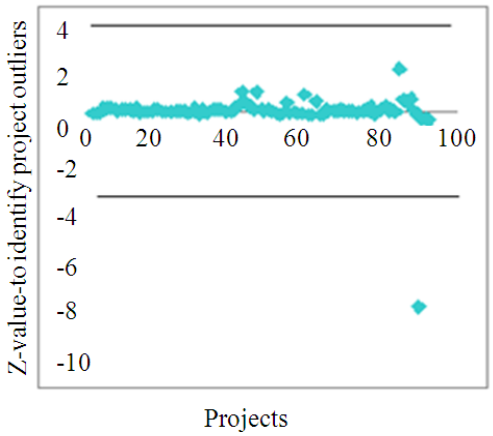

Fig. 4: Results of NASA93 dataset

DISCUSSION

Only statistical evidences are available which do not support the categorical datasets. Therefore, these proposed algorithms serve to support the categorical and numerical datasets. By predicting the relation of two matrices, Algorithm 1 identifies whether analogy is applicable to the dataset or not while Algorithm 2 detects the aberrant data points using the upper and lower confidence values. This enhanced algorithm successfully produces accurate results based on the concept of analogy for an effective estimation of effort.

CONCLUSION

This study is based on the concepts of an algorithmic method that provides an accurate result for delineation of the aberrant cases involved in the software project development. The algorithm:

- Assesses the suitability of the dataset for analogy
- Identifies the most appropriate feature subset
- Eliminates an aberrant project case and
- Determines the most suitable adaptive feature weights for the purpose of software effort estimation

Although the software cost could be estimated using the Analogy methodology, it failed to provide a complete statistical basis for the estimation. This drawback has been overcome by using the extended Analogy method. However, it is difficult to integrate the extended analogy concept with other machine learning techniques and also to design a tool for automation of the process. Therefore, this algorithm serves as a basis for designing a new tool as well as for

integrating the same with other techniques to handle both categorical and numerical datasets.

REFERENCES

Azzeh, M., D. Neagu, P.I. Cowling, 2011. Analogy-based software effort estimation using Fuzzy numbers. J. Syst. Software., 84: 270-284. DOI: 10.1016/j.jss.2010.09.028

Huang, X., D. Ho, J. Ren and L.F. Capretz, 2007. Improving the COCOMO model using a neuro-fuzzy approach. Applied Soft Comput., 7: 29-40. DOI: 10.1016/j.asoc.2005.06.007

Jorgensen, M. and M. Shepperd, 2007. A systematic review of software development cost estimation studies. IEEE Trans. Softw. Eng., 33: 33-53. DOI: 10.1109/TSE.2007.256943

Keung, J., 2008. Empirical evaluation of analogy-x for software cost estimation. Proceedings of the 2nd ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, (ESEM'08), ACM, New York, NY, USA., pp: 294-296. DOI: 10.1145/1414004.1414057

Keung, J.W., B.A. Kitchenham and D.R. Jeffery, 2008. Analogy-X: Providing statistical inference to analogy-based software cost estimation. IEEE Transactions Software Eng., 34: 471-484. DOI: 10.1109/TSE.2008.34

Kocaguneli, E., T. Menzies, A. Bener and J. Keung, 2011. Exploiting the essential assumptions of analogy-based effort estimation. J. IEEE Trans. Software Eng., PP: 1-1. DOI: 10.1109/TSE.2011.27

Le-Do, T.K., K.A. Kyung, Y.S. Seo and D.H. Bae, 2010. Filtering of inconsistent software project data for Analogy-based Effort estimation. Proceedings of the IEEE 34th Annual Computer Software and Applications Conference, July, 19-23, IEEE Xplore Press, Seoul, pp: 503-508. DOI: 10.1109/COMPSAC.2010.56

Li, J., G. Ruhe, A. Al-Emran, M.M. Richter, 2007. A flexible method for software effort estimation by analogy. Empirical Software Eng., 12: 65-106. DOI: 10.1007/s10664-006-7552-4

Li, Y.F., M. Xie and T.N. Goh, 2009a. A study of project selection and feature weighting for analogy based software cost estimation. J. Syst. Soft., 82: 241-252. DOI: 10.1016/j.jss.2008.06.001

Li, Y.F., M. Xie and T.N. Goh, 2009b. A study of the non-linear adjustment for analogy based software cost estimation. Empirical Software Eng., 14: 603-643. DOI: 10.1007/s10664-008-9104-6

Mendes, E., I. Watson, C. Triggs, N. Mosley and S. Counsell, 2003. A comparative study of cost estimation models for web hypermedia applications. Emperical Software Eng., 8: 163-196. DOI: 10.1023/A:1023062629183

Mukhopadhyay, T., S.S.Vicinanza and M.J. Prietula, 1992. Examining the feasibility of a case-based reasoning model for software effort estimation. MIS Q., 6: 155-171.

Omar, M., S.L. Syed-Abdullah and A. Yasin, 2011. The impact of agile approach on software engineering teams. Am. J. Econ. Bus. Admin., 3: 12-17. DOI: 10.3844/ajebasp.2011.12.17