Advances and Applications in Mathematical Sciences
Volume 18, Issue 8, June 2019, Pages 637-649

GA BASED DIMENSIONALITY REDUCTION FOR EFFECTIVE SOFTWARE EFFORT ESTIMATION USING ANN

SOMYA GOYAL and PRADEEP K. BHATIA

Department of Computer Science & Engineering
GJUS&T, Hisar, India
Email: somyagoyal1988@gmail.com
pkbhatia.gju@gmail.com

Abstract

Software effort estimation is highly crucial for successful delivery of the quality product within the budget limits. The accurate prior estimation of effort, to be consumed in the candidate project is essential for timely completion of the project. Machine learning is very effective in the field of predicting the effort for the software. Effort estimation is formulated as a learning problem, a prediction problem with features of dataset as inputs and the effort as target. This work reduces the dimensionality of dataset for effective prediction of software effort. For reducing the dimensions of feature-set, an evolutionary technique named Genetic Algorithm is deployed with performance of Artificial Neural Network as fitness criteria. Multilayer perceptron with back-propagation algorithm is trained simultaneously while Genetic algorithm selects the features. The reduced feature-set then deployed to predict the software effort using public dataset. A comparison is made between the performances of proposed predictor working with few selected features and a predictor built with all the features taken as inputs. It is found that the proposed model with reduced dimensionality of dataset using genetic algorithm is much better than predictor without any feature reduction technique.

I. Introduction

Accuracy in software effort estimation is highly desirable for effective software project management. Only 32% of software projects get successfully completed, as reported by a survey report [1]. It is solely due to the reason that effort required for the project is either overestimated or underestimated. Numerous methods have been proposed for estimating the efforts accurately like Expert judgement, Analogy based and COCOMO based parametric

2010 Mathematics Subject Classification: 92B20.
Keywords: Genetic Algorithms, Neural Network, Effort estimation, MMRE, Feature Selection.
Received February 3, 2019; Accepted March 17, 2019

models [2]. These methods were conventional and in the today's changing environment, these are not so suitable for accurate estimation. During past three decades, machine learning techniques are finding remarkable applications in the field of SEE (Software Effort Estimation). Artificial Neural Network, Decision Tree, Support vector machine, Bayesian Network and nearest neighbor [3] are majorly used in literature for effective SEE. From the studies, it is clear that the accuracy of the predictor depends to a significant extent on the input features of the dataset which are fed as input to the predictor in the data-oriented ML based models. The dimensionality reduction is about selecting the relevant features only and leaving the rest of the not so effective features. A variety of filters and wrappers are used for feature selection [4]. Reducing the dimensionality contributes two-fold: one, the computational time is reduced. And, second, the performance of predictor can be improved after removing irrelevant features of the dataset [5].

In this work, we propose to use meta-heuristic technique based on Genetic Algorithms (GA) for selecting the features of the dataset. The established research questions for our work are-

RQ1- Is GA suitable to reduce the dimensionality of data to improve the accuracy of the predictor?

RQ2- Does the proposed model with reduced dimensionality perform better than the baseline model?

To address the above questions, the present work has been carried out. The structure of the paper is as follows-

Section I introduces the problem statement and brings a light on the current state – of – the – art. Section II describes the proposed model and the work flow of the research carried out in this paper. Section III explains the research methodology including the dataset, the techniques and evaluation criteria used. Section IV presents the experimental results and analysis. The entire work is concluded in section V with remarks on the future scope of the proposed work.

II. Proposed Model

This section presents the model proposed in order to effectively provide the accurate estimation of software effort by using ANN after selecting

relevant features with GA based feature selector shown in figure 1. The proposed model has two components basically: (1) The GA based feature selector which reduces the dimensionality of our dataset. (2) The ANN based predictor which predicts the effort by learning the non-linear relationship between the input features and target effort. This predictor serves as fitness evaluator till the most relevant feature subset is not selected by GA based filter selector. Also, it predicts the effort for the unforeseen project instances after learning from the training dataset with only features selected by the GA feature selector. After the coming up with the most appropriate features, the feature subset obtained is supplied to the ANN predictor for its training and testing phases using the dataset with reduced dimensionality.

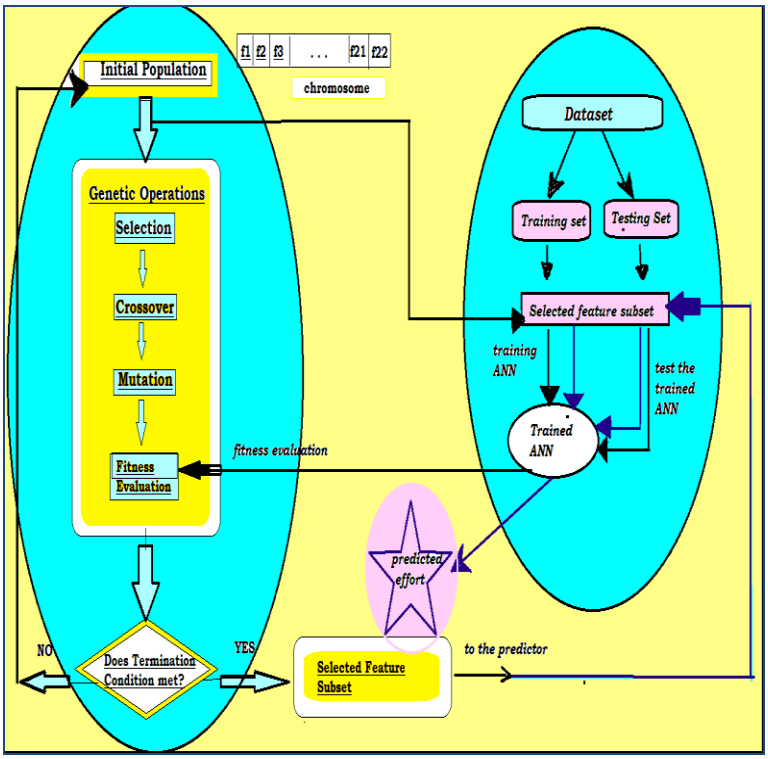

Figure 1. Model proposed for ANN based effort estimation with features selected using GA.

In this work, the performance of this proposed model is compared with the performance of a baseline, the ANN predictor with all features as its inputs. The comparison of proposed model with baseline clarifies the

contribution of the proposed work. The detailed experimental set-up with research methodology is explained in the next section.

III. Research Methodology

The detailed description of the dataset used and the techniques deployed for the presented work is covered under this section.

A. Dataset Used

The Maxwell dataset, which is publicly available has been used in this work. The dataset [7] [8] has 62 project instances with 22 attributes as showcased in figure 2. As it is clear from the figure 2 the dataset has 23 independent attributes and 4 dependent attributes. For this work we are considering only one output variable i.e. Effort. And, all 22 features are being considered for dimensionality reduction to obtain optimal feature subset using GA.

B. Techniques Used

In this work, Genetic Algorithms based technique [9] is used for feature selection to reduce dimensionality. The parameter settings are: population size = 22, number of generations = 10, selection method = Roulette wheel selection, cross-over function = two-point cross Random mutation is applied with the loss function computed from the ANN based predictor as fitness criteria. The population representation is show in figure 3.

Feature	Description	Mean	Std Dev	Min	Max
Time	Time	5.58	2.13	1	9
App	Application type	2.35	0.99	1	5
Har	Hardware platform	2.61	1	1	5
Dba	Database	1.03	0.44	0	4
Ifc	User interface	1.94	0.25	1	2
Source	Where developed	1.87	0.34	1	2
Telon use	Telon use	2.55	1.02	1	4
Nlan	Number of different development languages used	0.24	0.43	0	1
T01	Customer participation	3.05	1	1	5
T02	Development environment adequacy	3.05	0.71	1	5
T03	Staff availability	3.03	0.89	2	5
T04	Standards use	3.19	0.70	2	5
T05	Methods use	3.05	0.71	1	5
T06	Tools use	2.90	0.69	1	4
T07	Software's logical complexity	3.24	0.90	1	5
T08	Requirements volatility	3.81	0.96	2	5
T09	Quality requirements	4.06	0.74	2	5
T10	Efficiency requirements	3.61	0.89	2	5
T11	Installation requirements	3.42	0.98	2	5
T12	Staff analysis skills	3.82	0.69	2	5
T13	Staff application knowledge	3.06	0.96	1	5
T14	Staff tool skills	3.26	1.01	1	5
T15	Staff team skills	3.34	0.75	1	5
Duration	Duration	17.21	10.65	4	54
Size	Application size	673.31	784.08	48	3,643
Effort	Effort	8,223.21	10,499.90	583	63,694

MAXWELL DATASET

Figure 2. Feature Description of Maxwell dataset with central measures.

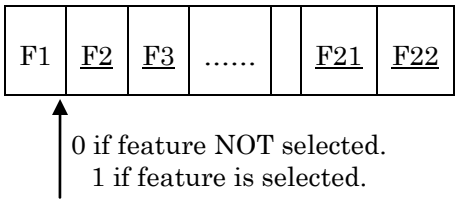

Figure 3. Population representation for dimensionality reduction.

One of the most popular ML technique is used for the prediction of effort namely Artificial Neural Network (ANN) [10]. The parameter setting for the predictor is summed up in Table I.

Table I. Parameter setting for ANN model.

Parameter	Value
Size of Input Layer	Variable
Size of Hidden Layer	10
Size of Hidden Layer	10
Size of Output Layer	01
Learning Algorithm	'trainlm'
Transfer Function	'tansig'

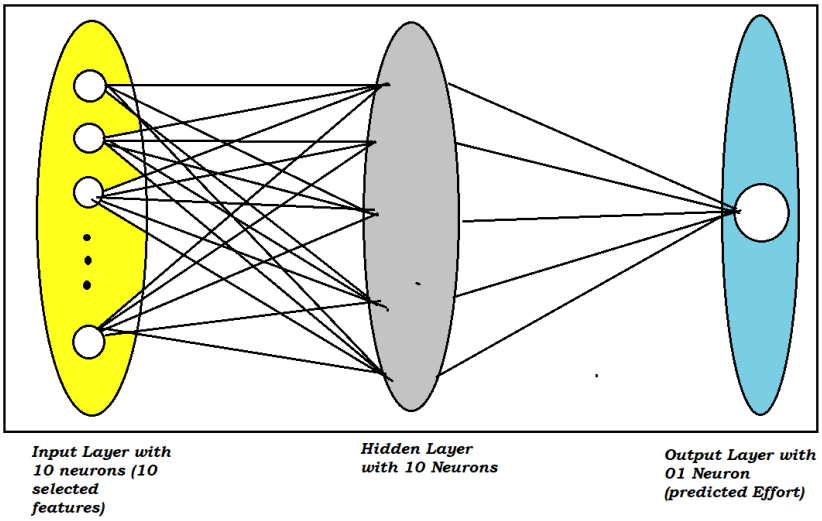

Figure 4. Fully interconnected ANN as predictor.

The ANN predictor shown in Figure 4 is used for two purposes: (1) To provide fitness value with the variable number of features as inputs. (2) To predict the effort using the only reduced feature subset.

C. Evaluation Criteria

The performance is evaluated using MRE and MMRE metrics [11]. The MRE stands for Magnitude of Relative Error and can be computed for a specific project instance using equation (1).

$$\text{MRE} = |\text{Effort(actual)}-\text{Effort(predicted)}| \div \text{Effort(actual)} \quad (1)$$

In equation (1) the Effort(actual) is the effort provided in the dataset for 62 project instances and Effort(predicted) is the effort predicted by the proposed model for that specific project instance.

The MMRE denotes Mean Magnitude of Relative Error and is the averaged value of MRE over the entire dataset and can be computed using equation (2) as follows:

$$MMRE = \Sigma\, MRE \div Num \tag{2}$$

In equation (2) the MRE over all the projects are summed up and Num represents the project count in the dataset which is 62 in this experimental dataset.

The Evaluation criteria MRE is used to assess the performance of prediction model for a specific instance and the MMRE is calculated for the overall evaluation of the prediction power.

IV. Experimental Results and Analysis

This section discusses the experimental results obtained and the analysis made. The contribution of the proposed work is two-fold; (1) To reduce the dimensionality of dataset using GA algorithms; We obtained the feature subset with 10 features out of total 22 features. (2) To predict accurately the Effort of the projects by learning the non-linear relationship between the features and effort. The experiment shows that the predictor works well with the value of 0.48 for MMRE.

The results clearly show that reduced dimensionality improves the accuracy of predictor. The performance of proposed model (GA + ANN with 10 features) is compared with the performance of a baseline model (ANN with all 22 features) depicted in Table 2. The table has 62 entries, one for each project. The unit of performance measurement is taken MRE. The value for MRE is computed by feeding the values from the dataset and the predicted by the models in equation (1). The same process is repeated for both the models and comparison is made between their performances.

Table II. Performance of predictors (in MRE).

Project # instance	***Effort by Proposed Model (in MRE)***	***Effort by Baseline Model (in MRE)***
1	0.17	0.32
2	0.10	1.08
3	3.87	0.09
4	0.00	0.03
5	3.06	0.41
6	0.61	0.23
7	0.00	0.13
8	0.24	0.72
9	1.36	0.02
10	0.02	0.28
11	0.01	0.08
12	0.09	0.32
13	0.04	1.55
14	0.00	0.01
15	2.36	0.20
16	0.03	0.47
17	0.00	0.80
18	0.21	0.01
19	0.00	0.06
20	0.33	0.11
21	0.01	0.02
22	0.01	0.09

23	0.01	0.10
24	0.02	1.89
25	0.01	0.01
26	0.61	0.85
27	0.03	0.12
28	0.00	0.40
29	0.04	0.32
30	0.03	1.01
31	0.73	0.05
32	0.04	0.25
33	0.02	2.74
34	0.02	0.52
35	0.32	0.08
36	0.01	0.35
37	0.02	0.13
38	0.00	0.08
39	0.01	0.00
40	0.00	2.15
41	0.00	0.40
42	1.99	0.25
43	0.05	0.38
44	1.81	0.22
45	0.03	0.15
46	0.01	0.16
47	0.00	3.01

48	0.00	0.85
49	2.72	0.17
50	2.99	0.03
51	0.01	5.95
52	0.00	1.18
53	0.05	0.07
54	0.02	0.03
55	0.01	1.18
56	4.87	0.69
57	0.17	0.92
58	0.78	0.11
59	0.03	0.75
60	0.01	4.69
61	0.00	0.01
62	0.00	0.02

The values obtained from both the models are plotted in a comparison graph to clearly analyze the difference between them as shown in Figure 5. In the plotted graph, the data values for proposed model (GA + ANN with 10 selected attributes) are shown with blue solid squares and resulted in a line graph represented with a blue line connecting the observations. Similarly, the pink solid squares represent the data points for MRE for baseline model (ANN with all 22 attributes). The connecting line shown with pink colour represents the overall performance of baseline model. It is evident from the plotted graph that the performance of proposed model is better from the baseline model.

Further, the proposed model is more accurate than the baseline model with the MMRE of 0.48 whereas the baseline model predicts with MMRE of 0.63.

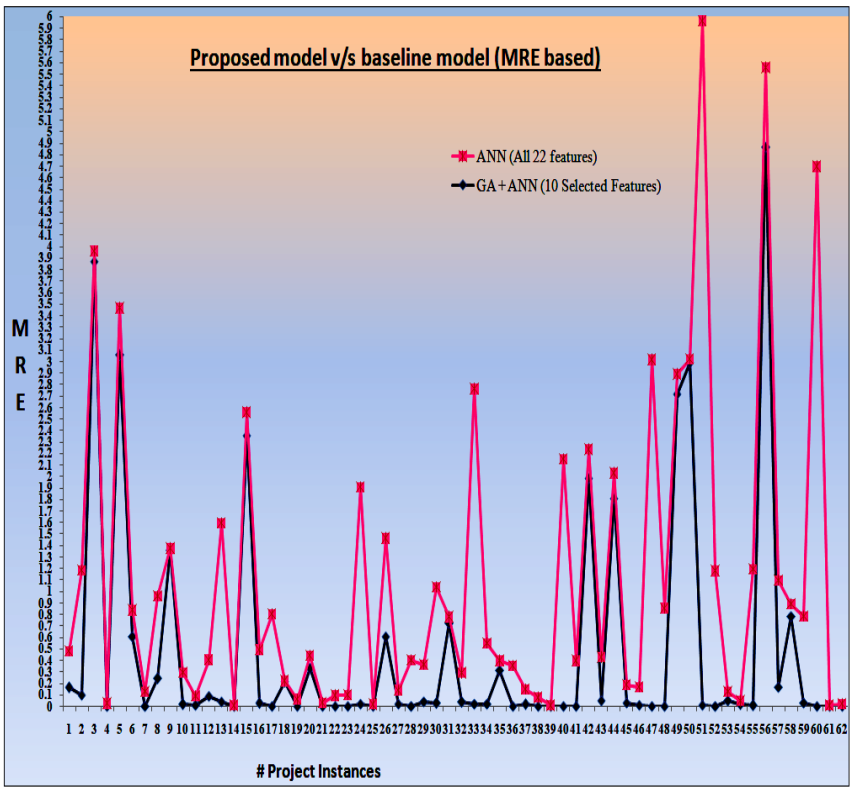

Figure 5. Comparison of Proposed model with Baseline Model (in terms of MRE).

The statistical validation is carried out of the results obtained using Wilcoxon Signed Rank Test [12] at 5% significance level. We assumed H0: there is no statistical difference between the performance of these two models (Proposed model and Baseline Model) and H1: There is statistically difference in the performances of these two models at significance level of 5%. The result of statistical test confirms that there is statistically significant difference between the performances of these two models with p of 0.02 rejecting the null hypothesis.

V. Conclusion

In this work, we investigated the Genetic algorithms based technique to reduce the dimensionality of the dataset which is to be supplied to a ANN

based Software Effort Estimation Model. The results show that the feature selection improves the accuracy of the prediction model. A comparison is also made between the performance of the proposed model (ANN + GA) with reduced dimensionality and the performance of a baseline model (ANN) without any feature selection. The comparative analysis confirms that feature reduction improves the prediction accuracy and Genetic Algorithms are suitable for dimensionality reduction. With the help of statistical test named Wilcoxon rank sign test, validation of the experiment is also done. In future, we propose to extend the work using other meta heuristic techniques for reducing the dimensionality. The experiment can also be replicated with some other project dataset.

References

[1] Chaos Report, 2015. The Standish Group.

[2] Somya Goyal and Anubha Parashar, Machine Learning Application to Improve COCOMO Model using Neural Networks, International Journal of Information Technology and Computer Science (IJITCS), Vol.10, No.3, pp.35-51, 2018. DOI: 10.5815/ijitcs.2018.03.05.

[3] Jianfeng Wen, Shixian Li, Zhiyong Lin, Yong Hu and Changqin Huang, Systematic literature review of machine learning based software development effort estimation models, Information and Software Technology, Vol. 54, 2012, pp. 41-59.

[4] M. Hosni, A. Idri and A. Abran, Investigating heterogeneous ensembles with filter feature selection for software effort estimation, in Proceedings of the 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement on - IWSM Mensura'17, 2017, pp. 207-220.

[5] Juan Murillo-Morera, Carlos Castro-Herrera, Javier Arroyo and Ruben, Fuentes-Fernandez, An Automated Defect Prediction Framework using Genetic Algorithms: A Validation of Empirical Studies, Inteligencia Artificial 19(57) (2016), 114; doi:10.4114/ia.v18i56.1159.

[6] K. Maxwell, Applied statistics for software managers, Englewood Cliffs, NJ, Prentice-Hall. 2002.

[7] http://www.promisedata.org/?p=108.

[8] K. Maxwell, Applied Statistics for Software Managers, Prentice, 2000.

[9] Jianfeng Chen, Vivek Nair and Tim Menzies, Beyond evolutionary algorithms for search-based software engineering, Software Technology (2017). http://dx.doi.org/10.1016/j.infsof.2017.08.007

[10] A. B. Nassif, M. Azzeh, L. F. Capretz and D. Ho, Neural network models for software development effort estimation: a comparative study, Neural Comput. Appl., vol. 27, no. 8, pp. 2369, 2016.

[11] C. López-Martín, Predictive accuracy comparison between neural networks and statistical regression for development effort of software projects. Appl. Soft Comput. 2014. doi:10.1016/j.asoc.2014.10.033.

[12] S. M. Ross, Probability and Statistics For Engineers And Scientists, Third Edition, Elsevier Press, 2005, ISBN: 81-8147-730-8.