Anticipation of Software Development Effort using Artificial Neural Network for NASA Data Sets

Kiran Kumar T.M, Sharada Aihole S, Soundarya Putage S
Assistant Professor, Student
Department of MCA
Siddaganga Institute of Technology, Tumkur, India

Abstract:
Failures of software are mainly due to the faulty project management practices, which include effort estimation. Continuous changing outlines of software development technology make effort estimation more challenging. Several methods are available in order to estimate the effort among which soft computing based method plays a prominent role. Software effort estimation is to find out the development effort required by a project. Before any project is being started initially, it is required to estimate the development effort required. This is one of the most important and challenging activities that has done before. Software cost estimation deals with lot of uncertainty among all soft computing methods neural network is good in handling uncertainty. In this project we have used artificial neural network to predict the effort estimation using the software Mat lab -11. The Dataset used for analysis and validation of the model can be got from historical projects of NASA. Various techniques have been proposed to calculate effort estimation, but providing accurate cost is still an unachievable goal.

Keywords: Software Effort Estimation, Soft computing techniques, Artificial neural network, MMER, MRE

1. INTRODUCTION
An important objective of the software engineer is to develop useful models that are accurately estimating the software effort. Effort estimation is a process of predicting probable cost and development time to develop a software, process or product. Estimating software development effort remains a complex problem, and the one which continues to draw significant research attention. Software project managers usually estimate the software development effort, cost, and duration in the early stages of a software life cycle in order to appropriately plan, monitor, and control the allocated resources, to complete the project in on-time, on-schedule without wasting any resources. Correctness in estimating the required software development effort plays a critical factor in the success of software project. Right amount of resource should be allocated to a project.

For good software estimation tool the estimated effort should be approximately equal to the actual effort. Accurate estimation allows manager to allocate the resource to plan and coordinate all activities. In soft computing approach several technique like neural network fuzzy logic, genetic engineering are used either individually or combined as hybrid approaches to predict the effort. Now a day’s estimation method using neural network is the interesting area for research compared to Theoretical estimation methods.

1.1 Artificial Neural Networks
An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. Neural network simulations appear to be a recent development. However, this field was established before the advent of computers, and has survived at least one major setback and several eras. Many important advances have been boosted by the use of inexpensive computer emulations. Following an initial period of enthusiasm, the field survived a period of frustration and disrepute.

During this period when funding and professional support was minimal, important advances were made by relatively few researchers. These pioneers were able to develop convincing technology which surpassed the limitations identified by Minsky and Papert. Minsky and Papert, published a book (in 1969) in which they summed up a general feeling of frustration (against neural networks) among researchers, and was thus accepted by most without further analysis. Currently, the neural network field enjoys a resurgence of interest and a corresponding increase in funding.

1.2 About Artificial Neural Networks
Artificial Neural Network (ANN) uses machine learning and pattern recognition methodology to find accurate estimates for software development effort. It is found that ANN improves performance of effort estimation on the basis of mean absolute error. ANN (Artificial Neural Network) has the ability to discover relationships between the dependent and independent variables. It can also learn from previous data. ANN is the interconnection of artificial neurons.

ANNs include the two fundamental components of biological neural networks are Neurons (nodes) and Synapses (weights). The main focus of this paper is to discuss how we can predict the effort using Artificial Neural Network techniques and a comparative analysis of ANN techniques for software development effort estimation.
1. Adaptative learning
2. Self-Organization
3. Real Time Operation
4. Fault Tolerance via Redundant Information Coding

2. LITERATURE SURVEY

A lot of literature is available on software effort estimation. Research has been done using machine learning techniques like Artificial Neural Networks, Decision Tree, Linear Regression, Fuzzy Logic, Genetic Algorithm, Empirical Techniques and Theory based techniques for predicting the software effort. Neural networks and statistical regression models have been investigated and results have shown that the performance of both techniques indicate that are competitive with models generated from data of large projects. Many literatures are found based on these techniques but the recent ones are presented here: Jorgensen [1] provides a detailed review of different studies on the software development effort. Neural networks have learning ability and are good at modelling complex nonlinear relationships; provides more flexibility to integrate expert knowledge into the model. Many researchers have applied the neural networks approach to estimate software development effort. Many different models of neural networks have been proposed [2]. They may be grouped in two major categories. First one is feed forward networks where no loops in the network path occur. Another one is feedback networks that have recursive loops. The paper by Srichandan,[3] have discussed that, with advancement in time number of software effort estimation models has been developed, but still there is no such model that can calculate the exact effort required. So, artificial intelligence techniques are being used such as neural network, fuzzy logic, genetic algorithm, case based reasoning etc., Jaswinder Kaur [4] has mentioned that, as the software has started being largely developed for commercial purpose. There is a huge requirement of software effort estimation. Finally, the main result is reliable and precise when applied to software datasets and the result obtained from ANN was more effective as compared to any other models. The paper by B. Tirimula Rao, [5] has focused basically on the FLAN techniques of neural network. It is one of the most important tasks to calculate the software effort. Since, there is no particular formula to calculate effort estimation. FLAN (Functional Link Artificial Neural Network) technique has been used to calculate the effort estimation using the COCOMO datasets. In Ref. [6] the author has explained that one of the biggest challenges for the software industry is to select the best approach to calculate the effort estimation cost of the software. Neural techniques have proved very effective in software effort estimation. The paper by Dr.Finnie, [7] has done a research study which compares three estimation techniques using function points. It was based on artificial neural network, regression analysis and case-based reasoning. The paper by Gerhard witting, [8] has mentioned the most effective requirement in terms of software is to find the best method for software effort estimation and Artificial Neural Network is considered as a good approach because of their capabilities to give good results in terms of large and complex software projects. Feed forward neural network technique has been used for cost estimation [9].

3. EFFORT ESTIMATION PROCEDURE

In this experiment we have used artificial neural network and for effort estimation. MATLAB11 NN tool is used for this experiment. We collect the required data from the source and other characteristics. Then we divide the number of data into two parts – training set & validation set. Maximum of the work in the
application of neural network to effort estimation made use of feed-forward multi-layer, Back-propagation algorithm. Using this data we design the neural network with number of neurons in input layers same as the number of characteristics (attributes) of the project.

The 7 steps for effort estimation using ANN can be outline as follows:

1. Data Collection: Collect data for already developed projects like method used, and other characteristics.

2. Division of data set: Divide the number of data into two factors – training set & validation set.

3. ANN Design: Construct the neural network with number of neurons in input layers like as the number of characteristics of the project.

4. Training: Grain the training set first to train the neural network.

5. Validation: Later training is over then validates the ANN with the validation set data.

6. Testing: Lastly test the created ANN by feeding test dataset.

7. Error calculation: Analysis the performance of the ANN. If satisfactory then stop, else again go to step (3), make some changes to the network parameters and proceed.

<table>
<thead>
<tr>
<th>Table 1. Evaluation Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRE</td>
</tr>
<tr>
<td>0.1171</td>
</tr>
<tr>
<td>0.1171</td>
</tr>
<tr>
<td>0.0307</td>
</tr>
<tr>
<td>0.0355</td>
</tr>
<tr>
<td>0.0247</td>
</tr>
</tbody>
</table>

The accuracy criterion for evaluating models of this study is the Magnitude of Error Relative to the estimate or MER defined as follows:

\[
\text{MER} = \frac{\text{Actual Effort} - \text{Predicted Effort}}{\text{Predicted Effort}}
\]

The MER value is calculated for each observation I whose effort is predicted. The aggregation of MER over multiple observations (N) can be achieved through the mean (MMER) as follows:

\[
\text{MMER} = \frac{1}{N} \sum \text{MER}
\]

The accuracy of an estimation technique is inversely proportional to the MMER.

Fig 3, 4 shows you mean square error dynamics for all our datasets in logarithmic scale. Training MSE is always decreasing, so its validation and test MSE you should be interested in. our plot shows a perfect training.
4. RESULT

Comparison results of BPN for training is given below in Table 1. A model which gives lower MMRE is better than the model which gives higher MMRE. A model which gives higher PRED (25) is better than the model which gives lower PRED (25). Similarly the model which gives lower RMSE is better than the model which gives higher RMSE. The model which mean and standard deviation nearest to Zero is better than the models which gives mean and standard deviation far away from zero.

Table 1. Results of Training for BPN

<table>
<thead>
<tr>
<th>Performance Measures</th>
<th>BPN</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Max</td>
<td>Mean</td>
</tr>
<tr>
<td>MMRE</td>
<td>0.0381</td>
<td>0.2158</td>
<td>0.0659</td>
</tr>
<tr>
<td>PRED(25)</td>
<td>72.712</td>
<td>92.002</td>
<td>85.806</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.3449</td>
<td>2.0085</td>
<td>0.5125</td>
</tr>
<tr>
<td>Std.Dev</td>
<td>0.2367</td>
<td>0.9875</td>
<td>0.0005</td>
</tr>
</tbody>
</table>

5. CONCLUSIONS

Estimation is one of the crucial and challenging tasks in software project management. This project with NASA dataset has been carried out using Matlab version 11 NN tool box for training the data. Different methods of neural network have been used to calculate effort estimation. Each and every technique focuses on providing best software effort estimation.

- Neural network is a good approach in estimating development effort. It was suggested that for complex and computationally large projects it’s better to use neural network approach. But there is a need to examine accuracy of methods which mostly required in software effort estimation.
- It was analyzed that neuron based models have better estimation capability and hence can be used to calculate software effort estimation of all kinds of project.
- The results from our project shows that predicted effort should be as low as possible then the actual effort.

6. REFERENCES


