Effort Estimation using Hybridized Machine Learning Techniques for Evaluating Student's Academic Performance

Mukesh Kumar and A.J. Singh
Department of Computer Science, Himachal Pradesh University, Summer-Hill, Shimla, Himachal Pradesh, India

Key words: Effort estimation, random forest, support vector machine, Naive-Bayes, MATLAB, artificial neural network

Corresponding Author:
Mukesh Kumar
Department of Computer Science, Himachal Pradesh University, Summer-Hill, Shimla, Himachal Pradesh, India

Page No.: 82-87
Volume: 19, Issue 4, 2020
ISSN: 1682-3915
Asian Journal of Information Technology

Abstract: Machine Learning (ML) Technology is an area of Artificial Intelligence (AI) to facilitate computer systems with the capability to involuntarily learn and improve from user experience without being explicitly programmed from outside. It mainly focuses on the expansion of computer programs that can access data from its user to learn for themselves. With this technology, we can predict the performance of the student in academic. Here, we are implemented Naive-Bayes (NB), Support Vector Machine (SVM), Random Forest (RF) and Artificial Neural Network (ANN) algorithms to find the total effort required for analysis. We proposed a Hybridized Support Vector Machine-Neural Network (SVM-NN) Machine Learning Algorithm which requires less effort to accurately analysis the student's academic performance.

INTRODUCTION

Machine learning algorithm is an application area of Artificial Intelligence which further used to develop a system program which automatically learn from the input given by user and gave the result. Nowaday, different companies are using this technology to automate there working.

ML algorithms are further divided into different categories like supervised and unsupervised algorithms. In supervised ML algorithm, the input data given to train the program is already level with some class. On the other hand in unsupervised ML algorithms, the input data given to train the algorithm is not known in advance. Artificial Intelligence (AI) refers to software technologies that make a computer to act and thinks like a human being. Artificial intelligence is an assumption and expansion of computer programming that can execute responsibilities that usually require human intelligence.

Data Mining (DM) is another technology which comes into the above categories but it limits itself only for the analysis purpose and finds the hidden information from the given dataset. It uses the ML algorithms to make an analysis and gave a conclusion. DM application area is also varies from telecommunication, marketing, production, hospitality, medical and education sector.

The study of Data Mining with respect to education application area is recognized as Educational Data Mining (EDM). In EDM, we are going to analysis the student dataset which are further collected from different source and analysis to predict the student result, placement, dropout and student progress in academics. Predicting Student Academic Performance (SAP) is very important for any organisation to be in the competition with other in the same market.

Predicting the number of work units compulsory to perform a particular assignment based on an understanding of analogous projects and other project

features that are supposed to be associated with the effort. The functions of the software application are the input and the effort we want to predict. The processing is used to predict the number of units of work required to perform a particular task, based on knowledge of similar projects and other project features that are believed to be related to the effort. It is important to organize, superiority and success of any software application development. The commonly used efficient categories of effort estimation are expert estimation, algorithmic estimation and machine learning. In this contribution, comparisons of different machine learning algorithms have been performed and which algorithm is more suitable in which situation is discussed. The application and scope of Education data mining are:

- Predict the student's undesirable behaviour
- Analysis the student social network activities
- Predict, whether the student is a dropout or not
- Predict the institution placement for the academic year
- Used to plan and schedule the study activities for the students
- Helping the students to form different students group according to their ability

Literature survey: There are so many effort estimation model are developed in recent years and has been surveyed. The effort done by different researcher has been discussed here: Malhotra and Jain[1] presented a paper titled "Software Effort Prediction using Statistical and Machine Learning Methods". In this study researchers, estimate and compares different machine Learning algorithms like Linear Regression, Artificial Neural Network, Decision Tree, Support Vector Machine and Bagging techniques on software project dataset. In his work they used a dataset which is further taken from 499 different software projects. Initially, the dataset contain 19 features but after pre-processing only 10 features are selected using feature selection algorithms (CFS algorithm). In his result, they found the estimation of decision tree algorithm is too good as compared to any other Machine Learning algorithm taken into consideration.

Bhatnagar and Ghose[2] presented a paper titled "Comparing Soft Computing Techniques for Early Stage Software Development Effort Estimation". In this study researchers, implemented Neural Network (NN) algorithm and FIS approach to estimate the effort. They compared Linear Regression Neural network with fuzzy Logic and found that Fuzzy Logic approach gave better performance as compared to other for effort estimation.

Sadiq *et al*.[3] presented a paper titled "Prediction of Software Project Effort Using Linear Regression Model". In this paper authors, implemented Linear Regression (LR) algorithm for estimating the software project effort. Author further explained the importance of software's function point count before estimating the total effort. In study the value of the MMRE is found to be 0.1356.

Saini and Khalid[4] presented a paper titled "Empirical Evaluation of machine learning techniques for software effort estimation". In this study researchers, they implemented different machine learning approaches (Decision Tree, Multi-layer perceptron, Decision Table, bagging, Radial Bias Networks) to estimate total effort required to develop a software project.

Seref and Barisci[5] presented a paper titled "Software Effort Estimation Using Multilayer Perceptron (MLP) and Adaptive Neuro Fuzzy Inference System (ANFIS)". In this paper authors, implemented Multi-layer perceptron and Adaptive Neuron Fuzzy interference algorithms for estimating effort by taking NASA and Desharnais dataset. They analysed these two data set foe Mean Magnitude relative Error and Percentage Relative Error. After implementation they found that ANFIS gave use better result as compared to MLP.

Boetticher, etc., presented a paper titled "An Assessment of Metric Contribution in the Construction of a Neural Network-Based Effort Estimator". In this paper authors, implemented 33,000 different Neural Network algorithm experiments which is further collected from different corporate domains. The experiments assessed the contribution of different metrics to programming effort. This research produced a cross-validation rate of 73.26%, using pred (30).

Hodgkinson and Garratt [6] presented a paper titled "A Neurofuzzy Cost Estimator". In this study authors, implemented Neural Fuzzy machine learning algorithm to predict the total cost of the project. They compared implemented algorithm with ML techniques like Least-squares multiple linear regression and Neural Network algorithm.

PROPOSED ALGORITHM IMPLEMENTATION AND RESULT DISCUSSION

Evaluating the academic performance of the students is crucial to check for the possibilities of improvement in academics. Here, we proposed a computerized resolution for the performance evaluation of the students using ML algorithms. A threshold-based segmentation is employed to complete the evaluation procedure over MATLAB simulation tool. The performance of machine learning is evaluated by accuracy and mean square error.

There are so, many software tools available for estimating total effort required for project development using Machine Learning techniques. These effort estimation prediction tools are WEKA, MATLAB, Orange, RapidMiner etc. In this study, we will be using

Fig. 1: MATLAB interface while uploading dataset

MATLAB tool for implementation of ML Techniques for predicting total effort required for project development. MATLAB is used to resolve huge amount of problems such as Classification, Clustering and Neural Networks techniques. Figure 1 show the MATLAB interface while uploading the dataset for pre-processing.

Proposed algorithm implementation: The proposed algorithm architecture is the combination of Neural Networks and Support Vector Machine. A hybrid classification mechanism is designed which utilizes both the structure of Neural Network and Support Vector Machine. First of all Neural Network is applied for all non matched Target Labels through Neural Network, Support Vector Machine algorithm is applied. The pseudo code is as follows.

Pseudo code for hybrid algorithm:

1. [r, c] = size (gr1ele); // group 1 elements
2. [r1, c1] = size (gr2ele); // group 2 elements
3. group = []; // Target Set Group
4. cnt = 1;
5. for i = 1:r
6. group (cnt) = 1; // Initialization of Target Label
7. $training_{data\ (i,\ 1)}$ = gr1ele (i, 1); // preparing the training data for group 1
8. $training_{data(I,\ 2)}$ = gr1ele(i,2);// training data for group 2
9. cnt++ // Increment in Counter
10. $foreachigroup_{element}$
11. group(cnt) = 2;
12. $training_{data(cnt,1)}$ = $gr2_{ele(i,1)}$;
13. $training_{data(cnt,\ 2)}$ = $gr2_{ele(i,2)}$;
14. cnt++ // Counter Increment
15. End For
16. net = newff ($training_{data}$, group, 20) // Initializing Neural Network
17. net.trainParam.epochs = 100-1000 // Propagating Iterations
18. net = train(net, $training_{data}$, group); // Training
19. res = sim (net, $training_{data}$); // Simulating
20. diff = res-group;
21. abe = (diff);
22. nonzero = [];
23. nzcount = 1;
24. for zx = 1: numel (abe)
25. if abe (zx)~ = 0
26. nonzero(nzcount) = zx;
27. nzcount = nzcount +1;
28. end
29. end
30. training data svm new = training data (nonzero);
31. group new = group(nonzero);
32. Figure (1)
33. svmstruct = svmtrain (training data svmnew, groupnew', 'showplot', 'true');
34. res = svm class ify (svmstruct, training data svmnew, 'showplot', 'true');
35. group = groupnew';
36. end
37. end

Figure 2 shows the hybrid structure of the proposed work. In this research, a hybrid form of neural with Support Vector Machine algorithm has been used to train the system.

Below is the implementation result of the above to steps which is implemented on MATLAB tool. Student data with different records are taken as input to the algorithm and then Effort estimation is calculated. Below Table 1 given us the result:

Table 1 and Fig. 3 depict the evaluation of effort estimation of Neural Network (NN), Support Vector Machine (SVM), Hybrid (SVM-NN), Naïve-Bayes (NB) and Random Forest(RF) Algorithm. The X-axis in Fig. 3 shows the total number of supplied student's record whereas Y-axis defines the values obtained for each algorithm being considered. The average value of effort estimation by neural network is 39.14, effort

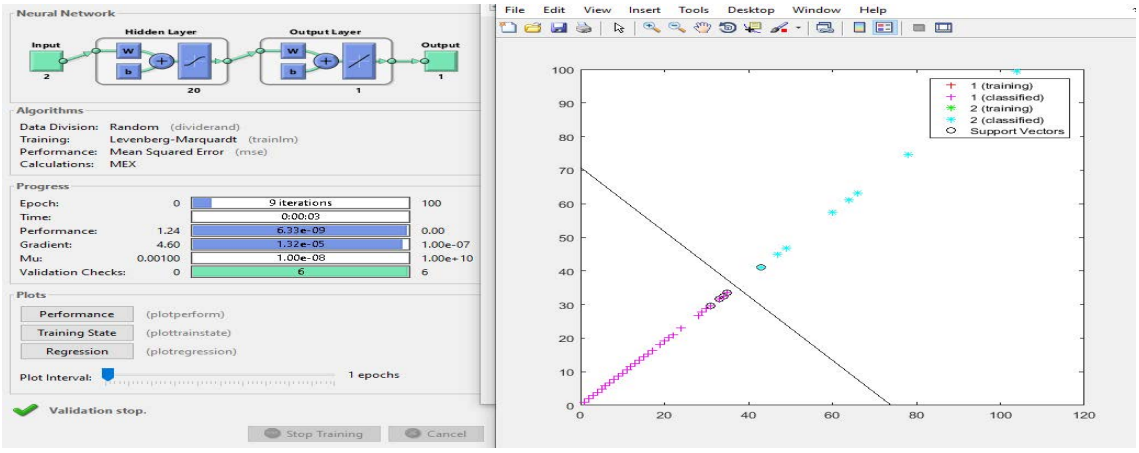

Fig. 2: Implementation of hybrid algorithm using MATLAB

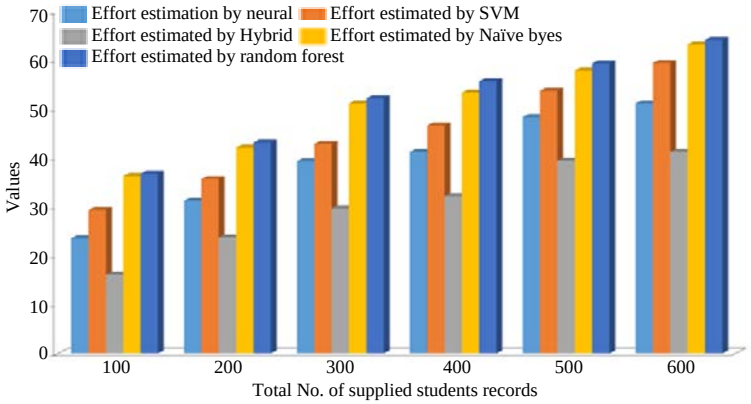

Fig. 3: Effort estimation evaluation different machine learning algorithms

Table 1: Effort estimation by different machine learning algorithms

Student records	ANN	SVM	SVM-NN	NB	RF
100	23.560	29.334	16.110	36.336	36.7854
200	31.250	35.698	23.750	42.125	43.1400
300	39.336	42.856	29.665	51.145	52.2210
400	41.256	46.667	32.145	53.332	55.6900
500	48.339	53.715	39.418	57.896	59.3250
600	51.148	59.413	41.259	63.210	64.1120

Table 2: Estimated effort for neural network and SVM-NN

Neuron count	NN	SVM-NN
10	26.110	19.156
12	25.114	18.114
15	25.100	17.269
20	24.124	17.102
25	24.103	16.936
30	23.221	16.105

Estimated by Support Vector Machine is 44.61, Effort Estimated by Hybrid (SVM-NN) is 30.3, Effort Estimated by Naïve Bayes is 50.67 and Effort Estimated by Random Forest is 51.87.

Figure 4 and Table 2 demonstrates the examination of Neural Network and Hybrid (SVM-NN) as well. The

Table 3: Kernel type linear for support vector machine algorithm

Student records	SVM	SVM-NN
100	29.996	18.6980
200	36.214	21.1120
300	42.265	23.3650
400	47.114	28.1450
500	53.145	29.1450
600	59.362	30.1145

X-axis shows the count of neurons and Y-axis defines the values that are being obtained after the evaluation. It has been seen that the estimated effort in case of Neural Network is more as compared to the Hybrid (SVM-NN) algorithm. The average value of Estimated Effort in case of Neural Network is 24.62 and for Estimated Effort in case of Hybrid (SVM-NN) are 17.44.

Figure 5 and Table 3 demonstrates the examination of Kernel type Linear for Support Vector Machine algorithm. The X-axis in Fig. 5 shows the total number of supplied student's record whereas Y-axis defines the values obtained for Support Vector Machine and Hybrid (SVM-NN) algorithm being considered. It has been seen that the value of Support Vector Machine only is more as

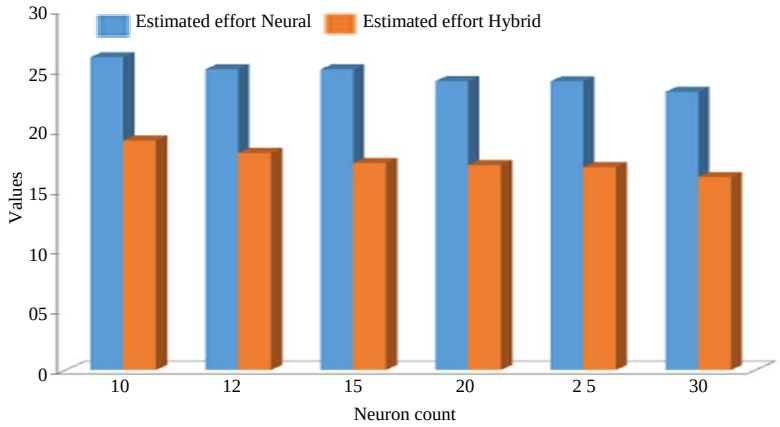

Fig. 4: Estimated effort for neural and hybrid

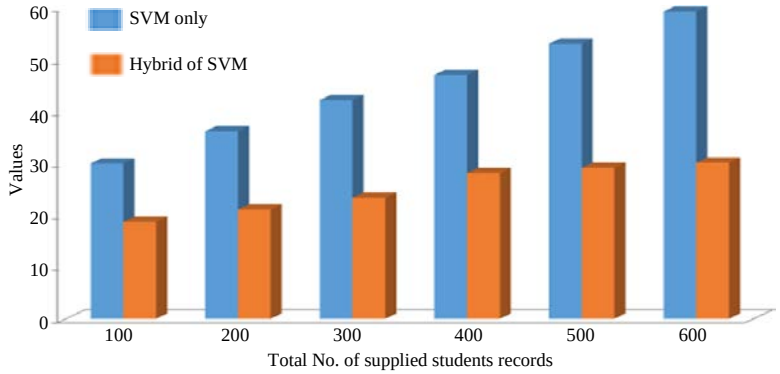

Fig. 5: Kernel type linear for support vector machine algorithm

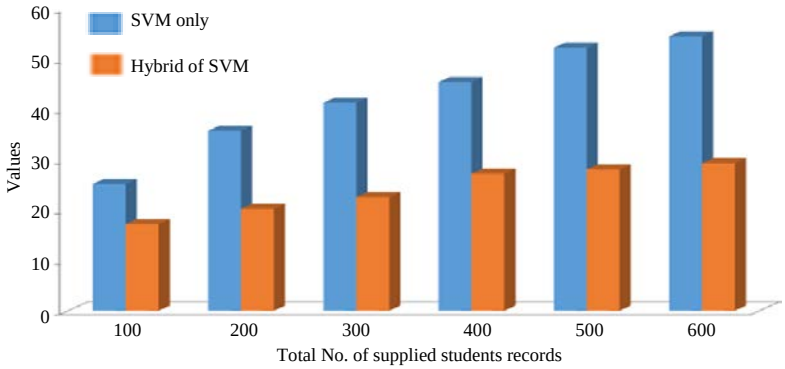

Fig. 6: Kernel type polynomial for support vector machine algorithm

compared to Hybrid (SVM-NN) algorithm. The average value for Support Vector Machine algorithm is 44.68 whereas the value in case of Hybrid (SVM-NN) algorithm is 25.09.

Figure 6 and Table 4 demonstrates the examination of Kernel type Polynomial for Support Vector Machine algorithm. The X-axis in Fig. 6 shows the total number of supplied student's record whereas Y-axis defines the values obtained for Support Vector Machine algorithm and Hybrid (SVM-NN) algorithm being considered. It

Table 4: Kernel type polynomial for support vector machine algorithm

Student records	SVM	SVM-NN
100	25.145	17.1160
200	35.654	20.2038
300	41.256	22.5120
400	45.339	27.1540
500	52.140	28.0010

has been seen that the value of Support Vector Machine algorithm only is more as compared to hybrid (SVM-NN) algorithm. The average value for Support Vector Machine algorithm is 42.31 whereas the value in case of Hybrid (SVM-NN) algorithmis 24.03.

CONCLUSION

The purpose of this research is to find the most effective student features and methods of studying data that help us estimate the student's academic performance. This research helps to locate a variety of data acquisition algorithms for research and effective students. In this study, the machine learning process for evaluation is offered. It is done to help in consuming time and accurate data. The machine will not be content with just evaluating data, making its results better and faster with time and way than traditional processes. Machine learning makes the assessment process better and faster but also allows you to get comments from the analysis. SVM, SVM, RandomForest and Naive Bayes algorithm are used for the classification purpose. The evaluation has been done on the basis of effort estimation. The average value of Effort Estimation by Neural is 39.14, Effort Estimated by SVM is 44.61, Effort Estimated by Hybrid is 30.3, Effort Estimated by Naïve Bayes is 50.67 and Effort Estimated by Random Forest is 51.87. The average value for SVM is 44.68 whereas the value in case of Hybrid SVM is 25.09 in case of Kernel type Linear for SVM. The average value for SVM is 42.31 whereas the value in case of Hybrid SVM is 24.03 in case of Kernel type polynomial for SVM.

RECOMMENDATIONS

The future scope of this research work lies in applying subspace clustering techniques to high dimensional student's academic performance datasets. Normally, student's academic performance data are divided into two parts such as sparse data and dense data. In the proposed technique, the sparse student's academic performance data are not supported and there is a need to improve the efficiency of the system in the future.

ACKNOWLEDGEMENT

I am grateful to my guide A.J. Singh for all help and valuable suggestion provided by them during the study.

REFERENCES

01. Malhotra, R. and A. Jain, 2011. Software effort prediction using statistical and machine learning methods. Int. J. Adv. Comput. Sci. Appl., 2: 145-152.
02. Bhatnagar, R. and M.K. Ghose, 2012. Comparing soft computing techniques for early stage software development effort estimations. Int. J. Software Eng. Appl. (IJSEA.), 3: 119-127.
03. Sadiq, M., A. Ali, S.U. Ullah, S. Khan and Q. Alam, 2013. Prediction of software project effort using linear regression model. Int. J. Inf. Electr. Eng., 3: 262-265.
04. Saini, N. and B. Khalid, 2014. Empirical evaluation of machine learning techniques for software effort estimation. IOSR. J. Comput. Eng. (IOSR-JCE.), 16: 34-38.
05. Seref, B. and N. Barisci, 2014. Software effort estimation using multilayer perceptron and adaptive neuro fuzzy inference system. Int. J. Innovation Manage. Technol., 5: 374-377.
06. Hodgkinson, A.C. and P.W. Garratt, 1999. A neurofuzzy cost estimator. Proceedings of the 3rd International Conference on Software Engineering and Applications (SAE), September 30, 1999, Queen's University Belfast, Belfast, Northern Ireland, pp: 401-406.