Student Performance Prediction using Multi-Layers Artificial Neural Networks: A Case Study on Educational Data Mining

Saud Altaf
Pir Mehr Ali Shah Arid Agriculture University
Rawalpindi, Pakistan
saud@uaar.edu.pk

Waseem Soomro
Manukau Institute of Technology, New Zealand
mwaseem@manukau.ac.nz

Mohd Izani Mohamed Rawi
Universiti Teknologi MARA, Malaysia
izanirawi@salam.uitm.edu.my

ABSTRACT

In recent years, Neural Network (NN) has seen widespread and successful implementations in a wide range of data mining applications, often surpassing other classifiers. This study plans to research of NN that are a fitting classifier to foresee understudy execution from Learning Management System information with regards to Educational Data Mining. The dataset utilized for this examination is a Moodle log document containing log data around 900 understudies more than 10 college classes. To assess the applicability of Neural Networks, two case studies compare their predictive performance on this dataset. The features used for training originate from LMS data obtained during the length of each course, and range from usage data like time spent on each course page, to grades obtained for course assignments and quizzes. After training, the Neural Network outperforms all six classifiers in terms of accuracy and is on par with the best classifiers in terms of recall. We also assessed the effect course predictors have on predictive performance by leaving out the course identifiers in the data. This does not affect predictive performance of the classifiers. Furthermore, the Neural Network is trained on individual course data to assess difference in classification performance between courses. The results show that half of these course classifiers better generally trained classifiers. The importance of individual predictors used for classification was also investigated, with previously obtained grades contributing most to successful predictions. We can conclude that the proposed neural network architecture works well with the selecting the feature data sets. It seems to the results, accuracy in student performance prediction in feature vector has been achieved and satisfactory through appropriate classification to take better decision for efficient prediction of student performance.

CCS Concepts

• **Information systems→ Information systems applications • Data mining→ Data Stream Mining.**

Keywords

Student Performance Prediction, Educational data mining, Neural Network, Classification, training, data sets.

ICISDM 2019, April 6–8, 2019, Houston, TX, USA

ACM ISBN 978-1-4503-6635-9/19/04…$15.00

https://doi.org/10.1145/3325917.3325919

1. INTRODUCTION

In recent years, the use of internet-based educational tools has grown rapidly [1] as well as the research surrounding them (see Figure 1). These tools provide a clear advantage for students and teachers alike, with the ability to access and share course data from anywhere in the world, track student progress and provide rich educational content. These tools generate vast amounts of data obtained in a non-obtrusive manner that can give a better look into the way students learn and interact with course materials. The challenge is to put these data to good use to improve on the educational process. One of the purposes these data can be used for is the prediction of whether a student is going to pass or fail a course. Being able to predict student performance enables a teacher or educational institution to provide appropriate assistance to students that are at risk to miss the mark. Assisting them in a timely manner will reduce the number of students failing a course and may indirectly reduce the amount of students dropping out of their educational program.

This is a societal interest that can have a positive impact on students, parents, teachers and educational institutions equally. When the data comes from an educational setting we are dealing with a sub domain of data mining called Educational Data Mining, or EDM. This is a field of research that applies data mining, statistics and machine learning to data derived from educational environments. It seeks to extract meaningful information from vast amounts of raw data that can be used to improve and understand learning processes [2]. In order to extract interesting information, like predicting if a student requires academic assistance, we can make use of machine learning algorithms that can automatically predict this outcome based on the data. In the field of EDM, a wide set of machine learning algorithms have already been used to various degrees of success like Naive Bayes Classifiers, k-Nearest Neighbours, Random Forests, Decision Tree Classifiers, Support Vector Machine algorithms and Neural Networks [3]-[6].

Most EDM studies investigating student performance prediction have used small samples with little diversity in the courses they analysed [4]. This results in potentially low portability of the results due to the small sample size and differences that might exist between courses; a liberal arts course requires a different approach from a technical course. Furthermore, most studies use a wide variety of grades obtained in previous courses or previous academic curricula [5], where these previous grades have been shown to be strong predictors of future academic success [5-[6]. But these grades might not always be available to use as predictors, thus limiting the predictive capacity of the algorithms devised in these studies.

The goal of the classifiers will be to predict if a student will require academic assistance, because he is at risk to fail the course, or does not require any assistance. As such, it can be cast as a binary classification problem, where the two prediction labels are "requires assistance" for students that are at risk to fail the course, and "does not require assistance" for students that are not at risk. The data used to perform this classification is extracted from the log file of a Campus Management System (CMS) containing information about 900 students over 10 courses. This allows us to compare the predictive performance between courses and assess if predictors identifying individual courses have an effect on performance. Additionally, the effects of sample size and the importance of individual predictors will be investigated.

2. EDUCATIONAL DATA MINING

Research performed in this study can be classified under EDM. EDM is a sub-group of data mining that focuses on researching, developing and applying various automated methods to explore large-scale data coming from educational settings. This is done to increase the understanding of the way students learn, study educational questions and improve the effectiveness of teaching and learning activities [7]. This goal is achieved by transforming the raw data into information that can have a direct impact on educational practice and research [8].

EDM is becoming increasingly widespread nowadays. The past couple of years has seen a rapid rise of the number of research papers dedicated to EDM in its various forms [9]. This has been linked to the increase of available educational data and the widespread availability of cheap computing power and accessible digital tools [10]. With such a wide availability of high-quality data and the potential to derive valuable educational insights, educational institutions, governments and researcher are increasingly looking for ways to put these techniques to good use.

The data analyzed in EDM come from various sources like Learning Management Systems, administrative data from universities and schools and other structured or unstructured databases pertaining to education. Due to the habitually large size of these databases, they require a computerized approach to discern the patterns and relationships they contain [10]. In this study, the data contains interaction records for 900 students, making it too voluminous to derive useful insights by hand or through non-automated means. In this case, an EDM approach is recommendable to extract the information it contains.

In this study, the insights that are extracted from the data concern the prediction of student performance, which is a subdomain of EDM. This can be used to prevent students from failing courses by intervening in their educational process, predict a student's potential to plan an optimal curriculum, give students insight in their learning process or develop more effective instruction techniques [6]. The focus of this study is the applicability of specific machine learning methods in the forecast of whether a student does or does not require academic assistance for a certain course. Such knowledge can help to prevent students from dropping out of their courses or educational program [8].

3. PROPOSED MULTILAYER FEED FORWARD NEURAL NETWORK ARCHITECTURE

Multi-Layer Feed Forward Neural Network (MLFFNN) is the easiest type of ANN in which information travels in unidirectional i.e. from input to output. MLFFNN permits the creation of decision boundary that formed by different hyper planes (n-dimensional space). Multilayer Feed Forward Neural Network is often called Multi-Layer Perceptions (MLP) because of their similarity to the Human perception [3]. MLP have minimum three layers of neurons, input, hidden and output respectively. This means, it is only interconnects within the network and not connected with the surroundings. Mostly, only hide layer is used for the perception. In certain cases, an MLP can have more than one hidden layer where the inputs units are linear. But, it has been proven that one hidden layer is sufficient to estimate any continuous non-linear function provided that sufficient number of input units has been inserted into network [4]. The general structure of a fully connected MLP with input nodes, hidden neuron and output neurons is displayed in following Figure (1).

Where, ω_{ji} represent the connection between i^{th} input layer neuron and j^{th} hidden layer neuron. Similarly, ω_{kj} symbolize the connection between j^{th} hidden layer and k^{th} output layer neuron and $S^{(v)}$ is the signal vector.

A feed-forward network with X_i input and Y_k output signal is shown in Figure 4.2. The computational procedure within i^{th} layer can be illustrated by the following Equation (1).

$$S^{(v)} = f^{(i)}\left[\omega^{(i)} g^{(i-1)}\right] \quad (1)$$

$$g^{(i-1)} = \begin{cases} A \\ \begin{bmatrix} 1 \\ S^{v-1} \end{bmatrix} \end{cases} \quad (2)$$

Where, $S^{(v)} = [\ S_1^{(v)}\ \ S_2^{(v)}\ \ S_3^{(v)}\ \ \ldots.\ \ S_{N_i}^{(v)}\ ]^{\mathrm{T}}$ is the signal vector at the output of the i^{th} layer; $f^{(i)}$ is the activation function of the neurons in the i^{th} layer; g is the bipolar sigmoid function of i^{th} layer; A is the vector containing the input signal for $i = 1.$

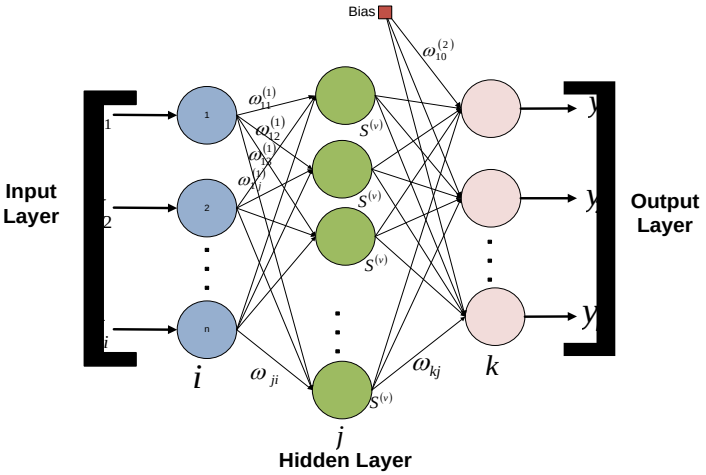


Figure 1: Proposed Schematic Architecture for the MLFFNN

The input array vector feature elements are inserted into network through the weight matrix ω between $(i-1)^{th}$ and i^{th} layer as follows:

$$\omega^{(i)} = \begin{bmatrix} \omega_{11}^{(i)} & \omega_{12}^{(i)} & \omega_{13}^{(i)} & \dots & \omega_{1X_i-1}^{(i)} \\ \omega_{21}^{(i)} & \omega_{22}^{(i)} & \omega_{23}^{(i)} & \dots & \omega_{2X_i-1}^{(i)} \\ \omega_{31}^{(i)} & \omega_{32}^{(i)} & \omega_{33}^{(i)} & \dots & \omega_{3X_i-1}^{(i)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \omega_{X_i1}^{(i)} & \omega_{X_i3}^{(i)} & \omega_{X_i3}^{(i)} & \dots & \omega_{X_iX_i-1}^{(i)} \end{bmatrix} \quad (3)$$

All the neurons in a certain layer are supposed to be similar in all features and the number of hidden layers can be dynamically adjusted according the inputs. So, the network output of the processed information in neural network is presented by the following output array vector.

$$y = S^{(r)} = [\ y_1\ \ y_2\ \ y_3\ \ \dots\ \ y_k\ ]^T \quad (4)$$

Where, r is the number of processing hidden layer toward output layer.

Mostly, MLP are support Back Propagation Neural Network (BPNN) algorithm with supervised training. It is primitively known as the generalized delta rule [6]. The training of a MLP network becomes more complex due to the effect of training algorithm and weight adjustment between the input, hidden and output layer. BPNN can be divided into two levels. Firstly, the comparable training of Perception and Adeline for weight calculation between the hidden and output layer is calculated. Secondly, if no desired output is available from the hidden layer, the errors from the output layer is back propagated and try different architecture of weights between the input and hidden layer.

4. DATA SETS AND PREDICTOR PREPARATION

The data used in this study was a log file of their Campus Management System (CMS) that containing every single user action logged by the system. It spans the academic year 2016-2017 and contains log information about 10 courses with a total of 900 students. The large sample size heightens the probability of the data being diverse and representative of a wide variety of students. We will use this large dataset to assess the importance of sample size on the classification performance of the Neural Network. Furthermore, the availability of the 10 courses allows us to compare the effect courses have on classification performance which can give an insight if the model could be applicable to other unseen courses.

Once the predictors were extracted, the data needed to be normalized. Normalization is necessary for some machine learning algorithms to work properly, like the k-Nearest Neighbour that depends on separate measurements for its goal work. In the event that a component has a scope of qualities (fluctuation) that outperforms that of different highlights, it may command the target capacity of the classifier and make it troublesome for different highlights with littler change to impact the learning procedure. Total 10 possible predictors are used and its decretive statistics analysis values are as follows:

1. CourseID
2. Total learning sessions
3. Total length of session
4. Average of all session length
5. Total assessments in one semester
6. Mean assessment grade
7. Number of quizzes made
8. Total number of emails sent
9. Number of CMS forum posts
10. Grade

The normalization was performed using Scikit-Learn Standard Scalar function. This scalar works by subtracting the column mean and dividing by the column standard deviation for each column. This results in a mean of 0 and variance of 1 for each feature. All predictors are used to forecast whether a apprentice will overtake or be unsuccessful the course as follows.

5. RESULTS

As a case study, in this experiment we want to measure the effect of knowing what course an instance belongs to has on the classification performance. In order to get a better insight in the importance of each predictor, feature importance statistics were extracted from the data including CourseID using the Random Forest classifier. The results can be found in Figure (2). These measures can give an insight in how informative certain features are for classification.

Table 1: Descriptive statistics of used predictors

Predictors	Existing rate in CMS Data Set	Standard Deviation	Mean
CourseID	10	-	-
Total learning sessions	900	660.8	661.2
Total length of session	900	30.7	28.6
Average of all session length	900	20.8	22.6
Total assessments in one semester	450	5.0	4.7
Mean assessment grade	430	6.9	6.8
Number of quizzes made	400	6.0	6.3
Total number of emails sent	280	1.9	2.3
Number of CMS forum posts	220	2.6	2.3
Grade	900	5.6	5.3

The feature with the highest importance is regularity, a measure of how regularly a student accessed the course page, with 12.9%. CourseID obtained a lower importance score at 5.3%, ranking ninth out of fourteen features while Mean Quiz Grade was ranked eight with 8.7%. Predictors like assignment grade, number of assignments made, number of messages sent and number of CMS forum posts all had feature importance below 1.0% and thus have low predictive value. The low importance of assignment grade, while still being a previously obtained grade, could be attributed to the fact that it is only available as a predictor for 2 out of the 10 courses.

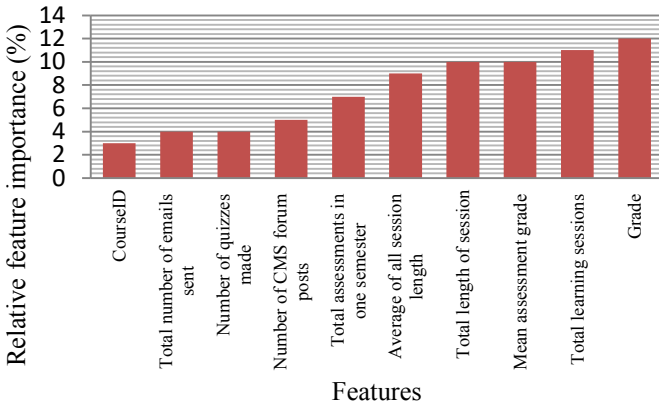


Figure 2: Random Forest relative feature importance.

Subsequently, Neural Network is trained on the data of each course individually and its classification accuracy and recall are measured for each course. This in order to determine if there is an advantage to training the network on all the data or on data for each individual course as well as examining the differences in predictive performance between courses. The results of this experiment can be found in Table (4).

The next step is to identify the uncertainty in data sets in different time frames from predictors using multi-layer FFNN. It can be seen in Table (3) that architecture [4x12x3] has presented the good Mean Squared Error (MSE) performance in classification among all tested architectures. The reasonable numbers of epochs were used during the training process in suitable processing time to attain the necessitated precision, which demonstrates the good efficiency in the all tested architectures, along with the smallest number of error percentage.

Table 4: Performance of Neural Network per course

Course Code	Course name	Credit hours	No. of Students	Accuracy	Baseline Accuracy	Recall	Quiz Grades	Assessments Grades
CS-301	Introduction to Computing	3 (2-2)	280	74.3	69.7	68.9	Yes	Yes
CS-323	Programming Fundamentals	4 (3-2)	270	75.8	69.6	72.7	No	Yes
SSH-303	Professional Ethics	3 (3-0)	110	97.1	96.2	97.6	Yes	Yes
ENG-325	Communication Skills	3 (3-0)	130	80.0	78.5	80.9	Yes	Yes
MTH-310	Multivariable Calculus	3 (3-0)	110	90.0	88.5	89.2	Yes	No
CS-572	Numerical Analysis	3 (2-2)	90	65.2	62.7	64.7	Yes	Yes
CS-632	Artificial Intelligence	3 (2-2)	70	95.0	94.9	95.9	Yes	Yes
CS-666	Web Engineering	3 (2-2)	87	92.1	90.2	92.0	No	Yes
CS-682	System Programming	3 (2-2)	75	83.1	79.4	80.2	Yes	Yes
CS-692	Visual Programming	3 (2-2)	90	90.0	87.9	88.0	Yes	No

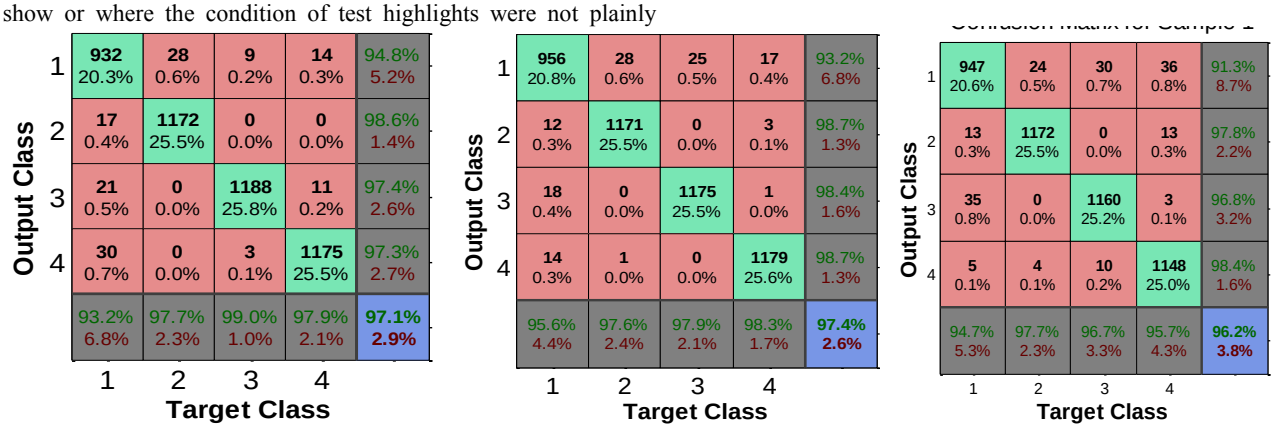
Table 3: Performance of different architectures for classification

Architecture	MSE	No. of Epoch	Accuracy	Error
[4x8x3]	$5.69x10^{-3}$	69	91.5	8.5
	$6.29x10^{-3}$	72	94.4	5.6
	$7.67x10^{-3}$	85	91.6	8.4
	$8.02x10^{-3}$	99	92	8
[4x12x3]	$9.49x10^{-3}$	117	96.2	3.8
	$8.99x10^{-3}$	125	96.3	3.7
	$9.02x10^{-3}$	132	97.4	2.6
	$9.79x10^{-3}$	131	97.1	2.9
[4x15x3]	$8.11x10^{-3}$	369	92.1	7.9
	$5.23x10^{-3}$	325	91.5	8.5
	$5.85x10^{-3}$	344	81.1	18.9
	$6.56x10^{-3}$	362	85.8	14.2

After measuring the neural system testing execution of test information, the following stage was to gauge the grouping perplexity grids for the different sorts of error that happened along with the preparation procedure and reduce them. To construct the

confusion network, test highlight information is given into the neural system display (see Figure 3). For measure the uncertainty level and accuracy, we tested the three architectures of available data sets and train it to achieve the reasonable accuracy rate. In display, the confusion grid is holding the data about the examination among anticipated and focused on grouping classes. Figure (3) demonstrates the confusion networks for the three procedure phase of preparing, testing and approval of data sets maturing process tests separately. Four other anticipated classes (level and vertical) were depicted to reflect about all sample highlight sequence collections. On account of abundant arrangement of a focused on class trial, the targeted cells are appeared in green. Every corner to corner cell demonstrates the quantity of cases that have been arranged effectively by the neural system, to distinguish highlight condition, regardless of whether sound or maturing of data. The cells in red shading call attention to the quantity of cases that have been wrongly ordered by the ANN show or where the condition of test highlights were not plainly recognized. The blue cell shows the general rate of tried cases that were characterized accurately in green and the other way around in red. If there should be an occurrence of test 1, Figure (3) demonstrates each class had most extreme of 1300 testing trials that are as of now foreordained in show. Keeping in mind the end goal to perusing vertically, 947 trials were effectively named class 1. A sum of 13 trials were wrongly named class 2, and 35 trials were wrongly grouped in class 3.

In class 4, an aggregate of 5 trials were just mistakenly grouped because of complex nature of flag and blending of various elements in informational indexes. At the point when the perplexity framework is perused on a level plane, 24 trials of class 2, 30 trials of class 3 and 9 trials of class 4 were erroneously characterized in display. At long last, last column (in dim shading) demonstrates the effective characterization rate of each objective class. The sums of 3952 testing trials were characterized and the last execution rate of accomplishment was 97.4 percent.



(a) (b) (c)

Figure 3: Confusion matrices (a) [4x8x3] (b) [4x12x3] (c) [4x15x3]

Just a 2.6 percent error ratio happened, which is a significant productive and sensible rate in natural prediction finding process. In the confusion networks of different examples, the achievement rate is additionally satisfactory and demonstrates the better execution in recommended NN building model. From Figure (3), it can be observed that the selected neural network matrix [4x12x3] expressed an acceptable and exceptional precision was attained in analysis of ripening process in the characteristic vector, ranging from 96 to 98 percent. This indicates the efficiency in ANN algorithm performance to decrease the altitude of imprecision in assessment creation and strength of sample trained data.

6. DISCUSSION

The goal of this study was to assess to what extent Neural Networks can be used to predict student performance: assessing whether they did or did not need academic assistance, based on the CMS data. Everything considered in analysis, when looking at accuracy, the Neural Network outperforms in this study, which agrees with results by [6]. In terms of recall, it is on par with the best performing classifiers tested here. Considering performance indicators we can say that the Neural Network is an excellent classifier to predict student performance, and can thus be used to predict whether a student requires academic help.

7. CONCLUSION AND FUTURE DIRECTION

The goal of this research was to assess to what extent Neural Networks can be used to predict student performance based on CMS data. We demonstrated that predictive performance of the Neural Network on all targeted courses at once exceeded in terms of accuracy. Leaving out the course predictor did not have a major impact on this performance. However, the Neural Network on each course individually, which resulted in an increase in performance for some course classifiers and a decrease for others architectures compared to the performance of the Neural Network in discussed case study. The effect of sample size was investigated, but no relation between sample size and non satisfactory accuracy was found. Additionally, the feature importance analysis showed that previously obtained grades were the most valuable predictors for individually trained classifiers. For classification and training purposes, a supervised ANN architecture is presented to show the efficiency of data prediction. The simulated results showed that the precise and generality behaviour of training process of different courses and predictors. To improve the Mean Squared Error rate, three type of ANN architecture are tested and [4x10x3] demonstrated the reasonable quantity of hidden layer with high accuracy rate in classification of features vector.

Future development of this research would be extend toward the utilization of complex data sets based of multiple departments and increase the number of targeted student's data through CMS database. A comparison of different faculties would be an interested area with another artificial intelligence technique for better precisely predicts when different parameters measurement

can be taken in parallel of complex datasets to create the complexity.

8. References and Citations

[1]. T. Devasia, Vinushree T P and V. Hegde, "Prediction of students performance using Educational Data Mining," 2016 International Conference on Data Mining and Advanced Computing (SAPIENCE), Ernakulam, pp. 91-95, 2016.

[2]. Asif, R., Merceron, Syed Abbas Ali, Najmi Ghani Haider. Analyzing undergraduate students' performance using educational data mining. Computer & Education vol. 113, 177-194, 2017.

[3]. O. Edin, S. Mirza, "Data Mining Approach For Predicting Student Performance" Economic Review – Journal of Economics and Business, Vol. X, Issue 1, May 2012

[4]. P. Krina, V. Dineshkuma, S. Priyanka, "Performance prediction of students using distributed data mining.", International Journal of Advanced Research in Computer and Communication Engineering", Vol. 6, Issue 3, March 2017.

[5]. Yukselturk, Erman, Serhat Ozekes and Yalın Kılıç Türel. "Predicting Dropout Student: An Application of Data Mining Methods in an Online Education Program" European Journal of Open, Distance and E-Learning, 17.1, 118-133, 2014.

[6]. Tran, Thi-Oanh, Hai-Trieu Dang, Viet-Thuong Dinh, , "Performance Prediction for Students: A Multi-Strategy Approach" Cybernetics and Information Technologies, 17.2, 164-182, 2017.

[7]. M. Goga, S. Kuyoro, N. Goga, "A Recommender for improving the student academic performance", Procedia - Social and Behavioral Sciences, vol. 180, pp. 1481–1488, May 2015.

[8]. T. Mishra, D. Kumar & D.S.Gupta, "Mining Students' Data for Performance Prediction." In Proceedings of International Conference on Advanced Computing & Communication Technologies, pp. 255-263, 2016.

[9]. Altaf, S. et al. 2014. Fault diagnosis in Distributed Motor Network using Artificial Neural Network. (SPEEDAM2014) 22nd IEEE International Symposium on Power Electronics, Electrical Drives, Automation and Motion (2014).

[10]. R. Campagni, D. Merlini, R. Sprugnoli, and M. C. Verri, "Data mining models for student careers," Expert Systems with Applications, vol. 42, no. 13, pp. 5508–5521, Aug. 2015.