University Student's Learning Pattern Analysis and Prediction in LMS Using Data Mining Techniques

Vani Vasudevan
Al Yamamah University, Riyadh, KSA

Sultan Almuhanna
Al Yamamah University, Riyadh, KSA

Badr Almozini
Al Yamamah University, Riyadh, KSA

Abdulaziz Aljubair
Al Yamamah University, Riyadh, KSA

***Abstract*—Due to more and more people seeking higher education, learning management systems (LMS) need to constantly improve the structure to further manage student data. This paper relies on the need to discover, classify and measure the similarities between successful students' activities, whether it is in grades or in university sites visits. This paper will focus mainly on the best data mining algorithm suiting Moodle's (LMS) data.**

***Keywords*—predictions; data mining; higher education; student management; LMS improvements; teaching learning process**

I. INTRODUCTION

Universities in Riyadh need a new method to predict the marks their students will get in their courses, predicting the students' marks in small quizzes in the middle of the term is required to check whether the student confronts any challenges they face in their studies. This will help the University and instructors to handle their issues early and can upgrade their learning level with special assistance and avoid failures in their courses. With the help of this proposed system in place the Universities can achieve better graduated rates. Also universities can determine the intelligent students among them by considering the time factor (quiz duration). By that, universities can rank students in collections even between the brightest ones that got full marks in their quizzes with the help of various data mining techniques.

Data mining techniques classify, summarize and predict the future [1] to allow higher management in almost any field to predict the future based on previous behavior patterns and achieve their objectives whether it is revenue or future growth by adding new policies, changing old methods implementing papers and workshops. The need for the paper comes from the fact that with the huge repository of student data stored in the Universities databases, it requires necessary process and mining to get various benefits. From both students and the management level for better teaching and learning such as predicting the student performance, classifying week and strong students to provide help or appreciation if needed. Also, comparing different mining algorithms and find the best one for education plays a vital role. The algorithms for data mining can be divided into two approaches supervised and unsupervised learning, sometimes it's necessary to use both depending on the company work nature [2].

II. RELATED WORKS

There are many related articles and research papers written for this data mining subject in more than one field, some of which are specific and summarizes predicting student performance. The following sub sections will present these related works.

A. A Review on Predicting Student's Performance Using Data Mining Techniques

This paper provides an overview on data mining techniques and algorithms that has been used to predict students' performance[14-17]. It goes into detail about how each prediction algorithm can be utilized to effectively identify the most significant attributes in a student's data (Figure 1). The article also mentions that these prediction methods actually improve the students by making them more successful in their education and pursuit of higher achievements [4].

Fig. 1. List of common attributes and methods used in predicting student's performance [4].

The prediction algorithms used in this research paper include Decision Tree, Naive Bayes, K-Nearest Neighbor,

and Support Vector Machine. It thoroughly described each algorithm and listed its benefits as shown in Figure 1, as well as examples of other research papers and studies for each algorithm that used the prediction methods. It also highlighted the accuracy results for each algorithm from several testing sources (Table I).

TABLE I. ACCURACY RESULTS USING FIVE DIFFERENT ALGORITHMS

Methods	Attributes	Results	Authors
Naive Bayes	CGPA, Student Demographic, Scholarship	76%	Osmanbegovic and Suljic (2008)
	Student Demographic, High school background	50%	Ramesh et al. (2013)
	CGPA	75%	Jishan et al. (2015)
	Internal assessment, CGPA, Extra-curricular activities	73%	Mayilvaganan and Kapalnadevi (2014)
K-Nearest Neighbor	Psychometric factors	69%	Gray et al. (2014)
	Internal assessment, CGPA, Extra-curricular activities	83%	Mayilvaganan and Kapalnadevi (2014)
	Internal assessment, CGPA	82%	Bigdoli et al. (2003)
Support Vector Machine	Psychometric factors	83%	Sembiring et al. (2011)
	Internal assessment, CGPA, Extra-curricular activities	80%	Mayilvaganan and Kapalnadevi (2014)
	Internal assessment, CGPA	80%	Hamalainen et al. (2006)

B. The State of Educational Data Mining in 2009: A Review and Future Visions

This research article reviews the field of Educational Data Mining (EDM) in terms of current trends and historical events, as well as discusses the emphasis on predictions and how it became a popular method of prediction [5]. It explains the EDM methods in detail and mentions several prominent papers, gone through a survey by Romero & Ventura's (2007) from year 1995 to 2005, to support the article and specifies the relationship between said papers.

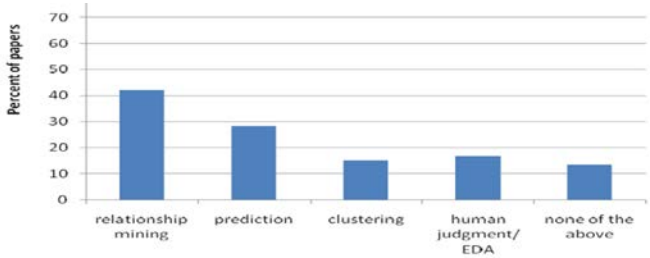

Fig. 2. The proportion of papers involving each type of EDM method, in Romero & Ventura's [2007] 1995-2005 survey [5].

An overview of the classifiers was given on the paper, but it didn't go into detail about specific algorithms they used or researched. However, it did give results of the research carried out for the 60 prominent papers that utilized EDM from 1995 to 2005. The result of that is 43% of the papers used relationship mining methods, 28% of them opted for various types of prediction methods (Figure 2). Figure 3 shows the survey of another chart that compares each type of EDM method across the researched papers.

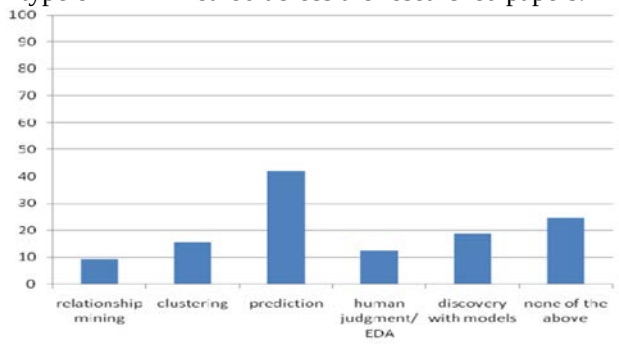

Fig. 3. The proportion of papers involving each type of EDM method.

The research work gave a general overview of some algorithms supported by the Romero & Ventura's [2007] 1995-2005 survey they used that compared 60 papers with their data mining results.

III. PROPOSED WORK

The proposed work studied and analyzed students' data from an anonymized dataset from MOODLE [3]. The dataset is offered for free by MOODLE for research purpose and it is extensively exploited in this work. The dataset consists of seven excel sheets with various information about the anonymous University students. The proposed system will use five sheets out of the seven available excel sheets and only take the predominant features (fields) to analyze them in using data mining packages. The filed names that structure the basis for further mining and analysis are listed below:

- Student ID: the ID of each student in the University.
- Age: the age of students.
- Gender: the gender of students.
- Final Result: the final result in a single word to indicate a pass or fail.
- Score: the score of students on regular tests.
- Site ID: ID of the University portal the student visits.
- Site Type: the type of the University site a student visits.
- Assessment ID: the ID of the assessment.
- Assessment Type: type of the assessment.
- Assessment weight: the weight that assessment carries.
- Sum of Clicks: the total amount of clicks each student accumulated when visiting University sites.
- Studied Credits: the amount of credits a student has earned.

A. Objectives

The objective of the proposed work is to provide an efficient prediction method that helps making decisions for education facilities and give them insight of future student performance. These objectives were accomplished by:

- Developing a tool that aids in predicting students' performance in learning management systems through data mining tools.
- Providing the ability to implement such tool to already existing LMS systems.

B. Paper Scope

The scope of this paper is to analyze the chosen MOODLE dataset and determine the best algorithm to use through data mining techniques in Weka. It is implemented by running various algorithms on the dataset and comparing their results with each other. Information is divided into training set and testing set and the testing set won't have the *final_result*. This will be used for comparison and calculating prediction accuracy.

To utilize only necessary data fields, the MOODLE data set have processed and cleansed and stored in a database. Then the Weka data mining tool is then used to mine the data retrieved from the database for prediction and ranking. Finally, the result is displayed on a graphical user interface (GUI) along with classification accuracy, alerts and charts.

Various data mining techniques and algorithms[18-20], such as Neural Network, Clustering, Regression, OneR, ZeroR, J48 and SimpleCart, are used to classify the data. It is then tested through various testing options such as training set, 10 fold cross-validation, and percentage split. The best model is finally identified based on the accuracy.

Clustering is used to group together a set of classes in a cluster to make them more similar to each other than the other classes in other clusters or groups. This can be accomplished by the available clustering methods in Weka such as Filtered Cluster, Simple K-Means, and Cascade Simple K-Means.

The results of the techniques and algorithms below are compared to determine the highest correctly classified algorithm for most of the dataset: The data mining techniques [11-13] are grouped under following categories:

- Function based classifiers:
  - Neural Network: [18] called "Multilayer Perceptron" in Weka, this classifier is mainly used for numerical attributes, since it accepts zeros and ones only.
- Rule based classifiers:
  - OneR: short for "One Rule," this accurate classification algorithm produces one rule for each predictor in the data.
  - ZeroR: it is made simpler than OneR, since it ignores all predictors and focuses on the class only.
- Tree based classifiers:
  - SimpleCart: this algorithm involves minimal cost-complexity pruning when implemented.
  - J48: J48 is used to generate pruned or unpruned C4.5 decision trees.

IV. Implementation

This section discusses the development process for creating and setting up the database schema, login page and interface. Connecting to the database is done through MySQL Server, and the paper's application is developed using Java Swing. The login form's validations, as well as loading up the data in the main form of the application are also mentioned. Also, the interface functionalities, capabilities and APIs used in the application are discussed in the following sub section.

A. Setting up the Database

1) Creating Database Schemas and Tables

The implementation started with cleansing of intended excels files, by removing unused columns and null values and then necessary tables with constraints were created in MySQL. Figure 4 shows various tables created using the MySQL Command Line client.

Fig. 4. List of Schema Tables Within MySQL.

B. Software Dependencies

1) Application Programming Interfaces

The application depends on several Application Programming Interfaces (APIs) to function properly. There are a total of 5 APIs required for the application to work, they are:

- Weka API [6].
- MySQL Connector/J version 5.1.45 [7].
- JMathPlot API [8].
- Resultset to XML API [9].
- Cage 1.0 Captcha API [10].

2) Database Schema and Tables

The application also depends on a database schema on the MySQL program, since it reads tables directly from the database. The MySQL server needs to be running and

connected with the application to ensure smooth interaction between them.

C. GUI Development

1) Login Screen

The login screen is the first screen that appears as the application starts running, and it verifies the user login information with a pre-set username and password (Figure 5). It also has an extra layer of security in the form of an image captcha code that has to be entered to proceed with the login. The captcha images are generated with the form, a set of one hundred captcha images, they're then shuffled and one of them is randomly gets chosen by the application to use for validation. The captcha code image can also be refreshed by simply pressing the reload captcha button on the login form; this can be done as many times as necessary. Once all correct values are entered into the text fields, the login screen calls the main frame for the application and the current frame is disposed.

Fig. 5. Login screen.

2) Main Application Screen

The interface is developed by using Java Swing in Netbeans IDE. The interface comprises of login and main application screen (Figure 6). The main screen displays a page that connects to the MOODLE database with required tables. Appropriate table were chosen for best mining along with providing the ability to choose classifiers. Also, provides the ability to choose the type of mining intended and shows the chosen table on the right along with giving type of validation. After necessary classifier and test options were chosen, the data can be classified by using Classify. All these functionalities are embedded in main screen by using Weka data mining tool API [6].

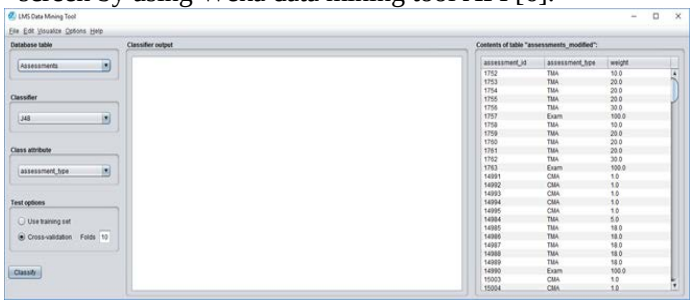

Fig. 6. Main application screen.

V. Analysis And Inferences

From the MOODLE database, five predominant tables were considered to draw inferences. These inferences were made for knowledge discovery by choosing the best fit in classification and the best fit for options whether it's cross fold validation or training set. Also, comparing different classifiers, such as J48 (a C4.5 decision tree classifier), PART and OneR, with different class attributes for each classification. This is done on prefixed tables that were created by joins for knowledge discovery. Classification approach is used for necessary prediction rather clustering as most of the date leads to supervised learning.

A. Table: Students Grades

TABLE II. J48 and PART Comparison for Students Grades

Classifier	Correctly classified	Incorrectly classified	Mean absolute error	Root relative squared error	Number of instances
J48	62.4096%	37.5904%	0.2613	91.6418%	3735
PART	64.0964%	35.9036%	0.3575	88.4723%	

The attributes chosen from student grades table are *final_result* and the semester summation of both *sum_of_clicks* and *score* for each active student. The benefit is to determine passed students semester summation range and to figure out their online activity, which made them take good marks. Also see what the ranges are of failed and withdrawal students' scores and online activities. *Final_result* will be used as classification and a value in the attribute *final_result* (distinction) will be removed. Due to the effect of unrelated situations and removing it also will further more improve classification accuracy and reduce the number of leaves along with the size of tree.

After experimenting with many classifiers and choosing the best, along with tree pruning and over fitting to get the best accuracy and choosing ones with most meaningful trees or rules. Using the same attribute *final_result* for PART and J48, PART results in higher accuracy with 64.0964% correctly classified instances. As for the root relative squared error, J48 slightly comes on top here compared to the PART classifier with a 3.1695% difference (Table II). Hence, the PART classifier for the given table is better for classification.

1) Approach One: J48 Classifier

Using the algorithm J48 for classification and the number of rows is 3735 for table student grades. 10 folds cross validation option divides data equally into separate folds. The result is 2313 correctly classified instances (62.4069%) and 1404 incorrectly classified, number of leaves is 6, the tree size is 11 and the root relative absolute error is 82.6423% (Table II). Furthermore, when using the distinction value, the accuracy decreases by 5% and the number of leaves increases to 8 making it more complex.

2) Approach Two: PART Classifier

While using PART rule classifier with use training option for the table student grades. The number of rules is

17 with 64.0964% correctly classified instances and 35.9036% incorrectly classified instances (Table II). The mean absolute error is 0.2475 and the root mean squared error 88.3825% and the number of instances used in this classification is 3735 record.

B. Table: Exam Results

TABLE III. J48 AND PART COMPARISON FOR EXAM RESULTS

Classifier	Correctly classified	Incorrectly classified	Mean absolute error	Root relative squared error	Number of instancing
J48	61.3501%	38.6499%	0.2582	88.4891%	41731
PART	81.2609%	18.7391%	0.1863	69.8412%	

The creation of a new data set by joining three table's student information, assessment and types of assessments. The chosen attributes are *final_result* for all students and their assessments types and each type *sum_of_score*. Each active student would have three records, a score summation for each *assessment_type* in that three records. To see students' progress in each *assessment_type* and their end results. Also find out the assessment types of students that get bad grades and to see what the passed student's summation for every type.

After experimenting on this table for many classifiers and many test options, the above chosen classifiers have the best accuracy and meaningful results. After Testing it for example OneR, naïve Bayes and for decision table in rules classifiers, the J48 describes patterns in *sum_of_score* and the assessments for them to get a pass grade, fail, or withdrawn. The second test on PART classifier was on patterns between *score* and *final_result* in each type to see witch types have the most failed results. Hence depending on the need of the user whether he wants to classify upon *final_result* or assessments type, the finalised two classifiers (Table III) for each scenario is the best choice.

1) Approach Two: J48 Classifier

Results are for the table exam results were 25002 correctly classified instances (61.3501%) and 38.6499% incorrectly classified, the relative absolute error is 88.4891%. The number of leaves is 24 and the tree size is 45. With the *final_result* as the class attribute, using J48 algorithm (10 folds cross-validation option) with 41731 total instances, the accuracy is high and the tree is simple to understand.

2) Approach Two: PART Classifier

While using algorithm PART rule classifier for table exam results table with use training set option and *assessment_type* as the class attribute. Total instances used are 41725 records and the number of rules is 130. The result is 33792 correctly classified instances (81.2009%) and 18.7391% for incorrectly classified. The mean absolute error is 0.163 and the root relative squared error is 69.8412%.

C. Table: Inactive Students

TABLE IV. ONER AND J48 COMPARISON FOR INACTIVE STUDENTS

Classifier	Correctly classified	Incorrectly classified	Mean absolute error	Root relative squared error	Number of instances
J48	80.336%	19.664%	0.2044	96.6481%	5416
OneR	80.4838%	19.5162%	0.1301	109.0523%	

A dataset created for inactive students that do not have any scores of online activity and may hold withdrawal or fail for *final_result*. The attribute choice is *gender*, *age*, *studied_credits* and *final_result* from table student information. The reason is to check if there is any pattern for failing or withdrawal, like if there is any similarity in age and *studied_credits* also *gender* that may be a major factor in those results and inactivity.

After experimenting with the above table we conclude that majority of students that are inactive have few credit hours. Each *gender* type has its own rules in less than 30 credits with different *final_result*. We also conclude that those students are freshmen students who may not understand university rules and procedures or they have changed their study plans. In results, we conclude after using same classifier *final_result* for both operations that J48 tree has better accuracy. The selection criteria here depended on the root relative squared error, which was 96.6481% for J48 and 109.0523% for OneR (Table IV). Hence, the J48 classifier is better for classification for inactive students table.

1) Approach Two: J48 Classifier

Using J48 tree training set option for table inactive students and *final_result* as classifier nominal. The number of instances used 5416 records, the number of leaves is 3 and the size of the tree is 6. The results of this instance are 4343 correctly classified instances (80.336%) and 19.664% for incorrectly classified instances, the mean absolute error is 0.2044 and the root relative squared error is 96.6481%. Figure 7 displays the pruned tree result for inactive student using J48 classifier.

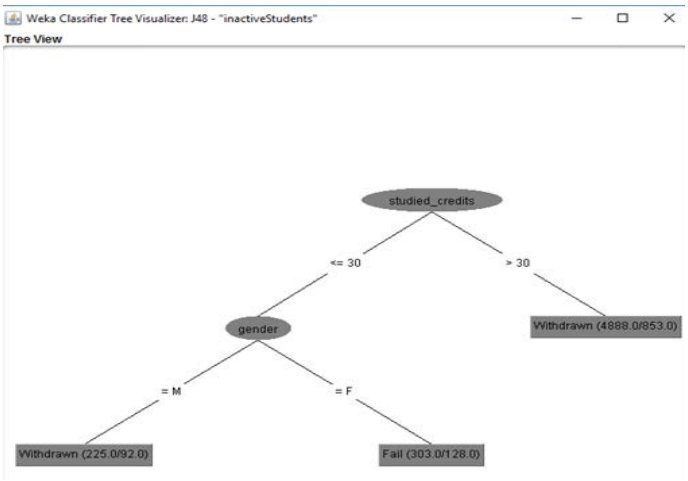

Fig. 7. Pruned Tree Structure for Inactive Students Table.

2) Approach Two: OneR Classifier

Using OneR rule classifier with use training set option for the same dataset given above. With 5416 total number of instances, the correctly classified instances are 4359 (80.4838%) and 19.5162% for incorrectly classified instances. The root relative squared error 109.9541%. For rules used in OneR classifier, the *final_result* results in a fail or withdrawn if *studied_credits* is less than 35 and more than 35, respectively.

D. Table: Students By Gender

TABLE V. ONER AND J48 COMPARISON FOR STUDENTS BY GENDER

Classifier	Correctly classified	Incorrectly classified	Mean absolute error	Root relative error	Number of instances
J48	81.6867%	18.3133%	0.2869	95.2127%	3735
OneR	82.3025%	17.6975%	0.177	105.7594%	

A data set created for differentiation for each *gender* in performance from table student information, assessments and online activity. Choosing attributes *gender*, *sum_of_score* and *sum_of_clicks* along with the *final_result*. This table is created to see classify results on *gender* to determine better performance in activity and overall scores between male and female. Also the pattern of each *gender*, *score* and online activity and what are the effects on their *final_result*.

After experimenting with the above table we conclude that using OneR classifier is the best choice, because OneR's accuracy is higher than J48 as well as the mean absolute error is lower having 0.177 and 0.2869 for J48 (Table V).

1) Approach Two: J48 Classifier

While using J48 tree classifier for table gender performance use training set option and *gender* as the classifier. Total instances used are 3735 records, the leaves count is 11 and the size of the tree is 19. The result is 81.6867% correctly classified instances and 18.3133% incorrectly classified instances. The root relative squared error is 95.2127% and the mean absolute error is 0.2869.

2) Approach Two: OneR Classifier

Using OneR rule classifier with use training set option for the same dataset given above. Results of this instance show that the correctly classified instances are 3074 (82.3025%), the incorrectly classified instances as 17.6975% and the root relative squared error 105.7594%.

VI. CONCLUSION

This paper aims to create a tool that can benefit various universities and higher education facilities by allowing them to predict students' performance based on their grades and learning patterns. The system will make heavy use of data mining of students' data to come up with the prediction patterns, making use of various tools to accomplish this.

In this paper, it was concluded that the classification results obtained through the various data mining algorithms were sufficient, and they were satisfactory to prove the strength of data mining when used to predict student performance in the future. Table 6 displays a comparison table to summarize all the optimum classifiers tested in this paper.

TABLE VI. CLASSIFICATION SUMMARY TABLE

Table	Classifier	Correctly classified	Incorrectly classified	Mean absolute error	RRSE[a]	Number of instances
Students Grades	PART	64.09%	37.59%	0.26	91.64 %	3735
Exam Results	PART	81.26%	18.73%	0.18	69.84 %	43731
Inactive Students	J48	80.33%	19.66%	0.20	96.64 %	5416
Student ByGender	OneR	82.30%	17.69%	0.17	105.75 %	3735

[a.] Root relative squared error.

ACKNOWLEDGMENT

We thank the management of Al Yamamah University, KSA for supporting financially to publish our research work.

REFERENCES

[1] OULAD Dataset. (2015). Retrieved from https://archive.ics.uci.edu/ml/machine-learning-databases/00349/

[2] Amjad Abu Saa, (2016) "Educational Data Mining & Students' Performance Prediction", International Journal of Advanced Computer Science and Applications(ijacsa),Vol 7, Issue 5, DOI:10.14569/IJACSA.2016.070531

[3] MOODLE Research. (2017) "Learn MOODLE August 2016" anonymised data set., Retrieved from http://research.MOODLE.net/158/

[4] Amirah Mohamed Shahiri, Wahidah Husain, Nur'aini Abdul Rashid. (2015), A Review on Predicting Student's Performance Using Data Mining Techniques,Procedia Computer Science,Volume 72 ,Pages 414-422,ISSN 1877-0509.

[5] Ryan S.J.d. Baker, Kalina Yacef. (2009). The State of Educational Data Mining in 2009: A Review and Future Visions, Journal of Educational Data Mining, v1 n1 p3-16, ISSN: EISSN-2157-2100.

[6] Weka Data Mining Software. Official website. (2017). Retrieved from https://www.cs.waikato.ac.nz/ml/weka/downloading.html

[7] MySQL Connector/J ZIP Archive. Official website. (2017). Retrieved from https://dev.mysql.com/downloads/connector/j/5.1.html

[8] JMathPlot: interactive 2D and 3D plots. Official website. (2015). Retrieved from https://github.com/yannrichet/jmathplot/blob/master/dist/jmathplot.jar

[9] Resultset to XML API. (2015). Retrieved from https://sourceforge.net/projects/finalangelsanddemons/files/rs2xml.jar/download

[10] Cage 1.0 Captcha API. Official website. (2011). Retrieved from http://repo1.maven.org/maven2/com/github/cage/cage/1.0/cage-1.0.jar

[11] Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal, (2016) ,"Data Mining: Practical Machine Learning Tools and Techniques".

[12] Jiawei Han, Micheline Kamber, and Jian Pei, (2011), "Data Mining: Concepts and Techniques,".

[13] Richard Roiger and Michael Geatz, (2002) "Data Mining: A Tutorial-based Primer".

[14] U. bin Mat, N. Buniyamin, P.M. Arsad, R. Kassim, An overview of using academic analytics to predict and improve students' achievement:

A proposed proactive intelligent intervention, in: Engineering Education (ICEED), 2013 IEEE 5th Conference on, IEEE, 2013, pp. 126-130.

[15] D. M. D. Angeline, Association rule generation for student performance analysis using apriori algorithm, The SIJ Transactions on Computer Science Engineering & its Applications (CSEA) 1 (1) (2013) p12-16.

[16] M. Mayilvaganan, D. Kalpanadevi, Comparison of classification techniques for predicting the performance of students academic environment, in: Communication and Network Technologies (ICCNT), 2014 International Conference on, IEEE, 2014, pp. 113-118.

[17] S.T. Jishan, R.I. Rashu, N. Haque, R.M. Rahman, Improving accuracy of students final grade prediction model using optimal equal width binning and synthetic minority over-sampling technique,Decision Analytics, 2 (1) (2015), pp. 1-25

[18] V. Oladokun, A. Adebanjo, O. Charles-Owaba, Predicting students academic performance using artificial neural network: A case study of an engineering course,The Pacific Journal of Science and Technology, 9 (1) (2008), pp. 72-79

[19] V. Ramesh, P. Parkavi, K. Ramar, Predicting student performance: a statistical and data mining approach, International Journal of Computer Applications, 63 (8) (2013), pp. 35-39

[20] T. Mishra, D. Kumar, S. Gupta, Mining students' data for prediction performance, in: Proceedings of the 2014 Fourth International Conference on Advanced Computing & Communication Technologies, ACCT '14, IEEE Computer Society, Washington, DC, USA, 2014, pp. 255-262. doi:10.1109/ACCT. 2014.105. URL http://dx.doi.org/10.1109/ACCT. 2014.105.