Machine Learning Algorithms and Predictive Models for Undergraduate Student Retention

Ji-Wu Jia, *Member IAENG*, Manohar Mareboyana

***Abstract*---In this paper, we have presented some results of undergraduate student retention using machine learning algorithms classifying the student data. We have also made some improvements to the classification algorithms such as Decision tree, Support Vector Machines (SVM), and neural networks supported by Weka software toolkit. The experiments revealed that the main factors that influence student retention in the Historically Black Colleges and Universities (HBCU) are the cumulative grade point average (GPA) and total credit hours (TCH) taken. The target functions derived from the bare minimum decision tree and SVM algorithms were further revised to create a two-layer neural network and a regression to predict the retention. These new models improved the classification accuracy.**

***Index Terms*---Decision Tree, Machine Learning, Neural Network, Student Retention, Support Vector Machines**

I. INTRODUCTION

THIS paper studies the HBCU undergraduate student retention [7]. We explore the effectiveness of machine learning techniques to determine factors that influence student retention at an HBCU and create retention predictive models [9]-[15].

In general, learning algorithms attempt to maximize classification accuracy (percentage of instances classified correctly) to obtain a correct solution of high quality (length, efficiency) [2].

We started collecting data from the HBCU Fall 2006 full-time and first-time undergraduate students, and tracked these students' activities in the following six years from Fall 2006 to Fall 2011. The data was queried from the Campus Solution database. The six-year training data set size is 771 instances with 12 attributes shown in Table I [5], [9]. The HBCU undergraduate six years retention rate 44.9% was derived from the six-year training data set [5]. The HBCU six-year training data set numeric attributes and statistics are shown in Table II.

We classified the data under two groups – "Retention" – students who were retained in the HBCU and "No Retention" – students who were not retained in the HBCU.

Manuscript received July 3, 2013; revised July 22, 2013.
Ji-Wu Jia is a PhD candidate in Department of Computer Science, Bowie State University, 14000 Jericho Park Road, Bowie, Maryland 20715, USA. Jiaj1130@students.bowiestate.edu
Manohar Mareboyana is Professor in Department of Computer Science, Bowie State University, 14000 Jericho Park Road, Bowie, Maryland 20715, USA. MMareboyana@bowiestate.edu

TABLE I
LIST OF DATA SET ATTRIBUTES

Number	Name	Description	Type
1	GPA	The last cumulative GPA while student enrolled	Number
2	TCH	The max total credit hours taken while student enrolled	Number
3	School	School that student enrolled in Fall 2006	Text
4	Plan	Academic program that student enrolled in Fall 2006	Text
5	Distance	Commuting distance of the student	Number
6	Gender	Student gender	Text
7	Age	Student age	Number
8	Race	Student race	Text
9	FINAID	The amount of financial aid that student awarded in Fall 2006	Number
10	SAT I Math	Student SAT I Math score	Number
11	SAT I Verb	Student SAT I Verbal score	Number
12	Retention	If student graduated or enrolled in Fall 2011 then yes, else no	Text

TABLE II
TRAINING DATA SET NUMERIC ATTRIBUTES

Naive Bayes	No Retention		Retention	
Attribute Name	Mean	Std. Dev.	Mean	Std. Dev.
GPA	1.9371	±0.8913	2.8864	±0.4276
TCH	50.9574	±35.9515	149.3327	±23.8385
Distance	5.8922	±9.6504	3.3919	±6.942
Age	18.6056	±1.5723	18.4133	±0.7819
FINAID	8165.95	±6924.45	8706.66	±7101.55
SAT I Math	425.6	±57.48	425.77	±59.7
SAT I Verb	442.27	±50.63	441.66	±54.63

Firstly, we used Weka to classify the cohorts' six-year training data set using different machine learning algorithms with a goal to maximize classification accuracy (percentage of instances classified correctly). The models derived by J48 decision tree, Simple Logistic, Naïve Bayes and JRip algorithms gave classification accuracies of about 94.03%.

Secondly, we pruned the J48 decision tree to get the bare minimum decision tree, and we found the learning rule for the HBCU undergraduate student six-year retention. Then we created a neural network model for predicting retention, the model's accuracy was 94.16%. We improved the model's performance from 94.16% to 94.42%.

In addition, we used SVM algorithm and created a regression that proved the selection of the two major factors that affect the retention, which are cumulative GPA and total credit hours taken.

ISBN: 978-988-19252-3-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Furthermore, we extended the six-year training classification to seven student academic level classifications, which are six-year, five-year, four-year, three-year, two-year, one-year, and zero-year classification. We created retention models for each level. These models are validated by each independent corresponding test data sets. The predictive accuracy of six-year neural network model is 93.05%.

In the following sections, we describe the methodology and algorithms.

II. METHODOLOGY

A. Weka J48 Decision Tree

J48 decision tree is an implementation of the C4.5 algorithm in the WEKA. C4.5 is an algorithm used to generate a decision tree developed by Ross Quinlan. C4.5 is an extension of Quinlan's earlier ID3 algorithm. C4.5 builds decision trees from a set of training data in the same way as ID3, using the concept of information entropy. At each node of the tree, C4.5 chooses the attribute of the data that most effectively splits its set of samples into subsets enriched in one class or on other. The splitting criterion is the normalized information gain (difference in entropy). The attribute with the highest normalized information gain is chosen as a leaf to make the decision [1], [8]. The information entropy and information gain are defined below [2].

$$Entropy(S) = \sum_{j} -p_j \log_2 p_j \quad (1)$$

Where p_j is the fraction of *j* type examples in *S*.

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (2)$$

Where Values(A) is the set of all possible values for attribute A, and S_v is the subset of S for which attribute A has value v (i.e., $S_v = \{ s \in S \mid A(s) = v \}$).

We applied J48 decision tree to the six-year training data set. The J48 decision tree algorithm classified 94.03% of the instances correctly and 5.97% of the instances incorrectly. The Weka created J48 decision tree for the six-year training data set is shown in Fig. 1. The numbers in (parentheses) at the end of each leaf tell us the number of instances in this leaf. If one or more leaves were not pure (= all of the same class), the number of missing classified instances would also be given, after a slash (/).

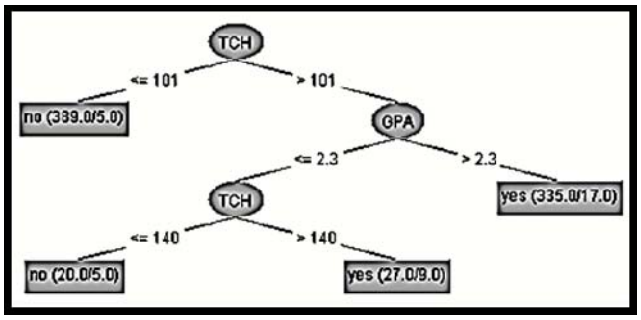

Fig. 1. J48 decision tree

B. Weka J48 Decision Tree Pruning

The goal of a decision tree learning algorithm is to have the resulting tree to be as small as possible to avoid overfitting to the training set [2], [8], [16], [17]. However, finding a minimal decision tree (nodes, leaves, or depth) is an NP-hard optimization problem [2]. We created an algorithm to further prune the Weka J48 decision tree to bare minimum decision tree. The algorithm is given below:

Step1: Prune the deepest node and assign a new leaf as the prediction of a solution to the problem under consideration.
Step 2: Calculate the new tree's estimated accuracy
Step3: If the new tree's estimated accuracy is improved, then the pruning is successful; otherwise stop, and exit.

We pruned the Fig. 1 Weka J48 decision tree using above algorithm below:

Step1: Pruned the parent of two leaves: no(20/5) and yes(27/9) and assigned the new leaf as "no retention (47/23)"
Step2: Calculated the new tree's estimated accuracy
True (data) to true (pruned tree) = (335 – 17) = 318
False (data) to false (pruned tree) = ((389 + 47) – ((27 – 9) + 5 + 5)) = 408
Pruned tree correct = 318 + 408 = 726
False (data) to true (pruned tree) = 17
True (data) to false (pruned tree) = ((27 – 9) + 5 + 5) = 28
Pruned tree incorrect = 17 + 28 = 45
Pruned tree accuracy = (726 / (726 + 45)) * 100% = 94.16% > 94.03% of the Weka J48 decision tree accuracy,
therefore, the pruning is successful and the heuristic bare minimum decision tree is shown in Fig. 2.

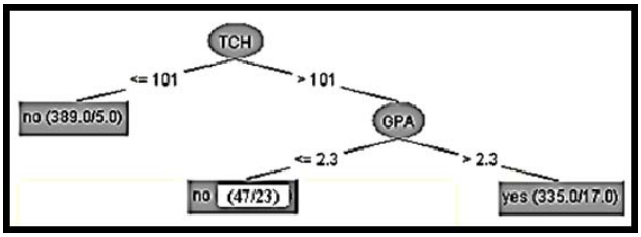

Fig. 2. Bare minimum decision tree

ISBN: 978-988-19252-3-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

The target function depicted in a decision tree can be represented as a first order logic rule by following each path in the tree from the root to a leaf and creating a rule with the conjunction of tests along the path as an antecedent and the leaf label as the consequent [2]. The target function in the bare minimum decision tree (Fig. 2) can be expressed as a first order logic rule as given in (3).

$$((TCH > 101) \cap (GPA > 2.3)) \rightarrow \text{Re}\,tention \qquad (3)$$

C. Six-year Neural Network Model

We used (3) derived from the bare minimal decision tree to build a two-layer Neural Network that can predict the HBCU undergraduate student six-year retention as illustrated in Fig. 3.

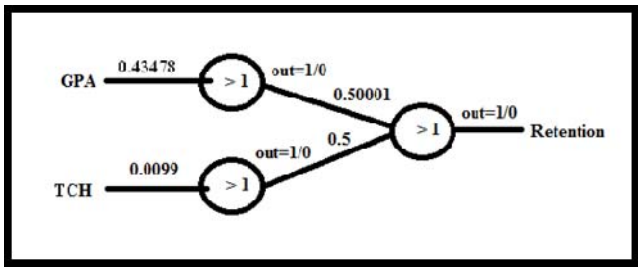

Fig. 3. Six-year training neural network model
(GPA – input, TCH -- input)

For example if a student's last cumulative GPA is 2.4 and total credit hours taken is 102, then the 2-layer neural network can determine whether the student will remain in school using the following calculation:

FirstLayer

$2.4 * 0.43478 = 1.04 > 1 \rightarrow 1$

$102 * 0.0099 = 1.01 > 1 \rightarrow 1$

SecondLayer

$1 * 0.50001 + 1 * 0.5 = 1.00001 > 1 \rightarrow 1(\text{Re}\,tention)$

D. Improved Model Accuracy

We also created an algorithm to improve the two-layer neural network model's accuracy shown below.

Step1: Test neural network model using the six-year training data set
Step2: Put the six-year training data set and neural network model output values to an array
Step3: Sort the array by the model output values
Step4: On the boundary of 1/0 of the model output values adjust the GPA and TCH weight with $W_0 \pm \Delta\omega$
Step5: Calculate the new model's estimated accuracy
Step6: If the new model's estimated accuracy is improved then the adjustment is successful; otherwise stop, and exit.

We used above algorithm and adjusted the weight of input GPA from 0.43478 to 0.437255, which improved the six-year neural network model's accuracy shown in Fig. 4.

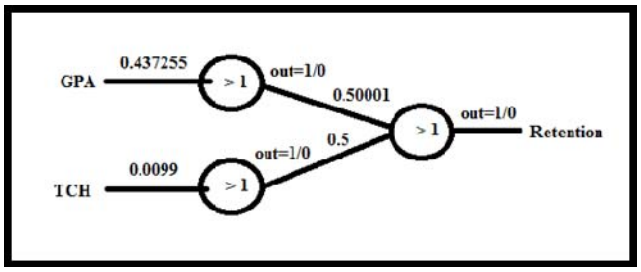

Fig. 4. Improved six-year neural network model

The improved six-year neural network shown in Fig. 4 was tested against the 771 training data set. The results showed that 728 are true and 43 are false, and the model's accuracy is 94.42%. The detail is shown in the following:

True (data) to true (model) = 321
False (data) to false (model) = 407
Model correct = 321 + 407 = 728

False (data) to true (model) = 18
True (data) to false (model) = 25
Model incorrect = 18 + 25 = 43
Model accuracy = (728 / (728 + 43)) * 100% = 94.42%.
Model in-sample error = (43 / 771) * 100% = 5.58%.

E. SVM Classification

Support vector machine is a supervised learning algorithm and it has the three following properties [3].

1) SVM constructs a maximum margin separator--a decision boundary with the largest possible distance to example points.

2) SVM creates a linear separating hyperplane, but it has the ability to embed the data into a higher-dimensional space, using the so-called Kernel Trick. This means the hypothesis space is greatly expanded over methods that use strictly linear representations.

3) SVM is a nonparametric method. It retains training examples, and potentially needs to store them all. In practice, it often ends up retaining only a small fraction of the number of examples; sometimes as few as a small constant times the number of dimensions. The SVM combines the advantages of nonparametric and parametric models: they have the flexibility to represent complex functions, but they are resistant to overfitting.

To prove the validity of our model, that the last cumulative GPA and total credit hours (TCH) taken are the major factors which affect the HBCU undergraduate student retention, we use the six-year training data set to model the retention by SVM algorithm. These points are shown in the normal space with a curve boundary in Fig. 5. The points above the curve correspond to "retention" and the ones below the curve

ISBN: 978-988-19252-3-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

correspond to "no retention." The x-axis is six-year cumulative GPA and the y-axis is six-year total credit hours taken.

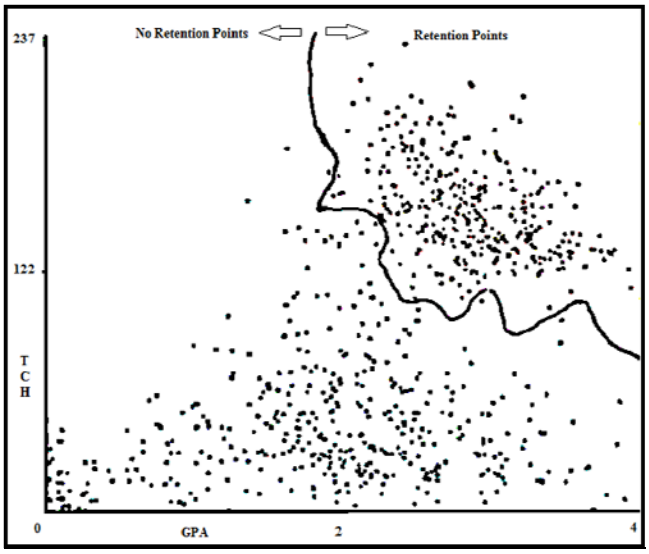

Fig. 5. The normal data set space

We mapped the data from the normal space x into a z space using the following transformed function $\phi(x)$ (Kernel function) [4], [18], [19].

$$x = \begin{bmatrix} 1 \\ GPA \\ TCH \end{bmatrix}$$
$$z = \phi(x) = \begin{bmatrix} 1 \\ GPA^2 \\ TCH^2 \end{bmatrix} \quad (4)$$

In the z space, the separating curve is changed to a line, and we created the retention regression as (5).

$$\frac{21857.5}{19.5} GPA^2 + TCH^2 - 21857.5 > 0 \quad (5)$$

The model has been tested and the model's accuracy is 93.64%.

We also applied the improvement performance algorithm to the above model and created the following (6):

$$\frac{20800}{21} GPA^2 + TCH^2 - 20800 > 0 \quad (6)$$

The new model's accuracy is improved to 94.29%.

III. RESULTS

A. Training Data Sets Results

Based on the HBCU six-year (2006-2011) training data set, we also collected training data sets corresponding to five-year (2006-2010), four-year (2006-2009), three-year (2006-2008), two-year (2006-2007), one-year (2006), and zero-year (which used high school GPA to replace the undergraduate academic data) periods for the Fall 2006 cohort students [5], [6], [9]. We applied several Weka algorithms, J48 decision tree, Simple Logistic, Naïve Bayes, and JRip on the seven training data sets. The results are shown in Table III.

TABLE III
TRAINING DATA SET ACCURACIES

Accuracy (%)							
Number of year	0-Year	1-Year	2-Year	3-Year	4-Year	5-Year	6-Year
J48	59.66	66.54	81.19	85.86	89.49	91.44	94.03
Simple Logistic	65.24	68.61	80.16	85.21	90.66	92.09	94.03
Naïve Bayes	60.57	67.06	78.47	84.96	88.72	91.31	93.39
JRip	59.27	67.06	80.16	84.57	90.92	91.70	94.03

We applied the pruning algorithm to the seven J48 decision trees, performance improvement algorithms on predictive models, and then we created seven different retention neural networks models by the seven student academic training data sets. The new results are shown in Table IV.

TABLE IV
RETENTION MODELS BY YEAR

Accuracy (%)							
Number of year	0-Year	1-Year	2-Year	3-Year	4-Year	5-Year	6-Year
J48	59.66	66.54	81.19	85.86	89.49	91.44	94.03
Neural Network	59.92	68.35	81.71	86.64	90.53	92.48	94.42
Improved	0.26	1.81	0.52	0.78	1.04	1.04	0.39

The HBCU undergraduate student retention results by seven student academic levels are shown in Table V.

TABLE V
THE TRAINING RETENTION RESULTS BY YEAR

Number of year	0-Year	1-Year	2-Year	3-Year	4-Year	5-Year	6-Year
GPA		2.391	2.04	2.443	2.333	2.292	2.3
TCH			36	60	81	106	101
Distance (Mile)	6.9	5.2					
HS GPA	2.28						

B. Test Data Sets Results

After the HBCU retention models are created by the seven student academic levels training data sets, we further collected test data sets from the Fall 2007 full-time and first-time undergraduate student, and tracked these students' activities in the following six years from Fall 2007 to Fall 2012. The data was also queried from the Campus Solution database. The six-year test data set size is 820 instances with 12 attributes, same as training data set attributes, which are shown in Table I. The HBCU undergraduate six years retention rate 43.5% was

ISBN: 978-988-19252-3-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

derived from the six-year test data set. We also collected test data sets for five-year (2007-2011), four-year (2007-2010), three-year (2007-2009), two-year (2007-2008), one-year (2007), and zero-year (which used high school GPA to replace the undergraduate academic data) periods of the Fall 2007 cohort students. The six-year test data set numeric attributes and statistics are shown in Table VI.

TABLE VI
TEST DATA SET NUMERIC ATTRIBUTES

Naïve Bayes	No Retention		Retention	
Attribute Name	Mean	Std. Dev.	Mean	Std. Dev.
GPA	1.8364	±0.9129	2.9063	±0.4445
TCH	38.8145	±27.4391	125.7639	±18.2481
Distance	6.9231	±10.8022	4.67	±8.2694
Age	18.1685	±0.5397	18.0756	±0.3778
FINAID	9337.6	±7080.19	9886.25	±6765.68
SAT I Math	390.75	±134.27	406.15	±120.36
SAT I Verb	405.52	±136.80	420.1	±117.72

We applied the same Weka algorithms on the seven test data sets. The results are presented Table VII.

TABLE VII
TEST DATA SET ACCURACIES

Accuracy (%)							
Number of year	0-Year	1-Year	2-Year	3-Year	4-Year	5-Year	6-Year
J48	52.2	62.9	80.2	85.1	88.1	91.7	93.8
Simple Logistic	59.3	63.9	76.6	83.4	88.9	92.9	93.8
Naïve Bayes	57.7	64.5	74.8	83.5	88.2	91.6	93.5
JRip	59.2	65.9	78.9	82.9	88.7	91.8	94.3

The HBCU undergraduate student retention test data set' results by year are shown in Table VIII.

TABLE VIII
THE TEST DATA SET RETENTION RESULTS

Number of year	0-Year	1-Year	2-Year	3-Year	4-Year	5-Year	6-Year
GPA		2.545	2.2	2.111	2.3	2.381	
TCH			34	49	69	76	99
HS GPA	2.78						

C. Retention Models' Validation

We used the six-year training neural network model shown in Fig. 4 to predict the six-year test data set and the model predicted correct students are 763, and errors are 57. The model's predicted accuracy is 93.05% and the model's out-of-sample error is 6.95% shown in Table IX. The out-of-sample error (6.95%) is close to the model in-sample error (5.58%), which was calculated earlier. We validated our models by the seven student academic levels of test data sets.

TABLE IX
SIX-YEAR PREDICTIVE RESULTS

Predictive Value	Correct	Error	Total
Retention	314	14	328
No-Retention	449	43	492
Total	763	57	820
Percentage (%)	93.05	6.95	100

The summary of the predictive accuracies for the seven neural networks is shown in Table X.

TABLE X
MODELS' PREDICTIVE ACCURACIES

Accuracy (%)							
Number of year	0-Year	1-Year	2-Year	3-Year	4-Year	5-Year	6-Year
Model	59.92	68.35	81.71	86.64	90.53	92.48	94.42
Predicted test data	55.85	64.88	79.88	82.44	86.71	87.93	93.05
Difference	4.07	3.47	1.83	4.20	3.82	4.55	1.37

The summary of the predictive errors for the seven neural networks is shown in Table XI.

TABLE XI
MODELS' PREDICTIVE ERRORS

Error (%)							
Number of year	0-Year	1-Year	2-Year	3-Year	4-Year	5-Year	6-Year
In-Sample error	40.08	31.65	18.29	13.36	9.47	7.52	5.58
Out-of-Sample Error	44.15	35.12	20.12	17.56	13.29	12.07	6.95
Difference	4.07	3.47	1.83	4.20	3.82	4.55	1.37

We used the following equation to validate the seven predictive models [4].

$$E_{out}(g) - E_{in}(g) <= \sqrt{\frac{1}{2N}\ln\frac{2M}{\delta}} \quad (7)$$

Where $E_{out}(g)$ is the model's out-of-sample error, $E_{in}(g)$ is the model's in-sample error, N is the size of training data set, and M is the size of test data set, δ is tolerance.

The size of training data set = 771, the size of test data set = 820, and the tolerance δ = 0.05, then the equation (7) became (8).

$$E_{out}(g) - E_{in}(g) <= \sqrt{\frac{1}{2N}\ln\frac{2M}{\delta}} \approx \sqrt{\frac{1}{2*771}\ln\frac{2*820}{0.05}} =8.21\% \quad (8)$$

Based on equation (8) the difference in errors ($E_{out}(g) - E_{in}(g)$) should be less than 8.21%. The differences in errors shown in Table XI are less than 8.21%, so the seven retention predictive models are validated by the seven test data sets.

ISBN: 978-988-19252-3-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

IV. CONCLUSIONS

The goal of a decision tree learning algorithm is to have the resulting tree to be as small as possible, per Occam's razor [2]. The Weka J48 decision tree is not a minimum decision tree. We further pruned it, improved the estimated accuracy, and simplified the learning rules for the HBCU undergraduate student retention. This is an effective way to find the most important factors that affect the retention and then to build the simplifying retention models. Occam's razor is the machine learning principle, where the "razor" is meant to trim down the explanation to the bare minimum that is consistent with the data. The simplest model that fits the data set is also the most plausible.

After the retention model was created, we used learning feedback based performance improvement algorithm to improve the neural networks models' accuracy.

We studied the HBCU undergraduate student retention in the six years period and split the six years to seven student academic levels. We classified and created retention models for each level, and then we validated the models by seven independent corresponding test data sets. The six-year retention model's out-of-sample error is 6.95%, which is close to the in-sample error 5.58%.

The SVM is currently the most popular approach for retention supervised learning. For the nonlinear SVM boundary, we used transformed function (Kernel function) to change the normal space x to a z space for linear separation, and then we created a retention regression. This is an effective way to directly create a retention regression without using any machine learning tool such as Weka. The SVM retention model used two significant attributes GPA and TCH, and the model's accuracy was improved to 94.29%.

REFERENCES

[1] M. H. Dunham, "Data mining introductory and advanced," ISBN 0-13-088892-3, Prentice Hall, 2003.
[2] T. Mitchell, "Machine learning," ISBN 0070428077, McGraw Hill, 1997.
[3] S. Russell and P. Norvig, "Artificial intelligence, a modern approach," Third Edition, Pearson, ISBN-13: 978-0-13-604259-4, ISBN-10: 0-13-604259-7, 2010.
[4] Y. S. Abu-Mostafa, M. Magdon-Lsmail, and H. Lin, "Learning from data," ISBN 10:1-60049-006-9, AMLbook, 2012.
[5] S. L. Hagedorn, "How to define retention: a new look at an old problem," In Alan Seidman (Ed), College student retention: Formula for student Success, Westport, CT: Praeger Publishers, 2005.
[6] R. Alkhasawneh and R. Hobson, "Modeling student retention in science and engineering disciplines using neural networks," IEEE Global Engineering Education Conference (EDUCON), 660-663, 2011.
[7] D. B. Stone, "African-American males in computer science – examining the pipeline for clogs," The School of Engineering and Applied Science of the George Washington University, Thesis, 2008.
[8] E. Frank, "Pruning Decision Trees and Lists," Department of Computer Science, University of Waikato, Thesis, 2000.
[9] C. H. Yu, S. Digangi, A. Jannasch-pennell, and C. Kaprolet, "A data mining approach for identifying predictors of student retention from sophomore to junior year," Journal of Data Science 8, 307-325, 2010.
[10] S. K. Yadav, B. Bharadwaj, and S. Pal, "Mining educational data to predict student's retention: a comparative study," International Journal of Computer Science and Information Security, 10(2), 113-117, 2012.
[11] A. Nandeshwara, T. Menziesb, and A. Nelson, "Learning patterns of university student retention," *Expert Systems with Applications*, *38*(12), 14984–14996, 2011.
[12] S. A. Kumar and M. V. N., "Implication of classification techniques in predicting student's recital**,"** International Journal of Data Mining & Knowledge Management Process (IJDKP), 1(5), 2011.
[13] D. Kabakchieva, "Student performance prediction by using data mining classification algorithms," International Journal of Computer Science and Management Research, 1(4), 686-690, 2012.
[14] S. Lin, "Data mining for student retention management," The Consortium for Computing Sciences in Colleges, 2012.
[15] S. Singh and V. Kumar, "Classification of student's data using data mining techniques for training & placement department in technical education," International Journal of Computer Science and Network (IJCSN) 1(4), 2012.
[16] F. Esposito, D. Malerba and G. Semeraro, "A Comparative Analysis of Methods for Pruning Decision Tree," IEEE, Transaction on Pattern Analysis and Machine Intelligence, Vol 19, No 5, 1997.
[17] D. D. Patil, V. M. Wadhai and J. A. Gokhale, "Evaluation of Decision Tree Pruning Algorithms for Complexity and Classification Accuracy," International Journal of Computer Applications (0975-8887), Vol 11, No 2, 2010.
[18] N. Stanevski and D. Tsvetkov, "Using Support Vector Machines as a Binary Classifier," International Conference on Computer Systems and Technologies – CompSys Tech' 2005.
[19] S. Sembiring, M. Zarlis, D. Hartama, and E. Wani, "Prediction of student academic performance by an application of data mining techniques," International Conference on Management and Artificial Intelligence IPEDR, 6, 2011.

ISBN: 978-988-19252-3-7
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)