ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 9, September 2017

Student's Performance Predictor using Multi Channel Classifier

[1] Himanshu Maniyar, [2] Dr. S. O. Khanna
[1][2] C. U. Shah University, Wadhwan, Gujarat.

Abstract - **Data mining refers to the set of techniques to derive hidden patterns from the large existing data. These patterns can be useful for the analysis and prediction purpose. Education data mining refers to the set of data mining applications in education field. In today's competitive world, it is essential for an institute to predict performance of students. Students could be informed well in advance to focus in a particular direction for the betterment of their academic performances. This research work predicts students' performances in a course, based on their previous performances in related courses. Association rule mining is used to find out a set of related subjects. Students' performances are predicted using various classification algorithms like decision tree, naive bayes etc. The database itself covers each and every piece of information related with student's skills. Classification smoothing algorithm is introduced to select one of the most appropriate classified performance from set of available predictions. This research work has been tested for a database of students of Bachelor of Computer Applications.**

Keywords:--- **Classification, Decision Tree, Naive Bayes, Accuracy, Sensitivity, Smoothing, Prediction**

INTRODUCTION

Data mining refers to the set of techniques to derive hidden patterns from the available data. These hidden patterns are used for efficient analysis and prediction purpose. Education data mining uses such techniques for the analysis of student's and teachers' attitude towards the academia. In the world of digitalization, data mining techniques could help an institute to predict future performances of their students. Even though it is very difficult to predict performance of a student as it depends on a lot of external parameters, this research work tries to find it with utmost accuracy with the help of a database having every small piece of information related with every student's skills. From written communication to the verbal communication, from logic to the leadership, every student is ranked. [1][2].

The main aim of this research work is to predicate student's performance in end semester examination by analyzing student's current activities. Student's performance can be measured with various activities like Homework, Practicals, Quiz, Tests, Viva etc. As each of these categories is having different importance and requires different skills, it is not easy to guess student's end semester examination performance based on analyzing any one of them manually. At the same it is required to consider difficulty levels as well as students interests too[2].

Let's discuss two cases. One student performs home work nicely but he is not performing well in other activities. The reason might be having good resources like books, Internet Access, help from family members or friends or other teachers. Student might be having good skills in finding answers from the books. One student performs good in Quiz, Tests but not in Tests, Vivas. The reason might be having poor communication skills. Student might be interested in practical approach only [2].

So to predicate the performance efficiently, every activity is considered as a decision channel and a Multi-Channel classifier is developed. Multi-Channel Classifier uses previous semester student's performance to develop classifier algorithm which it applies on current semester student's performance in various activities. This paper explains the proposed work with an example of analysis related with four subjects. Subject "C" is offered in semester 1. Subject "Data Structures" is offered in semester 2. Subject "C++" is offered in semester 3. Subject "Java" is offered in semester 4. We want to predict performance of students in "Java" based on their performances in "C", "Data Structures" and "Java". The same concept can be applied for the prediction of other subjects. One such approach is proposed to detect students who are at risk[10]. This research work extends the concept given in [10]. Similar set of subjects are determined from association rule mining with the help of apriori algorithm[3][4][5].

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 9, September 2017

2. DATABASE DESIGN

2.1. Training Database

Training Database is divided into three sections. Every subject needs to be evaluated separately. A table needs to be prepared for each of the subjects with following information. This database is for past students who have passed or failed a specific subject. Results are marked as Grades A,B,C and Fail.

Section 1. Student's information
(name, enrollment number)

Section 2. Student's assessment information
(hoemwork marks, mid sem marks, quiz marks, practical marks, termwork marks, viva marks, theory marks,total marks,result, teacher's_view)

Section 3. Student's skill information
(critical thinking, problem solving, oral communication, written communication, resource finding, sincerity, imagination, leadership)

For example, if we are preparing dataset for "JAVA" then we can consider "C", "Data Structures" and "C++" as past dependent subjects. Performance of a student in "JAVA" may depend on his performances in "C", "Data Structures" and "C++".

2.2. Testing Database

This data is of all students who are studying or going to study specific subject. This data must be having information of all students for whom atleast one component from section 2 (student's assessment information) is pending. So we can predicate performances for remaining components.

Section 2 and section 3 information could be partially available. For example, if we want to predicate end semester performance just after announcing mid semester result, then section 2 can have information upto mid semester result and section 3 has few skills information which have been evaluated upto mid semester exam but not all. If all information is possible then there will be no issue too.

If we want to evaluate performance in the beginning of a semester, there will be no information from section 2. Section 3 information could be copied based on the most recent dependent subject's section 3. For example, if we want to predicate student's performance in "java" – at the beginning of a semester, then we will not have any information of section 2. We will not have any information of section 3. In this case, we will put information of student's "c" and "c++" results in section 4. Now section 3 information could be prepared by coping section 3 information of "c" or "c++". The most suitable information to copy is of most recent subject. If "c" is offered in semester 1 and "c++" is offered in semester 2 then we should copy section 3 information from student's section 3 of "c++".

3. CLASSIFICATION

3.1. Introduction

Classification is a systematic approach to build models from input data. For example, decision tree method, rule-based method, neural networks, support vector machines, and naive Bayes methods are classification methods. Each method prepares a model that best fits the relationship between the attributes and classes of data. The main goal of a classifier is to predict class of a new data based on the analysis of classes of previously available data[6].

3.2. Decision Tree Method

Decision Tree is most widely used classification method. Decision Tree method finds a series of questions in the forms of attribute-value conditions. These conditions are ordered priority wise based on the most appropriateness. The new data is checked across these conditions. All conditions are represented in the form of a tree called decision tree. Interior nodes are the conditions and exterior nodes are the classification classes. The new data is propagated from root to the leaves[7].

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 9, September 2017

3.3. Naive Bayes Method

Naive bayes classification method is based on the bayes theorem of probabilities. The main concept here is to assume independency among the attributes which we are checking of a new data. Naive bayes method calculates conditional and joint probabilities among the available attributed. The main reason of doing so is to find the probability of each of the classes for a newly available data. The newly available data has certain fields available while certain fields unavailable. Naive bayes method calculates conditional probability of each of the classes given partial data. The probability of the class with highest value is the classification value[8].

4. IMPLEMENTATION AND TESTING

4.1. Decision Tree Method

Implementation has been done with R [9]. Decision Tree Method has been implemented for the training data of 166 students. These students have completed examinations of “C”, “Data Structures” , “C++” and “Java”. Based on these results we want to design a model which will predict grade for “Java” given the data of “C”, “Data Structures” , “C++”. The compact version of decision tree is shown in Figure 1. The decision tree can also be represented in the form of conditions as below. S1 to S4 refers to the subjects. In our case, S1=”C”, S2=”Data Structures”, S3=”C++”, S4=”Java”

1) S1OralCommunication <= 1; criterion = 1, statistic = 132.33
  2) S1MidSemExam <= 24; criterion = 1, statistic = 30.607
    3) S1MidSemExam <= 9; criterion = 0.994, statistic = 14.703
      4)* weights = 9
    3) S1MidSemExam > 9
      5)* weights = 56
  2) S1MidSemExam > 24
    6)* weights = 10
1) S1OralCommunication > 1
  7) S1TeacherView <= 2; criterion = 1, statistic = 25.374
    8)* weights = 66
  7) S1TeacherView > 2
    9) S1EndSemExam <= 67; criterion = 0.985, statistic = 12.912
      10)* weights = 15
    9) S1EndSemExam > 67
      11)* weights = 10

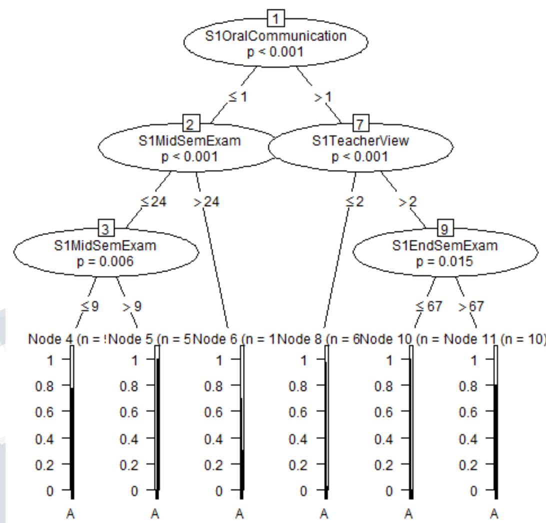

Figure 1. Decision Tree

4.2. Naive Bayes Method

Naive Bayes method is also implemented with the same data of 166 students. The output will be a set of conditional probabilities matrices. A few of such matrices are shown below.$tables

```
$tables$S1Homework
      S1Homework
Y          [,1]      [,2]
  A    9.000000 0.0000000
  B    8.202128 0.7839765
  C    7.338235 0.7041517
  Fail 7.500000 0.7071068

$tables$S1Quiz
      S1Quiz
Y           [,1]      [,2]
  A    10.000000 0.0000000
  B     9.191489 0.7658939
  C     8.367647 0.7310677
  Fail  8.500000 0.7071068
```

IFERP
connecting engineers... developing research

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 9, September 2017

4.3. Confusion Matrix

Decision Tree Method and Naive Bayes Method have been tested across the same set of 166 students to find whether they can predict the result of "Java" accurately or not. These 166 students have already got results of "Java". The actual results are compared with the predicted results to calculate accuracy. Confusion Matrix of both the methods are compared as below.

Parameter	Decision Tree					Naive Bayes				
Confusion Matrix	DTree	A	B	C	Fail	NBayes	A	B	C	Fail
	A	0	0	0	0	A	2	10	0	0
	B	2	94	5	0	B	0	77	3	0
	C	0	0	63	2	C	0	7	65	0
	Fail	0	0	0	0	Fail	0	0	0	2
Accuracy	0.9458					0.8795				
95% CI	(0.8996, 0.9749)					(0.8201, 0.9248)				
Kappa	0.8905					0.7803				

Table 1 – Confusion Matrix

Decision Tree	Class: A	Class: B	Class: C	Class: Fail
Sensitivity	0	1	0.9265	0
Specificity	1	0.9028	0.9796	1
Pos Pred Value	NaN	0.9307	0.9692	NaN
Neg Pred Value	0.98795	1	0.9505	0.98795
Prevalence	0.01205	0.5663	0.4096	0.01205
Detection Rate	0	0.5663	0.3795	0
Detection Prevalence	0	0.6084	0.3916	0
Balanced Accuracy	0.5	0.9514	0.953	0.5

Table 2 - Statistics by Class – Decision Tree

Naïve Bayes	Class: A	Class: B	Class: C	Class: Fail
Sensitivity	1	0.8191	0.9559	1
Specificity	0.93902	0.9583	0.9286	1
Pos Pred Value	0.16667	0.9625	0.9028	1
Neg Pred Value	1	0.8023	0.9681	1
Prevalence	0.01205	0.5663	0.4096	0.01205
Detection Rate	0.01205	0.4639	0.3916	0.01205
Detection Prevalence	0.07229	0.4819	0.4337	0.01205
Balanced Accuracy	0.96951	0.8887	0.9422	1

Table 3 - Statistics by Class – Naive Bayes

4.4. Graphical Representations

Figure 2 plots Actual Result vs Predictions made by decision tree method and naive bayes method. Figure 3 plots Actual Result vs Predictions made by decision tree method and naive bayes method in stacked format.

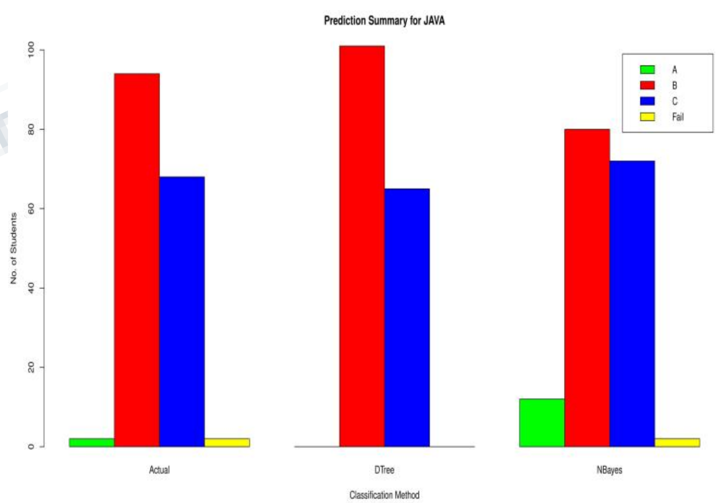

Figure 2 – Group Plot Actual Result vs Predictions

IFERP connecting engineers... developing research

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 9, September 2017

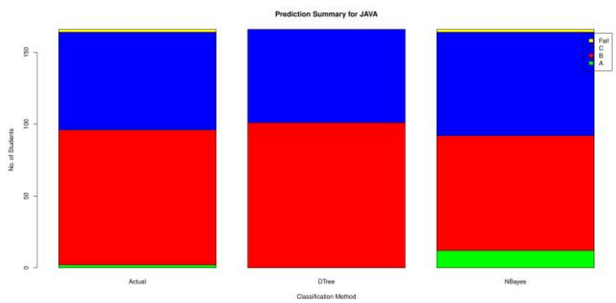

Figure 3 – Stacked Plot Actual Result vs Predictions

5. SMOOTHED CLASSIFICATION

5.1. Algorithm

At the end of the classification, we have two predicted values (Decision Tree's prediction and Naive Bayes' prediction) for each of the students. Classification smoothing algorithm refers to the selection of one of these two values as final prediction.

```
For each student Si
{
Find di (Grade predicted by decision tree) and bi (Grade predicted by naive bayes)
    If (di == bi)
    {
    Grade(Si) = di      // No need of smoothing
        }
        Else
        {
    If (dtree_sensitivity(di) > nbayes_sensitivity(bi))
            {
                Grade(Si) = di
            }
Else if (dtree_sensitivity(di) < nbayes_sensitivity(bi))
            {
                Grade(Si) = bi
            }
            Else
            {
    If(dtree_accuracy > nbayes_accuracy)
                {
                Grade(Si) = di
                }
                Else
                {
                    Grade(Si) = bi
                }
            }
        }
}
```

5.2. Result

At the end of the classification, we have done classification smoothing as per the algorithm given in 5.1. Decision tree method was able to classify with accuracy 0.9458 – 157 correct predictions over 166. Naive bayes method was able to classify with accuracy 0.8795 – 146 correct predictions over 166. Smoothed classifier has accuracy 0.9639 – 160 correct predictions over 166. The other statistics are shown in Table 4 and Table 5.

<table>
<tr><th>Parameter</th><th colspan="5">Smoothed Classification</th></tr>
<tr><td rowspan="5">Confusion Matrix</td><td>SC</td><td>A</td><td>B</td><td>C</td><td>Fail</td></tr>
<tr><td>A</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>B</td><td>1</td><td>94</td><td>5</td><td>0</td></tr>
<tr><td>C</td><td>0</td><td>0</td><td>63</td><td>0</td></tr>
<tr><td>Fail</td><td>0</td><td>0</td><td>0</td><td>2</td></tr>
<tr><td>Accuracy</td><td colspan="5">0.9639</td></tr>
<tr><td>95% CI</td><td colspan="5">(0.923, 0.9866)</td></tr>
<tr><td>Kappa</td><td colspan="5">0.9282</td></tr>
</table>

Table 4 – Smoothed Classification

	Smoothed Classification	Class: A	Class: B	Class: C	Class: Fail
	Sensitivity	0.5	1	0.9265	1
	Specificity	1	0.9167	1	1
	Pos Pred Value	1	0.94	1	1
	Neg Pred Value	0.993939	1	0.9515	1
	Prevalence	0.012048	0.5663	0.4096	0.01205
	Detection Rate	0.006024	0.5663	0.3795	0.01205
	Detection Prevalence	0.006024	0.6024	0.3795	0.01205
	Balanced Accuracy	0.75	0.9583	0.9632	1

Table 5 - Statistics by Class – Naive Bayes

6. CONCLUSION

Education data mining is one of the most recent trends of data mining applications. For every institute, it is beneficial to predict students' performances to guide them accordingly. It is very difficult to predict results as a lot of parameters may affect. This research work has tried to cover all skills which may affect the result. At the same time, teacher's view is also used as one of the attributes to include human prediction component. The database is designed in a such a way that it can be customized easily for new set of subjects. Decision tree and naïve bayes methods are selected. Our data has four classes: A,B,C, Fail. Majority of the data lies within classes B and C. Minority of the data lies within classes A and Fail. It could be seen that decision tree method fails to classify data for the minority classes naïve bayes method classifies data for minority classes very well. Classification smoothing is introduce to select best out of the predictions. Sensitivity of a class in a method and accuracy of a method are used to select most appropriate prediction. It has been found that decision method is better than naïve bayes method. Classification smoothing makes the overall prediction better.

REFERENCES

[1] Xingquan Zhu, Ian Davidson, "Knowledge Discovery and Data Mining: Challenges and Realities", ISBN 978- 1-59904-252, Hershey, New York, 2007.

[2] Koprinska, Irena, Joshua Stretton, and Kalina Yacef. "Students at Risk: Detection and Remediation." EDM.2015.

[3] Pelánek, Radek. "Measuring Similarity of Educational Items Using Data on Learners' Performance."

[4] Yanfei Zhou, Wanggen Wan, Junwei Liu, Long Cai, "Mining Association Rules Based on an Improved Apriori Algorithm", 978- 1-4244-585 8- 5/10/$26.00 ©2010 IEEE.

[5] Huan Wu, Zhigang Lu, Lin Pan, Rongsheng Xu, Wenbao Jiang, "An Improved Apriori-based Algorithm for Association Rules Mining" Sixth International Conference on Fuzzy Systems and Knowledge Discovery, 978-0-7695-3735-1/09 $25.00 © 2009 IEEE DOI10.1109/FSKD.2009.193.

[6] Venkatadri.M and Lokanatha C. Reddy ,"A comparative study on decision tree classification algorithm in data mining" , International Journal Of Computer Applications In Engineering ,Technology And Sciences (IJCAETS), Vol.- 2 ,no.- 2 , pp. 24- 29 , Sept 2010.

[7] Safavian, S. Rasoul, and David Landgrebe. "A survey of decision tree classifier methodology." IEEE transactions on systems, man, and cybernetics21.3 (1991): 660-674.

[8] McCallum, Andrew, and Kamal Nigam. "A comparison of event models for naive bayes text classification." AAAI-98 workshop on learning for text categorization. Vol. 752. 1998.

[9] Team, R. Core. "R language definition." Vienna, Austria: R foundation for statistical computing (2000).