Automated System for Student's Intellectual Performance Prediction

Deepshikha Chaturvedi, Shashikant Radke, Malay Shah, Prerna Rao, Hitarth Dani
Computer Engineering Department
Shah and Anchor Kutchhi Engineering College, Mumbai, Maharashtra, India
deepshikha.chaturvedi@sakec.ac.in, *shashikant.radke@sakec.ac.in*,
malayshah099@gmail.com, *prernarao@gmail.com*,
hitarthdani09@gmail.com

Abstract

In the academic world, in order to ensure that quality education is provided to the students. To prevent or take precautions against dropouts or student failures and to improve the quality of managerial decisions, predicting a student's intellectual performance is extremely vital not only for the higher education management bodies but also for the students themselves. In our system, different categories of metrics for predictive analytics that would influence the performance of First year engineering students has been collected and used in what-if scenario predictions for their Second Semester exam grading. Various Data mining algorithms and techniques have been studied and the best one which will be most suitable for this particular classification problem has been identified and is used for Predictive Analysis computations. Cloud services are used for storing large voluminous Student Data. Thus, using this student performance model, appropriate and timely warning can be given to students if they are at a risk, educational institutions would also be able to provide them with better additional training, moreover this system would also help in identifying potentially meritorious students and thereby encourage them for academic scholarships.

Keywords: *Android, classification, cloud, data mining, decision tree, educational data mining (EDM), J-48, prediction, student performance*.

I. INTRODICTION

The data from the educational field needs to be explored in order to get better understanding of the education system in holistic is called Educational Data Mining (EDM). EDM emerges in such a way to design models, tasks, methods, and algorithms for exploring data from educational settings. EDM pursues to find out patterns and make predictions that characterize learners' behaviors and achievements, domain knowledge content, assessments and educational functionalities.[1]

A study done by Jawaharlal Nehru Technological University (JNTU) shows that engineering drop outs rates are on a steep increase. Time of India, in one of its reports stated that about 3% to 4% of students, who join engineering colleges, dropout every year and that this figure would add up to thousands of dropouts on an average in reality. Using Educational Data Mining, patterns in the student data can be found out and predictions can be made that would characterize learners' behaviors and help in assessing their intellectual performance.

Mining in the educational domain helps the students to track their strengths and shortcomings and subsequently work and improve on the same. It also helps students to achieve their goal grades. It also allows the educators to assist students enabling them to focus on students' weaknesses, enables early detection of students who need more attention and consequently paves a way of provision for appropriate counselling. Other than this, it also helps the college management and faculty in identifying potentially meritorious students to encourage them for academic scholarships.

The main objective of this work is to obtain student data and analyze it using data mining techniques in order to predict student performance and thereby help both the students as well as educators.

II. LITERATURE SURVEY

A. Classification

The objective of our work is to predict the performance of the student (semester grade). This value is discrete and hence is classification task.

Classification (also known as classification trees or decision trees) is a data mining algorithm that creates a step-by-step guide for how to determine the output of a new data instance. Here, each node in the tree represents a spot where a decision must be made based on the input, and we move to the next node and the next until a leaf is reached that tells us the predicted output[5].

The classification model summarizes the relationships, and they can then be applied to a data set in which the class assignments are unknown. Algorithms like C4.5, ID3, k-nearest neighbor classifier, Naive Bayes, SVM etc. are used for classification.

Bayes Network

Bayesian Belief Network algorithm is almost same as Naive Bayes classifier with the difference that this algorithm does not show dependency between attributes. They are also called as Probabilistic Networks. This algorithm is Condition independent. However, practically there will always be some dependency between attributes. Bayes Network uses Joint Probability Concept for this purpose. Bayesian Belief Network can be trained and used for classification however doing full Bayesian Learning is extremely computationally expensive. Moreover, Bayesian network models are hard to interpret.

Naive Bayes

Naive Bayesian Classification is based on Bayes Theorem. Bayesian classifiers are statistical classifiers. Naive Bayesian Classification is referred as Naive because it makes the. assumption that each of the inputs are independent of each other. This is an assumption which rarely holds true in real life scenarios.

Use of Naive Bayes Classifier would cause inaccurate results in our work as for our data set, final Semester Grades depends upon many attributes unlike this algorithm which proceeds with the assumption that each of the attributes are independent of each other.

Random forest

Random forest algorithm works as a large collection of correlated decision trees. Essentially, it has a lot of Decision trees and it uses each of them to classify instances. It is said to be a technique based on the bagging technique wherein on using different combinations of learning models for classification, accuracy increases. However, it poses a major limitation in our work. Since we have limited data this algorithm will not give appropriate classification.

Decision Tree

A decision tree is a predictive machine-learning model which examines various attribute values of the available data and decides the target value (dependent variable) of a new sample. Attributes are denoted by internal nodes of the decision tree whereas the branches between the nodes indicate the possible values that these attributes can have in the observed samples. The terminal nodes give the classification of the final value of the dependent variable.

In our work, the decision tree classifies student information based on various academic and personal attribute. This tree will have 2nd semester grades (O, A, B, C, D, E, F) as the terminal nodes. Decision tree helps us store this data with the help of which our further calculation for training and prediction would occur.

Creating a decision tree using modified J48

J48 classifier is a simple C4.5 decision tree for classification. In this technique, in order to model the classification process, a tree is constructed. Once the tree is generated, it is applied to each tuple in the database and this results in classification for that tuple.

J48 classifier is an extension of ID3. Furthermore, we modified this algorithm in such a way that missing values are accounted for and proper branching rules are derived for the same. Hence, we used modified J48 to classify our Student Dataset.

B. WEKA Tool

Weka is a machine learning software suite developed in Java. It provides facilities for all the steps involved in solving a machine learning problem- data conversion, pre-processing techniques, classification, categorization and visualization. Weka commands can also be carried out via the command line and the Weka API can be called from Java as well as is the case in this work. [1]

Weka GUI provides many options to work with data. The WEKA GUI enables interaction with the data files and has provisions for visualizing results. This makes it very easy to understand the flow of data, and envision the outcomes. However, the GUI component does not work well when dealing with very large data.

WEKA API

In our work, we embed WEKA API, in order to carry out automated server-side data-mining task of retrieving decision tree rules. Basically, we use WEKA API to train, test and classify our data programmatically.

We are using Weka APIs as they can be easily incorporated into our work and is lightweight in terms of memory. It also provides much more scope for dealing with very large files.

C. AWS (Amazon web services) Server

The first step is to set up an Amazon Web Services account. This is freely available and requires only a valid email id and a method of payment such a credit card. Once logged, in, we have to navigate to the AWS Management Console and then to the EC2 (Elastic Cloud Compute) module.

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon Web Services (AWS), By allowing users to rent virtual computers on which computer applications can be run.

D.MongoDB

MongoDB is a free and open-source cross-platform document-oriented database program that provides high performance, high availability, and easy scalability. It works on concept of collection and document. Classified as a NoSQL database program, MongoDB uses JSON-like documents with schemas. MongoDB is developed by MongoDB Inc., and is published under a combination of the GNU Affero General Public License and the Apache License. [10]

Database is a physical container for collections. Each database gets its own set of files on the file system. A single MongoDB server typically has multiple databases. Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database. Collections do not enforce a schema. Documents within a collection can have different fields. Typically, all documents in a collection are of similar or related purpose. A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data. Also, in MongoDB, there is no concept of relationship.

III. Proposed System

After appropriate collection of student data, we predict performance of 1st year Engineering students in the following 2nd Semester Subjects, Applied Mathematics 2, Applied Physics 2, Applied Chemistry 2 and further identify possible result grade wise based on their previous academic records and other parameters. This is based on the assumption that the college maintains the same academic environment throughout the semester and the result processing pattern does not change. Following the assumption of technical analysis & the patterns that exist in student attributes, it is possible to discover these patterns using data mining techniques. Once these patterns have been discovered, a range of scores can be predicted.

IV. System Design Implementation

A. System Blocks

The System consist of *Android Application, REST API As Middleware and Database on cloud.*

1) *Android Application*

Android Application is used as User Interface for users to Login; fill required details; to get the predicted results of Applied Mathematics 2, Applied Physics 2 and Applied Chemistry 2 and to view the entered details and Predicted results later on. Android

Application is used for processing in order to fetch the user data and to find the predicted Grade of each subject using Modified J48 Algorithm. Furthermore, Android Application is used for user Authentication. It does so by fetching the Credentials (Email_id and Reg. No) entered by user and verifies entered Credentials with database stored in cloud. It thereby grants different permissions based on different types of users.

2) *REST API As Middleware*

REST API is used as middleware to transfer data between Android Application and database on Cloud. It provides an additional layer between the application and the database on cloud thereby providing a layer of added security.

3) *Databases on cloud*

There are two databases stored on mLab cloud viz. Registered User's database and Student's Detail Database. Registered User's database is to be maintained by Mentor. It will consist of Students Credentials (Email id and Registration Number). In Student's Detail Database all academic details of the student will be stored along with other attributes. This database will also store the Predicted Results of Each Subject. This database will be providing the stored details to Student as well as Mentor in Android Application.

B. *System Phases*

Phase 1

The first phase includes data collection and pre-processing of the obtained data. Our Training Dataset comprises of Student Data of 2nd Year, 3rd Year and 4th Year Engineering Students. This is collected using Google forms and processed locally using Microsoft Excel operations. Any redundant information is removed and computations or modifications required is done. For ex. Pointer to Percentage conversion for the attribute of 10th Boards is done for CBSC students.

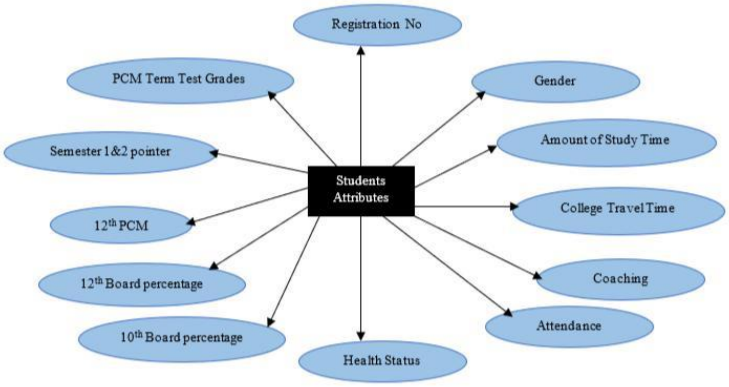


Figure 1: Attributes taken into consideration while collecting data

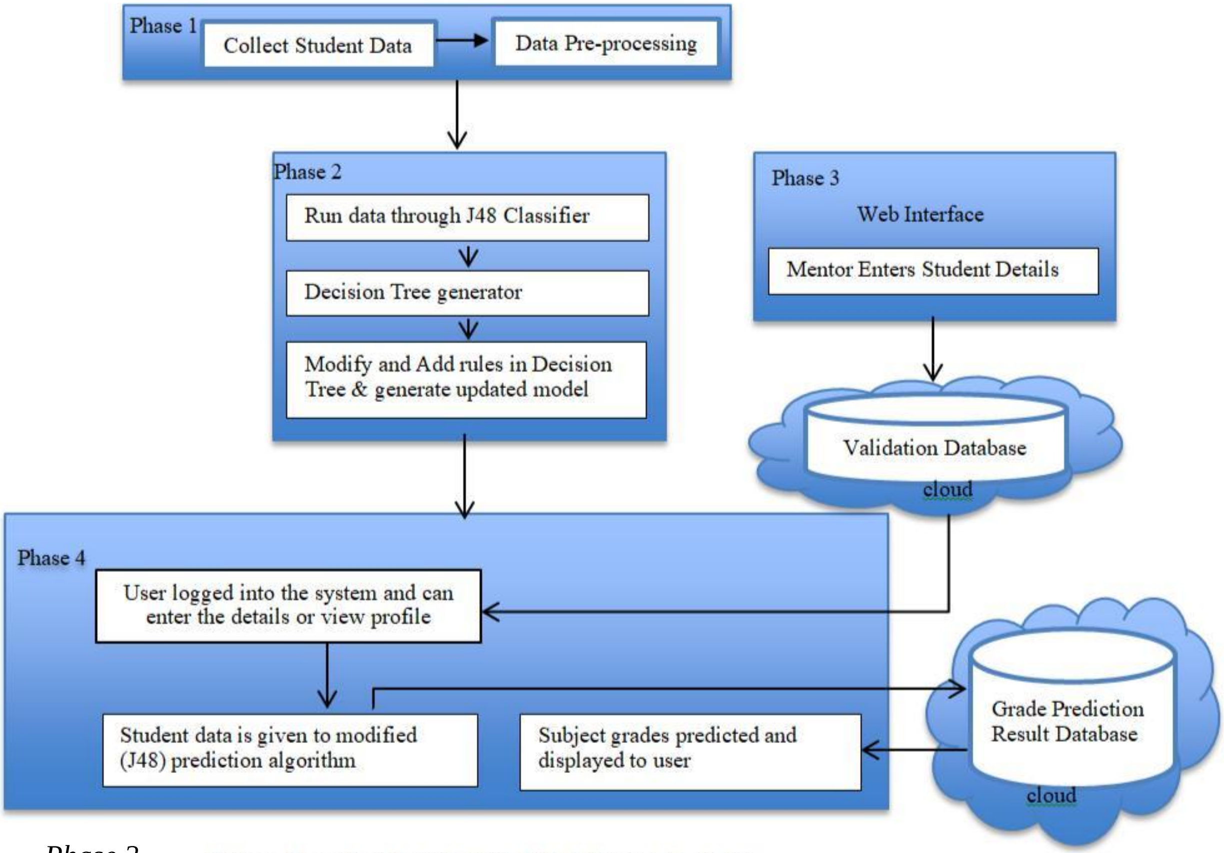

Phase 1
Collect Student Data
Data Pre-processing
Phase 2
Run data through J48 Classifier
Decision Tree generator
Modify and Add rules in Decision Tree & generate updated model
Phase 3
Web Interface
Mentor Enters Student Details
Validation Database
cloud
Phase 4
User logged into the system and can enter the details or view profile
Student data is given to modified (J48) prediction algorithm
Subject grades predicted and displayed to user
Grade Prediction Result Database
cloud

Figure 2. *System Flow diagram for Students Performance Prediction*

Phase 2

In this phase, the obtained cleaned training data from Phase 1 is used to generate prediction model using J48 Algorithm. This will be done locally. Next, the obtained decision tree is studied and based on the data set, and we add new rules to compensate for missing values in the data. A new model is generated using this Modified J48.

Phase 3

In this phase, the Mentor will register students by entering student's Email-id and Reg. No on Web Application Interface. This data will be stored in the MongoDB database in the cloud.

Phase 4

In this phase, the Android Application is deployed. In Our system there are two types of users, Mentor and Student. Student will enter his/her Email-id and Reg. No to login. If the student is logging in for the first time then he'll have to fill a form which consist of his/her academic details and some other psychometric attributes. After submitting all the details, our system will show the predicted Grades of AM-2, AP-2 and AC-2 to the user. This will be preceded by sending of the student details along with

the predicted grades to the database created in Phase 2. When the student is logging in again into the system at that time our App will simply display the predicted Grades and all the Details entered by that student by fetching the details from the database stored in cloud. When Mentor will login into the system, he will be able to see the predicted grades of all the students who have filled the form along with essential details of the student.

C. *Algorithm of J48 Decision Tree Classifier*

a) Calculating Original Entropy of the Dataset

Given a set of examples D Original entropy of the dataset is given as follows where C is the set of desired class.

$$[\,] = -\sum_{=1}^{|\,|} (\,)\ {}_2(\,)$$

b) Calculating Entropy of an Attribute Ai

If the root of the current tree is made as attribute Ai, with v values, D will be partitioned into v subsets D1, D2,…,Dv. The expected entropy of Ai is used as the current root and is given as:-

$$[\,] = -\sum_{=1} \frac{}{} [\,]$$

c) Calculating the Information gain

Information gained by selecting attribute Ai to branch or to partition the data is given by the difference of prior entropy and the entropy of selected branch as:

$$(\,,\quad) = [\,] - \quad [\,]$$

d) Attribute with the highest gain to split the current tree is chosen.

e) Create child nodes based on split.

f) Recurse on each child using child data until a stopping criterion is reached, if all examples have same class or the amount of data is too small or the tree is too large.

D. Database Schema

1) Students (Reg_no: integer, Stud_name: string, Stud_pass: string, SEmail id: string)

2) Students_Dataset (Reg_no: integer, Stud_name: string, SEmail id: string, Gender: string, Div: string, SSC: integer, HSC: integer, UT_Grade: string, External_Coaching: string, Travel_time: integer, Attendence: integer, health_status: string, Sem1_grade: string)

3) Predicted_Result (Reg_no: integer, Stud_name: string, SEmail id: string, Gender: string, Div: string, SSC: integer, HSC: integer, UT_Grade: string, External_Coaching: string, Travel_time: integer, Attendence: integer, health_status: string, Sem1_grade: string, Predicted grade: integer)

4) Mentor-Mentee (MEmail id: string, Mentor_pass: string)

V. Results and Analysis

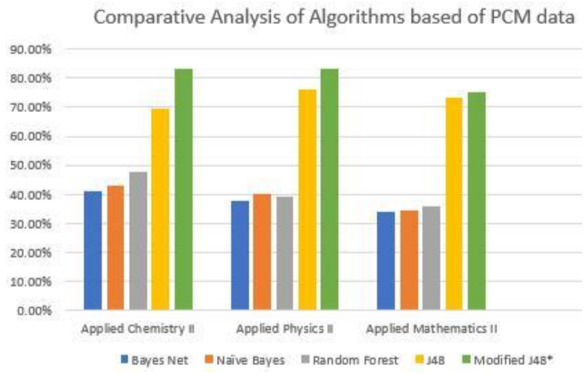
One of the objectives was to find out the most suitable data mining classification algorithm for our classification problem. Several different algorithms like Bayes Network, Naive Bayes, Random Forest and J48 were applied on the data set for building the classification model using Weka Explorer.

In order to check the accuracy of various classification algorithms, we collected a data set and we make use of the WEKA Explorer tool. So we perform classification on the data set by running the data set through the WEKA Explorer and applying Bayes Net, Naive Bayes, Random Forest and J48 Algorithm on it.

In order to calculate the accuracy for our proposed algorithm of modified J48, the same data set is considered and an approximated estimate of the accuracy is calculated. The results for the performance of various classification algorithms against the subjects are summarized and presented in the table 1.

Table I. - Classification algorithms performance comparison *suggests approximated accuracy based on dataset obtained.

Performance Comparison		Subjects		
		Applied Chemistry II	Applied Physics II	Applied Mathematics II
Algorithms	Bayes Net	40.83%	37.76%	34.02%
	Naïve Bayes	42.92%	40.25%	34.44%
	Random Forest	47.5%	39.00%	35.68%
	J48	69.58%	76.35%	73.44%
	Modified J48*	83.33%	83.33%	75%



The achieved results reveal that our proposed algorithm performs the best with highest overall accuracy against all the other existing algorithms. Our proposed algorithm which is a modification of J48 predicts grades of students with 83.33% accuracy for the subject of Applied Chemistry II and Applied Physics II followed by 75% accuracy in Applied Mathematics II.

Our proposed algorithm followed by the J48 decision tree classifier are most reliable because they perform with the highest accuracy for all the subjects. The predictions for grades of Applied Physics II and Applied Chemistry II is the most precise using Modified J48. Prediction done using Bayes Net, Naive Bayes and Random Forest performed with prediction accuracy around 34-48%. Bayesian Network classifiers are less accurate than the others with an average accuracy of 37.583%.

From our experimental analysis, we can conclude that classification using Bayesian Networks will not give highly accurate results. One of the reasons for this could be presence of cyclic relationships in the data. We can also infer that Naive Bayes approach is also not suitable for this particular data set. The major reason could be the fact that Naive Bayes assumes that attributes are independent of each other. However, in an educational setup, certain parameters may contribute to affecting the way a student studies and thereby have an effect on his grades. Random Forest approach tends to work better on larger data sets which is not a case in our project. Hence, even it doesn’t fare that well.

On the other hand, we can see that both the decision tree-based classifiers are extremely well. Thus, we can see that decision trees are powerful tools for classification in an educational data mining setting like ours. Moreover, they can be easily understood and interpreted by users as they represent rules.

VI.Conclusion

With increase in low grades among many 1st year Engineering students, a lot of them even end up dropping out of the course. In our project, we successfully integrated data mining in the educational environment to combat the increasing dropout rates by building an android application in order to actively predict how a student will perform in the subjects of Applied Mathematics 2, Applied Physics 2 and Applied Chemistry 2 so that they can be intimated and measures can be undertaken for their betterment. A data set of 250 students containing their academic records as well as other psychometric attributes were collected for this purpose. Modified J48 algorithm was deployed in the android application after experimentally analyzing various classification algorithms. The modified J48 algorithm was used to increase the quality of the data mining procedure as it yielded the best approximated prediction accuracy of 80.5% among all the other experimented algorithms for the three classes viz. Applied Chemistry 2, Applied Physics 2 and Applied Mathematics 2. We used Amazon Web Service’s Elastic Cloud Compute to deploy the application with REST API as the middleware. All of the data was successfully stored on mLab, a large cloud MongoDB service which made data retrieval to the application hassle free and smooth. Thus the objectives were met and the project was an overall success.

For future scope, recommendations can be provided to the College management and authorities, concerning with the availability and sufficiency of student data and the data collection process can be further improved. Further, improved data mining techniques on an expanded data set with more distinctive attributes can be undertaken to get more accurate results. Moreover, the underlying theme of prediction of grades can be extended to other subjects in the course as well enabling a wider application of data mining.

REFERENCE

[1] K. P. Shaleena and Shaiju Paul, "Data mining techniques for predicting student performance", *IEEE International Conference on Engineering and Technology (ICETECH)*, pp. 1-3, 2015

[2] Oshin Mundada, “Mining Educational Data from Student’s Management System”, International Journal of Advanced Research in Computer Science, Vol 7, no. 3, 2016

[3] Chitra Jalota, Rashmi Agrawal, “Analysis of Educational Data Mining using Classification”, *International Conference on Machine Learning Big Data Cloud and Pa rallel Computing (COMITCon)*, pp. 243-247, 2019.

[4] C. L. Sa, D. H. b. Abang Ibrahim, E. DahlianaHossain and M. bin Hossin, "Student performance analysis system (SPAS)," Information and Communication Technology for The Muslim World (ICT4M), 2014 The 5th International Conference on, Kuching, 2014, pp. 1-6.

[5] Samrat Singh, Dr. Vikesh Kumar, “Performance Analysis of Engineering Students for Recruitment Using Classification Data Mining Techniques”, IJCSET Vol 3, no 2, Pg31-37, 2013

[6] Dorina Kabakchieva, “Predicting Student Performance by Using Data Mining methods for Classification”, Bulgarian Academy of sciences, Cybernetics and Information Technologies, Vol 13, no. 1, Sofia University, 2013.

[7] P. K.G., S. S. and S. Asokan, "Enhanced Performance of Engineering Students through REA: A Comparative Analysis”, *International Conference on Advances in Computing and Communications*, Kerala, 2012, pp. 178-181. 2012

[8] R. R. Kabra "Performance Prediction of Engineering Students using Decision Trees", International Journal of Computer Applications(0975- 8887) Volume 36- No.11,December 2011

[9] Crist´obal Romero, Sebasti´an Ventura, “Educational Data Mining: A Review of the State of the Art”, IEEE Transactions on Systems, Man, And Cybernetics—Part C: Applications And Reviews, Vol. 40, Issue. 6, 2010.

[10] Tjen-Sien Lim, Wei-Yin Loh and Yu-Shan Shih, "A comparison of prediction accuracy complexity and training time of thirty-three old and new classification algorithms", *Machine learning*, vol. 40.3, pp. 203-228, 2000