IMPLEMENTATION OF DATA MINING ALGORITHMS FOR STUDENTS DATA USING RAPIDMINER: AN OVERVIEW

[1] *Miss. Sapna Tamboli,* [2] *Prof. Bhushan Chaudhari*

[1] Student, Department of Computer Science, Sandip University, Nasik, India
[2] Professor, Department of Computer Science, Sandip University, Nasik, India

Abstract: It is necessary to review and analyze educational data particularly students' performance. educational data mining (EDM) is that the field of study involved with mining educational data to seek out attention-grabbing patterns and data in educational organizations. though data mining has been successfully enforced within the business world for a few time currently, its use in higher education remains comparatively new, i.e. its use is meant for identification and extraction of new and potentially valuable knowledge from the data. exploitation data mining the aim was to develop a model which may derive the conclusion on students' academic success. This study is equally involved with this subject, specifically, the students' performance. This study explores multiple factors in theory assumed to have an effect on students' performance in higher education, and finds a qualitative model that best classifies and predicts the students' performance supported connected personal and social factors.

Keywords—Data Mining; Education; Students; Performance; Patterns, Educational Data Mining, Rapid Miner.

Introduction

Educational data mining (EDM) could be a new trend within the data processing and data Discovery in Databases (KDD) field that focuses in mining helpful patterns and discovering helpful data from the academic data systems, such as, admissions systems, registration systems, course management systems (Moodle, blackboard, etc…), and the other systems coping with students at totally different levels of education, from colleges, to high schools and universities. Researchers during this field target discovering helpful data either to assist the educational institutes manage their students higher, or to assist students to manage their education and deliverables higher and enhance their performance. Analyzing students' data and information to classify students, or to create decision trees or association rules, to make better decisions or to enhance student's performance is an interesting field of research, which mainly focuses on analyzing and understanding students' educational data that indicates their educational performance, and generates specific rules, classifications, and predictions to help students in their future educational performance. Classification is the most familiar and most effective data mining technique used to classify and predict values. Educational Data Mining (EDM) is no exception of this fact, hence, it was used in this research paper to analyze collected students' information through a survey, and provide classifications based on the collected data to predict and classify students' performance in their upcoming semester. The objective of this study is to identify relations between students' personal and social factors, and their academic performance. This newly discovered knowledge can help students as well as instructors in carrying out better enhanced educational quality, by identifying possible underperformers at the beginning of the semester/year, and apply more attention to them in order to help them in their education process and get better marks. In fact, not only underperformers can benefit from this research, but also possible well performers can benefit from this study by employing more efforts to conduct better projects and research through having more help and attention from their instructors. There are multiple different classification methods and techniques used in Knowledge Discovery and data mining. Every method or technique has its advantages and disadvantages. Thus, this paper uses multiple classification methods to confirm and verify the results with multiple classifiers. In the end, the best result could be selected in terms of accuracy and precision.

Literature Survey

The approaches followed by every data mining technique are different. Researchers are using different data mining techniques for the diagnosis of many diseases. Some of the classification techniques are as under:

Baradwaj and Pal [1] conducted a research on a group of 50 students enrolled in a specific course program across a period of 4 years (2007-2010), with multiple performance indicators, including “Previous Semester Marks”, “Class Test Grades”, “Seminar Performance”, “Assignments”, “General Proficiency”, “Attendance”, “Lab Work”, and “End Semester Marks”. They used ID3 decision tree algorithm to finally construct a decision tree, and if-then rules which will eventually help the instructors as well as the students to better understand and predict students’ performance at the end of the semester. Furthermore, they defined their objective of this study as: “This study will also work to identify those students which needed special attention to reduce fail ration and taking appropriate action for the next semester examination” [1]. Baradwaj and Pal [1] selected ID3 decision tree as their data mining technique to analyze the students’ performance in the selected course program; because it is a “simple” decision tree learning algorithm.

Abeer and Elaraby [2] conducted a similar research that mainly focuses on generating classification rules and predicting students’ performance in a selected course program based on previously recorded students’ behavior and activities. Abeer and Elaraby [2] processed and analysed previously enrolled students’ data in a specific course program across 6 years (2005–10), with multiple attributes collected from the university database. As a result, this study was able to predict, to a certain extent, the students’ final grades in the selected course program, as well as, “help the student's to improve the student's performance, to identify those students which needed special attention to reduce failing ration and taking appropriate action at right time” [2]

Pandey and Pal [3] conducted a data mining research using Naïve Bayes classification to analyse, classify, and predict students as performers or underperformers. Naïve Bayes classification is a simple probability classification technique, which assumes that all given attributes in a dataset is independent from each other, hence the name “Naïve”. Pandey and Pal [3] conducted this research on a sample data of students enrolled in a Post Graduate Diploma in Computer Applications (PGDCA) in Dr. R. M. L. Awadh University, Faizabad, India. The research was able to classify and predict to a certain extent the students’ grades in their upcoming year, based on their grades in the previous year. Their findings can be employed to help students in their future education in many ways.

Yadav, Bhardwaj, and Pal [5] conducted a comparative research to test multiple decision tree algorithms on an educational dataset to classify the educational performance of students. The study mainly focuses on selecting the best decision tree algorithm from among mostly used decision tree algorithms, and provide a benchmark to each one of them. Yadav, Bhardwaj, and Pal [5] found out that the CART (Classification and Regression Tree) decision tree classification method worked better on the tested dataset, which was selected based on the produced accuracy and precision using 10-fold cross validations. This study presented a good practice of identifying the best classification algorithm technique for a selected dataset; that is by testing multiple algorithms and techniques before deciding which one will eventually work better for the dataset in hand. Hence, it is highly advisable to test the dataset with multiple classifiers first, then choose the most accurate and precise one in order to decide the best classification method for any dataset.

A. C4.5 algorithm

This algorithm is one of the types of decision tree that was introduced after upgrading the ID3 algorithm. This algorithm can classify the records with noisy and continuous amplitude. When the records are with discrete amplitude, this algorithm operates like ID3 algorithm but when the data amplitude is continuous, it will consider a threshold for all selectable modes and an effective standard is assessed for the threshold and then, the threshold with the highest rate is chosen as the decision index of that node [6 and 7].

Bhardwaj and Pal [8] conducted a significant data mining research using the Naïve Bayes classification method, on a group of BCA students (Bachelor of Computer Applications) in Dr. R. M. L. Awadh University, Faizabad, India, who appeared for the final examination in 2010. A questionnaire was conducted and collected from each student before the final examination, which had multiple personal, social, and psychological questions that was used in the study to identify relations between these factors and the student’s performance and grades.

Bhardwaj and Pal [8] identified their main objectives of this study as: "(a) Generation of a data source of predictive variables; (b) Identification of different factors, which effects a student's learning behavior and performance during academic career; (c) Construction of a prediction model using classification data mining techniques on the basis of identified predictive variables; and (d) Validation of the developed model for higher education students studying in Indian Universities or Institutions" [8]. They found that the most influencing factor for student's performance is his grade in senior secondary school, which tells us, that those students who performed well in their secondary school, will definitely perform well in their Bachelors study. Furthermore, it was found that the living location, medium of teaching, mother's qualification, student other habits, family annual income, and student family status, all of which, highly contribute in the students' educational performance, thus, it can predict a student's grade or generally his/her performance if basic personal and social knowledge was collected about him/her

B. SVM algorithm

Support Vector Machine(SVM) algorithm is a supervised data mining algorithm in which we plot each data item as a point in n-dimensional space (where n is the no. of features we have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well. C. KNN algorithm K-Nearest Neighbor is an algorithm which is based on similarity with other items. The items which are similar to each other are called neighbors. Once a new item is found, its distance from other items in the model is calculated. This classification partitions the item to the nearest neighbor which is also the most similar one; so places the item in a group that includes the nearest neighbors [8].

Mathematical Formulation: Dual. It is computationally simpler to solve the dual quadratic programming problem. To obtain the dual, take positive Lagrange multipliers α_i multiplied by each constraint, and subtract from the objective function:

$$L_P = \frac{1}{2}\langle w,w\rangle - \sum_i \alpha_i \left(y_i \left(\langle w,x_i \rangle + b \right) - 1 \right),$$

where you look for a stationary point of L_P over w and b. Setting the gradient of L_P to 0, you get

$$w = \sum_i \alpha_i y_i x_i$$

$$0 = \sum_i \alpha_i y_i.$$

Substituting into L_P, you get the dual L_D:

$$L_D = \sum_i \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle,$$

which you maximize over $\alpha_i \geq 0$. In general, many α_i are 0 at the maximum. The nonzero α_i in the solution to the dual problem define the hyperplane, as seen in which gives w as the sum of $\alpha_i y_i x_i$. The data points x_i corresponding to nonzero α_i are the *support vectors*.

D. Neural Network

Artifical Neural Network is inspired from the brain that is considered as a data processing system. In this algorithm, many microprocessors are responsible for data processing and they are acting as an interconnected and parallel network with each other to solve a problem. By using programming science in this network, a data structure is designed that can act as a neuron and this data structure is called neuron. By setting a network between the neurons and applying a learning algorithm, the network is trained. In this Neural Network, neurons are divided into two enable (NO or 1) or disable (OFF or 0) modes and each edge (synapses or connections between nodes) has a weight. Edges with positive weight, stimulate or enable the next disable nodes and edges with negative weights, disable or inhibit the next connected [9].

The formal neuron has *n***,** generally real, inputs x_1, x_2… x_n that model the signals coming from dendrites. The inputs are labeled with the corresponding, generally real, synaptic **weights** w_1, w_2… w_n that measure their permeabilities. According to the neurophysiological motivation, some of these synaptic weights may be negative to express their inhibitory character.

Then, the weighted sum of input values represents the **excitation level** of the neuron:

$$\xi = \sum_{i=1}^{n} w_i x_i$$

The value of excitation level □, after reaching the threshold *h*, induces the output *y* (state) of the neuron which models the electric impulse generated by axon. The non-linear grow of output value y = □(□) after the threshold excitation level *h* is achieved, is determined by the **activation** (transfer, squashing) **function** □. The simplest type of activation function is the *hard limiter,* which is of the following form:

$$\sigma(\xi) = \begin{cases} 1 \; if \; \xi \geq h \\ 0 \; if \; \xi < h \end{cases}$$

By a formal manipulation it can be achieved that the function □ has zero threshold and the actual threshold with the opposite sign is understood as a further weight, *bias* $w_0 = -h$ of additional formal input $x_0=1$ with constant unit value. Then, the mathematical formulation of neuron function is given by the following expression:

$$y = \sigma(\xi) = \begin{cases} 1 \; if \; \xi \geq 0 \\ 0 \; if \; \xi < 0 \end{cases} \; where \; \xi = \sum_{i=0}^{n} w_i x_i$$

E. Naïve Bayes Naïve

Bayes classifier is based on Bayes theorem. This classifier uses conditional independence in which attribute value is independent of the values of other attributes. The Bayes theorem is as follows:

Let X={x1, x2,……,xn} be a set of n attributes.

In Bayesian, X is considered as evidence and H be some hypothesis means, the data of X belongs to specific class C.

We have to determine P (H|X), the probability that the hypothesis H holds given evidence i.e. data sample X. According to Bayes theorem the

P (H|X) is expressed as P(H|X) = P(X|H) P(H) / P(X)

System Overview

Before we begin with the technical part, it would be helpful to clarify some terms first. RapidMiner uses terminology from the area of machine learning. A typical goal of the latter is, based on a series of observations for which a certain target value is known, to make forecasts for observations where this target value is not known. We refer to each observation as an example. Each example has several attributes, usually numerical values or categorical values such as age or sex. One of these attributes is the target value which our analysis relates to. This attribute is often called a label.

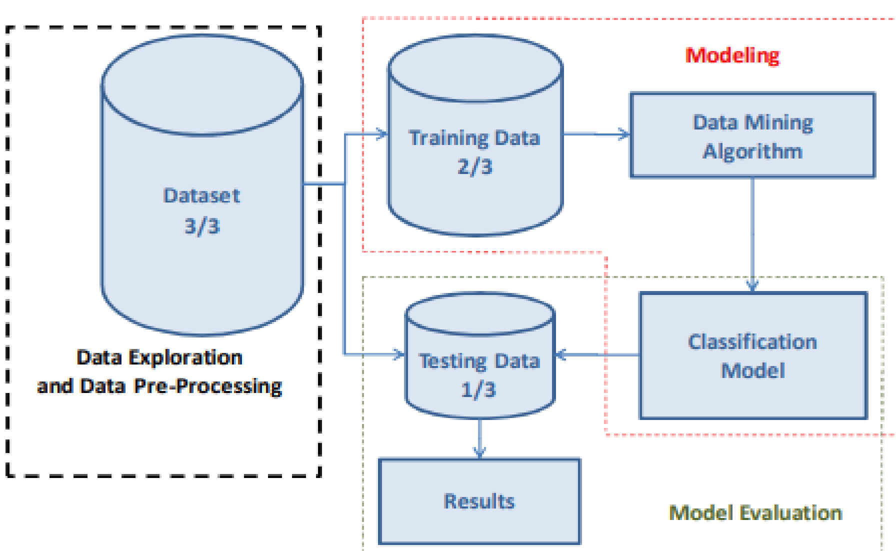

Figure 1 : Proposed System Module

All examples together form an example set. If you write all examples with their attributes one below the other, you will get nothing other than a table. We could therefore say ”table“ instead of ” example set“, ” row“ instead of ” example“ and ” column“ instead of ” attribute”“. It is helpful however to know the terms specified above in order to understand the operator names used in RapidMiner.

In proposed system we are applied to the dataset using the holdout method (WEKA “Percentage Split” test option, 66%/34%), as shown on Fig.1. The dataset is divided into 3 parts and, each time an algorithm is run, 2/3 of the data is used for training of the classification model and 1/3 of the data is used for testing and evaluation of the model.

Performance evaluation and cross-validation

The numerous operators that apply machine learning procedures to datasets can be easily used in combination with other operators. Typical examples of operators used in the evaluation of learning procedures are cross-validation, operators for computing standard quality measures, parameter optimizations and last but not least logging operators for creating procedure performance profiles. Since RapidMiner supports loops, processes can also be created that apply the new procedure to several datasets and compare it with other procedures. A process that enables such a validation of one’s own procedure can be found in the example repository.

Preprocessing

In many cases, procedures need preprocessing steps first in order to be able to deal with the data in the first place. If we want to use the operator k-NN for the k-nearest-neighbor method, we must note for example that scale differences in the individual attributes can render one attribute more important than all others and make the neighborhood relation dominate in the Euclidean space. We must therefore normalize the data in this case by adding a Normalize operator first. However, if we add the Normalize operator before the cross-validation, all data will be used to determine the average value and the standard deviation. This means however that knowledge about the test part is already implicitly contained in the training part of the normalized dataset within the cross-validation. Possible outliers, which are only present in the test part of the dataset, have already affected the scaling, which is why the attributes are weighted differently. This is a frequent error which leads to statistically invalid quality estimations. In order to prevent this, we must drag all preprocessing steps into the cross validation and execute them in the training sub process. If we do not execute any further adjustment in the process, the model generated in the training process will of course be confronted with the not yet normalized data in the test process. This is why all preprocessing operators, the results of which depend on the processed data, offer so-called preprocessing models. These can be used to execute an identical transformation again. Thus the same average values and standard deviations are used to transform at the time of normalization, instead of re-computing these on the current data.

Parameter Optimization

It is therefore very easy on the whole to perform a real validation of a procedure in RapidMiner. However, almost every procedure has certain parameters with which the quality of the models can be influenced. The results will be better or worse depending on the setting. So if it is to be shown that a new procedure is superior to an existing one, you cannot just optimize the parameters of your own procedure or even set the parameters arbitrarily. The performance of procedures such as the support vector machine or a neural network in particular depends greatly on the parameter settings. Therefore RapidMiner offers the possibility of looking for the best parameter settings automatically. To do this, you use one of the Optimize Parameters operators. The operator Optimize Parameters (Grid) can be controlled most simply. It iterates over a number (previously defined by the user) of combinations of the parameters to be optimized. For each parameter combination it executes its internal sub process. Accordingly, only parameters of operators of this sub process can be optimized. The sub process must return a performance vector here (e.g. the Accuracy), using which Optimize Parameters can recognize the quality of the current combination. After it has tested all parameter combinations, the Optimize Parameters operator returns the combinations with maximum performance measured in their cycle.

We have now seen how algorithms can be compared. However, in order to evaluate new (i.e. self-developed) algorithms in this way, these must of course be integrated into RapidMiner.

RapidMiner was originally developed exactly for this application: New algorithms were to be comfortably, quickly and easily comparable with other algorithms. Implementing new learning procedures in RapidMiner is very easy. It is merely necessary to create two Java classes, of which one performs the learning on the training dataset, i.e. the estimating of the model parameters. The other class must save these parameters and be able to apply the model to new data i.e. make predictions. This will be introduced briefly in the following based on a fictitious learning procedure.

Discussions

The proposed system provides overall system overview model for propose educational data mining. Various classification as well as recommendation algorithm as illustrated in entire architecture. Various data mining tools also available achieve inbuilt results from input data and gain the background knowledge. For the proposed statistical analysis first system needs to generate the train module and extract the important features according to different class labels. Once the training module has built system deals with testing data, the basic objective behind the prediction on test data to generate the statistical analysis. For this work we have evaluate the system in waka 3.8 open source environment with 1000 instances. Various classifiers already available in weka in which is used to evaluate the system. Various kind of confusion matrix has achieved based on different kind of fold validations.

According to existing studies and evaluation of existing algorithms we have shown system performance in below table 1 as well as figure 2

Table 1 : Performance analysis of various algorithm for mining a large data

Algorithm	Accuracy	Error Rate
C 4.5	87.30	8.50
SVM	89.40	7.40
ANN	90.10	9.10
NB	91.30	7.60

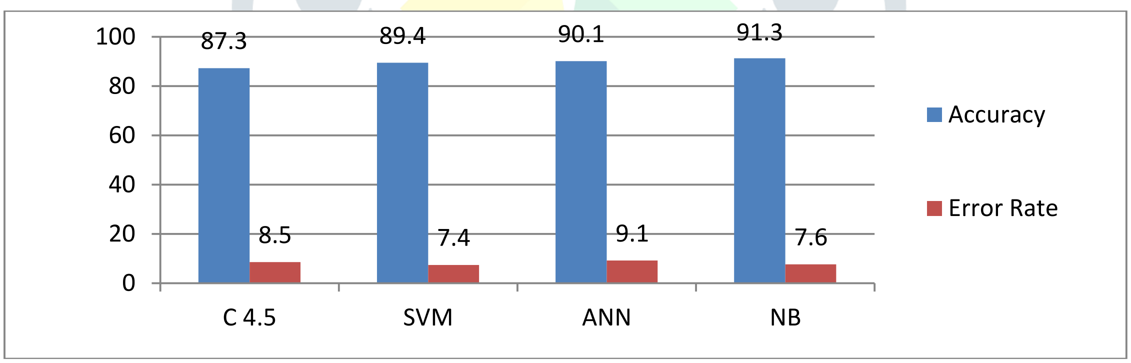

Figure 2: System performance with various existing algorithm

Finally we got the complete idea of proposed work and how it deals with different classification algorithms, and we got the clear idea for future work.

Conclusion

This work , multiple data processing tasks were used to produce qualitative precognitive models that were efficiently and effectively able to predict the students' grades from a collected coaching dataset. First, a survey was created that has targeted university students and picked up multiple personal, social, and academic data related to them. Second, the collected dataset was preprocessed and explored to become acceptable for the information mining tasks. Third, the implementation of data mining tasks was given on the dataset in hand to come up with classification models and testing them. Finally, fascinating results were drawn from the classification models, as well as, fascinating patterns at intervals the Naïve mathematician model was found. Four decision tree algorithms are implemented, as well as, with the Naïve mathematician algorithmic program. at intervals this study, it completely was slightly found that the student's performance is not totally hooked in to their educational efforts, in spite, there unit many different factors that have adequate larger influences to boot. lastly, this study can inspire and facilitate universities to perform data mining tasks on their students' data usually to go looking out fascinating results and patterns which could facilitate every the university additionally as a result of the students in some ways.

References

[1] Baradwaj, B.K. and Pal, S., 2011. Mining Educational Data to Analyze Students' Performance. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 2, No. 6, 2011.
[2] Ahmed, A.B.E.D. and Elaraby, I.S., 2014. Data Mining: A prediction for Student's Performance Using Classification Method. World Journal of Computer Application and Technology, 2(2), pp.43-47.
[3] Pandey, U.K. and Pal, S., 2011. Data Mining: A prediction of performer or underperformer using classification. (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (2), 2011, 686- 690.
[4] Bhardwaj, B.K. and Pal, S., 2012. Data Mining: A prediction for performance improvement using classification. (IJCSIS) International Journal of Computer Science and Information Security, Vol. 9, No. 4, April 2011.
[5] Yadav, S.K., Bharadwaj, B. and Pal, S., 2012. Data Mining Applications: A Comparative Study for Predicting Student's Performance. International Journal of Innovative Technology & Creative Engineering (ISSN: 2045-711), Vol. 1, No.12, December
[6] Quinlan j r. (1986). Induction of decision trees. machine learning. pp.(4): 81-106.
[7] Quinlan j r. (1994). C4.5: Programs for machine learning. machine learning. pp.(3): 235-240.
[8] Yazdani a, Ebrahimi t, Hoffmann u. (2009), " Classification of eeg signals using dempster shafer theory and a k-nearest neighbor classifier",IEEE. in proc of the 4th int embs conf on Neural Engineering, pp. 327-30.
[9] Demuth h, Beale m, Hagan m. (2009). Neural Network toolbox™ user's guide. the mathworks, inc, pp. 1-901.