Indian Journal of Science and Technology, Vol 8(14), DOI: 10.17485/ijst/2015/v8i14/72730, July 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

An ID3 Algorithm for Performance of Decision Tree in Predicting Student's Absenteeism in an Academic Year using Categorical Datasets

N. Venkatesan[1*], K. Arunmozhi Arasan[2] and S. Muthukumaran[2]

[1]Department of IT, Bharathiyar College of Engineering and Technology, Karaikal - 609609, Puducherry, India; envenki@gmail.com

[2]Department of Computer Science, Siga College of Management and Computer Science, Villupuram, India

Abstract

Objective: The main objective of higher educational institutes is to provide quality education to its students and to improve the quality of managerial decisions. Objective of this paper is to use the data mining techniques like decision tree induction, rule mining to predict the student's behavior. **Methods:** We have used ID3 algorithm to predict the reason why students take leave from classes. The data collected from 123 students studying in an arts and science college located in semi-rural area in Villupuram district. Data collection is base on the questionnaire in five point scale method. Decision tree induction algorithm (ID3) was used to predict the reason and the decision tree was constructed using Entropy and Information Gain. **Findings:** The result obtained from Tanagra tool shows that job attribute plays a key role in predicting the absenteeism of students. That is lack of attendance is due to the job attribute. The results can be used to take managerial decisions. **Application:** Discovering the knowledge from the educational data helps to achieve the highest level of quality in higher education. The results shows that the decision tree induction algorithms can be used to predict the student's behavior as well as to improve the managerial decision making process.

Keywords: Categorical Data, Data Mining, Decision Trees Induction, ID3 Algorithm, Rule Mining

1. Introduction

Currently many educational institutions especially small-medium education institutions are facing problems with the lack of attendance among the students. The universities will allow the students who have attendance above than 80% to the semester exam; if a student who have attendance percentage below 80% will lack attendance and are not permitted to write the semester exam. All educational institutions are facing this problem so this research aims to find the reason for a student to put leave to the college and take immediate actions to overcome this problem.

In[1] presented their study on student recruiting on higher education institutions. The aim of this research is applying data mining technique to classify student recruiting data in higher education institutions. The objectives of this study are to test the validity of the model derived from decision rules and to find the right algorithm for data classification task[2]. From comparison of 4 algorithms; J48, Id3, Naive Bayes and OneR, The goal is to predict the features of students who are likely to undergo the student admission process[3]. Compared the five classification algorithm to choose the best classification algorithm for Course Recommendation system[4]. These five classification algorithms are ADTree, Simple Cart, J48, ZeroR and Naive Bays Classification Algorithm. They compare these six algorithms using open source data mining tool Weka and present the result.

Databases are rich with hidden information that can be used for intelligent decision making. Classification and prediction are two forms of data analysis that can be used to extract models describing important data classes or to predict future data trends[5]. Such analysis can

*Author for correspondence

help provide us with a better understanding of the data at large. Classification and prediction have numerous applications, including fraud detection, target marketing, performance prediction, manufacturing and medical diagnosis[6]. Classification is used to find the class label for the data and prediction is used find the value in the class label[7].

1.1 Classification by Decision Tree Induction

Decision tree induction is the learning of decision trees from class-labeled training tuples. A decision tree is a flow chart like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (or terminal node) holds a class label. The topmost node in a tree is the root node[8].

1.2 Decision Tree Induction Algorithm

ID3 (Iterative Dichotomiser) adopts a greedy (i.e. nonbacktracking) approach in which decision trees are constructed in a top-down recursive divide-and-conquer manner. The training set is recursively partitioned into smaller subsets as the tree is being built[9]. A basic decision tree algorithm is summarized below;

Algorithm: Generate decision tree. Generate a decision tree from the training tuples of data partition D.

Input:

- Data partition, D, which is a set of training tuples and their associated class labels.
- Attribute_list, the set of candidate attributes.
- Attribute_selection_method, a procedure to determine the splitting criterion that "best" partitions the data tuples into individual classes. This criterion consists of a splitting_attribute and possibly either a split point or splitting subset.

Output: A decision tree

Method:

1. Create a node N:
2. **If** tuples in D are all of the same class, C then
3. Return N as a leaf node labeled with the class C
4. **If** attribute_list is empty then
5. Return N as a leaf node labeled with the majority class in D; //majority voting
6. Apply Attribute_selection_method(D, attribute_list) to find the "best" splitting_criterion";
7. Label node N with splitting_criterion;
8. If splitting_attribute is discrete-valued and multiway splits allowed then //not restricted to binary trees
9. Attribute_list→attribute_list-splitting attribute; // remove splitting_attribute
10. For each outcome j of splitting_criterion //partition the tuples and grow subtrees for each partition
11. Let Dj be the set of data tuples in D satisfying outcome j; //a partition
12. If Dj is empty then
13. Attach a leaf labeled with the majority class in D to node N;
14. Else attach the node returned by Generate_decision_tree (Dj, attribute_list) to node N; **endfor**
15. Return N;

1.3 Information Gain

1.3.1 Entropy

Entropy[10] uses ***information gain*** as its attributes selection measure. The attribute with highest information gain is chosen as the splitting attribute for node N. This attribute minimizes the information needed to classify the examples in the resulting partitions and reflects the least randomness or "impurity" in these partitions. The expected information needed to classify an example in dataset D is given by

$$\mathrm{Info}(\mathrm{D}) = \sum_{-i=1}^{m} p_i log_2(p_i)$$

Where p_i is the probability that an arbitrary example in dataset D belongs to the class Ci and is estimated by |Ci,D|/|D|. A log function to the base 2 is used because the information is encoded in bits. Info (D) is just the average amount of information needed to identify the class label of an example in dataset D. Partitioning (e.g., where a partition may contain a collection of examples from different classes rather than from a single class)[11] to produce an exact classification of the examples by

$$info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} * info(D_j)$$

The term $|D_1|/|D|$ acts as the weight of the j^{th} partition $Info_i(D)$ is the expressed information required to classify

an example from dataset D based on the partitioning by A. The information gain is defined as the difference between the original information requirement and the new requirement that is.

$$Gain\ (A) = Info\ (D) - Info_A(D)$$

The attribute A with the highest information gain (Gain(A)) is chosen as the splitting attribute at node N[12].

1.4 Data Collection

The data are collected from a private college at Ulundurpet in Villupuram district. There were 123 records collected from the students who are doing under graduate course who belongs to the age group 18 to 23. Among the 123 students 85 were male and 38 were female candidates.

The data used for data mining contains 123 records and have 30 dimensional attribute namely name, gender, age, department, year, mode of transport, college location, home location, test, cinema, festival, sick, miss bus, friend leave, subject boring, staff question, exam study, result, occasionally, institution work, part time job, assignment, pay fees, native, accident, dress code, commitment friends, college care, impress, problem in college. For our study name is not necessary so we omit the attribute and take the 29 attributes for classification.

1.5 System Framework

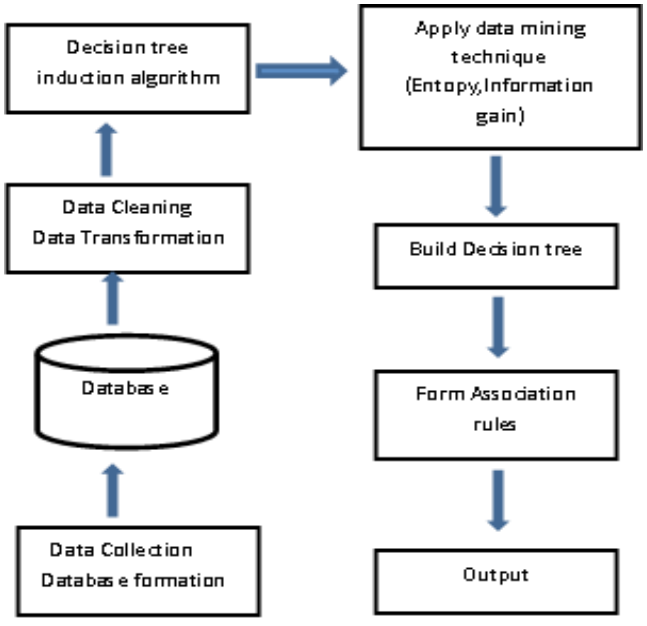

Figure 1. System Frame Work.

2. Experimental Setup

Tanagra is open source software used for data mining. It supports all the basic type of data formats like *.xls, *.txt etc. and it is very user friendly[13]. The data set is implemented in Tanagra by the following method.

- Open the Tanagra software and go to the file menu and click open then insert the data you want to evaluate
- Go to Data Visualization and select view dataset and drag into the data set
- Select the view dataset and right click and click execute and then click view then the data is displayed on the right side screen
- Drag the Define status from the icon and give the target attribute as gender and in input select all the attributes.
- Go to spv learning and select ID3 and drag into the Define status and right click it and click execute and then click view then the results are displayed in the right side screen[14].

2.1 Evaluation

The analysis of the data is done on the basis of gender [Table 1 and 2]. This is the reason why a male student take leave and the reason why a female student take leave to the college by applying Entropy and Information gain

$$\text{info}(D) = \sum_{-i=1}^{m} p_i log_2(p_i)$$

$$= \left(\frac{85}{123}\log_2\frac{85}{123} - \frac{38}{123}\log_2\frac{38}{123}\right)$$

$$= 0.892$$

The gain for the attribute part time job is

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} * Info(D_j)$$

$$\frac{26}{123}\left(\frac{25}{26}\log_2\frac{25}{26} - \frac{1}{26}\log_2\frac{1}{26}\right) + \frac{30}{123}\left(\frac{28}{30}\log_2\frac{28}{30} - \frac{2}{30}\log_2\frac{2}{30}\right)$$
$$+ \frac{14}{123}\left(\frac{6}{14}\log_2\frac{6}{14} - \frac{8}{14}\log_2\frac{8}{14}\right) + \frac{27}{123}$$
$$\left(\frac{14}{27}\log_2\frac{14}{27} - \frac{13}{27}\log_2\frac{13}{27}\right) + \frac{26}{123}\left(\frac{12}{26}\log_2\frac{12}{26} - \frac{14}{26}\log_2\frac{14}{26}\right)$$

$$= 0.050+0.086 + 0.112 + 0.219 + 0.210 = 0.678$$

Gain (A) =Info (D) –InfoA(D)

$$= 0.892\text{-}0.678 = 0.214$$

The information gain for the attribute job has the highest value in the database and the other values in the

Table 1. Total records

Male	Female	Total
85	38	123

Table 2. Total records for the attribute Job

Job	Male	Female	Total
Strong agree	25	1	26
Agree	28	2	30
Neutral	6	8	14
Disagree	14	13	27
Strong Disagree	12	14	26
TOTAL	85	38	123

table are listed. The job attribute is taken as the root node of the tree [Figure 2] and the 123 records are split by the branches of the job as strongly agree 26 records and Agree 30 records and Neutral 14 records and Disagree 27 records and Strongly Disagree 26 records and the process is applied for the each table and decision tree is formed as below[15].

3. Result and Discussion

The results obtained from Tanagra is shown in the figure below. Here job attribute is taken as the root node [Figure 3] because it has the highest information gain so all job is the root node and it has five discrete attributes as its values called A) Strongly Agree, B) Agree, C) Neutral, D) Dis Agree and E) Strongly Dis Agree. Here in Tanagra the root is violet in colour and next child node is red in colour using this dots we can identify the nodes in Tanagra [Figure 2].

3.1 Rules Extraction from Decision Tree

Rules are good way of representing information or bits of knowledge. A rule based classifier uses a set of IF-THEN rules for classification[7]. An IF_THEN rule is an expression of the form

IF condition THEN conclusion

To extract rules from a decision tree, one rule is created for each path from the root to a leaf node. Each splitting criterion along a given path is logically ANDed to form the rule antecedent (IF part). The leaf node holds the class prediction forming the rule consequent (THEN part)[8].

R1: IF (Job=strongly agree) ^ (Dept. =BCA) → (gender = male)

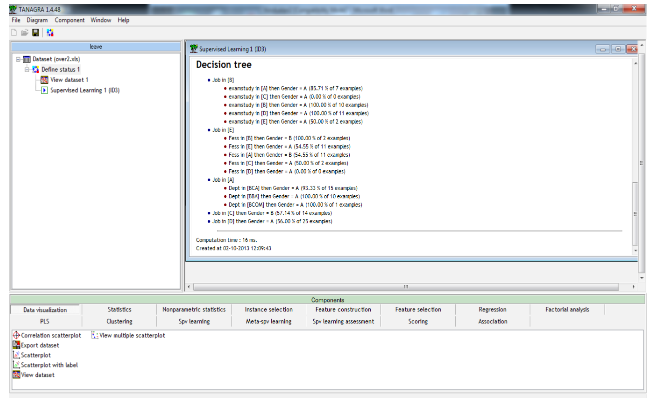

Figure 2. Decision Tree in Tanagra.

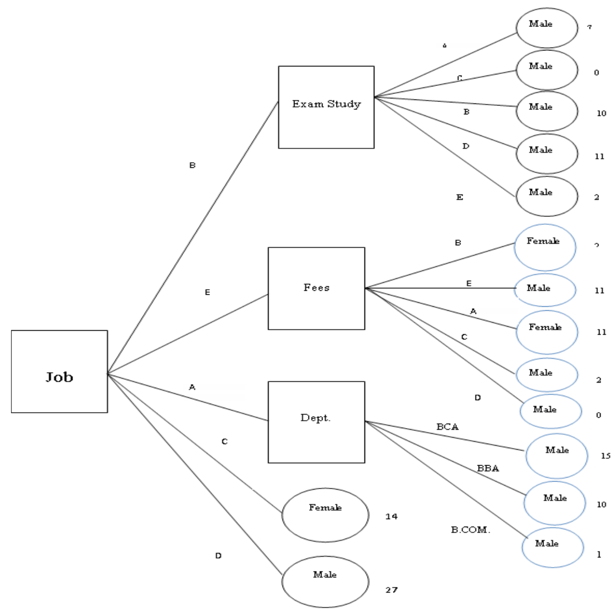

Figure 3. Decision Tree formed by the results obtained from Tanagra.

Coverage (R1) =15/123= 12.19 %

R2: IF (Job = strongly agree) ^ (Dept. =BBA) → (gender = male)

Coverage (R2) =10/123= 8.13 %

R3: IF (Job=agree) ^ (Exam study= strongly agree) → (gender=male)

Coverage (R3) =7/123= 5.69 %

R4: IF (Job=agree) ^ (Exam study = agree) → (gender = male)

Coverage (R4) =10/123= 8.13 %

R5: IF (Job=agree) ^ (Exam study= disagree) → (gender=male)

Coverage (R5) = 11/123= 8.94 %

R6: IF (Job=neutral) → (gender=female)
Coverage (R6) = 14/123= 11.38 %
R7: IF (Job=disagree) → (gender=male)
Coverage (R7) = 27/123= 21.95 %
R8: IF (Job=strongly disagree) ^ (Fees=strongly agree) → (gender=female)
Coverage (R8) = 11/123= 8.94 %
R9: IF (Job=strongly disagree) ^ (Fees=strongly disagree) → (gender=male)
Coverage (R9) = 11/123= 8.94 %

The decision tree obtained for our dataset is shown in the diagram below [Figure 3] and the rules are drawn from the root to the leaf. There are many rules got but they are pruned by their percentage of coverage of the rules. The rules with less coverage are removed because many rules are very complex to understand.

3.2 Recommendations

The job attribute plays a key role on job going students to earn money. To improve good learning environment and the quality of education in the rural and semi-rural areas, our suggestion is that change the college timings such as morning and evening sessions to avoid the students absenteeism for the classes. It was found that student put leave due to the purpose of studying for the examinations so if we give enough study holidays we can avoid students putting leave to college.

4. Conclusion

The purpose of this study was to identify the reason for student absenteeism. From the results obtained clearly shows that rural based college students have financial problems for studying. In this research, the decision tree was constructed using ID3 algorithm and it is easy to interpret and contributing to the improved results to be compacted.

5. References

1. Naenudorn E, Singthongchai J, Kerdprasop NA, Kerdprasop K. Classification model induction for student recruiting. Latest Advances in Educational Technologies. 2012; 117–22.
2. Aher SB, Lobo LMRJ. Comparative Study of Classification Algorithms. International Journal of Information Technology. 2012; 5(2):239–43.
3. Han J, Kamber M, Pei J. Data Mining, Southeast Asia edition: Concepts and techniques. Morgan kaufmann; 2006.
4. Kumar SA, Vijayalakshmi MN. Efficiency of decision trees in predicting student's academic performance. First International Conference on Computer Science, Engineering and Applications. 2011; 2:335–43.
5. Hongjie SH. Research on Student Learning Result System based on Data Mining. IJCSNS. 2010; 10(4):203.
6. Sehgal L, Mohan N, Sandhu PS. Quality prediction of function based software using decision tree approach. International Conference on Computer Engineering and Multimedia Technologies (ICCEMT); 2012. p. 43–7.
7. Questier F. The use of CART and multivariate regression trees for supervised and unsupervised feature selection. Chemometrics and Intelligent Laboratory Systems. 2005; 76(1):45–54.
8. Antonia V. Diagnosis of ovarian cancer using decision tree classification of mass spectral data. BioMed Research International. 2003; 2003(5):308–14.
9. Lu SH, Chiang DA, Keh HC, Huang HH. Chinese text classification by the Naive Bayes Classifier and the associative classifier with multiple confidence threshold values. Knowledge-based systems. 2010; 23(6):598–604.
10. Bodon F, Ronyai L. Trie: an alternative data structure for data mining algorithms. Mathematical and Computer Modelling. 2003; 38(7):739–51.
11. Lin KC, Liao IE, Chang TP, Lin SF. A frequent itemset mining algorithm based on the Principle of Inclusion–Exclusion and transaction mapping. Information Sciences. 2014; 276:278–89.
12. Verma A, Khan SD, Maiti J, Krishna OB. Identifying patterns of safety related incidents in a steel plant using association rule mining of incident investigation reports. Safety science. 2014; 70:89–98.
13. Liu YH, Wang CS. Constrained frequent pattern mining on univariate uncertain data. Journal of Systems and Software. 2013; 86(3):759–78.
14. Wu M, Wang L, Li M, Long H. An approach of product usability evaluation based on Web mining in feature fatigue analysis. Computers & Industrial Engineering. 2014; 75:230–8.
15. Yu KM, Zhou J. Parallel TID-based frequent pattern mining algorithm on a PC Cluster and grid computing system. Expert Systems with Applications. 2010; 37(3):2486–94.