The Open Automation and Control Systems Journal, **2015**, *7*, 2300-2305

Application of Research on Decision Tree Algorithm for Sports Grade Analysis

Zhu Lini*

Xi'an Physical Education University 710068, Shaanxi, China

Abstract: This paper introduces and analyzes the data mining in the management of students' sports grades. We used the decision tree in the analysis of grades and investigated attribute selection measures including data cleaning. We took sports course score of some university for example and produced decision tree using ID3 algorithm which gives the detailed calculation process. Because the original algorithm lacks termination condition, we proposed an improved algorithm which can help us to find the latency factor which impacts the sports grades.

Keywords: Classification, decision tree algorithm, ID3 algorithm, sports grade analysis.

1. INTRODUCTION

With the rapid development of higher education, sports grade analysis as an important guarantee for the scientific management constitutes the main part of the sports educational assessment. The research on the application of data mining in the management of students' grades explores how to get the useful uncovered information from the large amounts of data using the data mining and grade management techniques [1-5]. It introduces and analyzes the data mining in the management of students' grades. It uses the decision tree for the analysis of grades. It describes the function, status and deficiency of the management of students' grades. It tells us how to employ the decision tree in the management of students' grades. It improves the ID3 arithmetic to analyze the students' grades so that we could find the latency factor which impacts the grades. If we can find out the factors, we can offer the decision-making information to teachers. It also seeks to advance the quality of teaching [6-10]. The sports grade analysis helps teachers to improve the teaching quality and provides decisions for school leaders.

The decision tree-based classification model is widely used for its unique advantage. Firstly, the structure of the decision tree method is simple and it generates easy to understand rules. Secondly, the high efficiency of the decision tree model is more appropriate for the case of a large amount of data in the training set. Furthermore, the computation of the decision tree algorithm is relatively not large enough. The decision tree method usually does not require knowledge of the training data, and it specializes in the treatment of non-numeric data. Finally, the decision tree method has high classification accuracy, and it is to identify common characteristics of library objects, and classify them in accordance with the classification model.

The original decision tree algorithm uses the top-down recursive way [11, 12]. Comparison of property values is done in the internal nodes of the decision tree and according to the different property values judge down branches from the node. We get conclusion from the decision tree leaf node. Therefore, although a path from the root to the leaf node corresponds to conjunctive rules, the entire decision tree corresponds to a set of disjunctive expressions rules. The decision tree generation algorithm is divided into two steps [13-15]. The first step is the generation of the tree, and at the beginning all the data is in the root node, then do the recursive data slice. Tree pruning is to remove some of the noise or abnormal data. Conditions of decision tree to stop splitting is that a node data belongs to the same category and there are not attributes used to split the data.

In the next section, we introduce construction of decision tree. In Section 3, we introduce attribute selection measure. In Section 4, we do empirical research based on ID3 algorithm and propose an improved algorithm. In Section 5, we conclude the paper and give some remarks.

2. CONSTRUCTION OF DECISION TREE USING ID3

The growing step of the decision tree is shown in Fig. (**1**). Decision tree generation algorithm is described as follows: The name of the algorithm is $Generate\_decision\_tree$ which produces a decision tree by the given training data (Fig **1**). The input is training samples which is represented with discrete values. Candidate attribute is the set of attributes. The output is a decision tree.

Step 1. Set up node N. If sample is in the same class C then return to N as lead node and label it with C.

Step 2. If attribute_list is empty, then return to N as leaf node and label it with the most common class in the samples.

Step 3. Choose $test\_attribute$ with information gain in the attribute_list, and label N as $test\_attribute$.

Step 4. While each a_i in every $test\_attribute$ do the following operation.

*Address correspondence to this author at the Xi'an Physical Education University 710068, Shaanxi, China; Tel: 18986139113;
E-mail: linizhu@163.com

1874-4443/15 2015 Bentham Open

Step 5. Node N produces a branch which meets the condition of $test\_attribute = a_i$

Step 6. Suppose S_i is sample set of $test\_attribute = a_i$ in the samples. If S_i is empty, then add a leaf and label it as the most common class. Otherwise add a node which was returned by:

$Generate\_decision\_tree(s_i, attribute\_list - test\_attribute)$

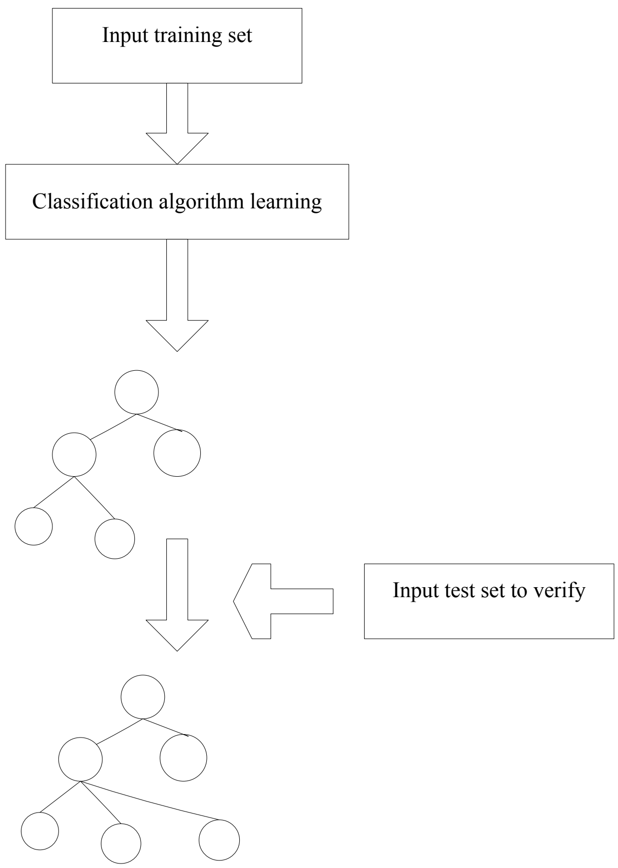

Fig. (1). Growing step of the decision tree.

3. AN IMPROVED ALGORITHM

3.1. Attribute Selection Measure

Suppose S is data sample set of s number and class label attribute has m different values $C_i (i = 1, 2, \cdots, m)$. Suppose S_i is the number of sample of class C_i in S. For a given sample classification, the demanded expectation information is given by formula 1.

$$I(s_{1j}, s_{2j}, K, s_{mj}) = -\sum_{i=1}^{m} p_{ij} \log_2 p_{ij} (i = 1, 2, \cdots, K, n) \tag{1}$$

$$E(A) = \sum_{j=1}^{V} \frac{(S_{1j} + S_{2j} + \cdots + S_{mj})}{S} I(S_{1j}, S_{2j}, K, S_{mj}) \tag{2}$$

p_i is probability that random sample belongs to C_i and is estimated by s_i / s. Suppose attribute A has V different values $(a_1, a_2, \cdots, a_V)$. We can use attribute A to classify S into V number of subset $(S_1, S_2, \cdots S_V)$. Suppose S_{ij} is the number of class C_i in subset S_j. The expected information of subset is shown in formula 2. $\frac{(S_{1j} + S_{2j} + \cdots + S_{mj})}{S}$ is the weight of the j-th subset. For a given subset S_j formula 3 sets up.

$$I(s_{1j}, s_{2j}, K, s_{mj}) = -\sum_{i=1}^{m} p_{ij} \log_2 p_{ij} (i = 1, 2, \cdots, K, n) \tag{3}$$

$p_{ij} = \frac{s_{ij}}{|s_j|}$ is the probability that samples of s_j belongs to class C_i. If we branch in A, the information gain is shown in formula 4[14].

$$Gain(A) = I(s_1, s_2, \cdots, s_m) - E(A) \tag{4}$$

3.2. The Improved Algorithm

The improved algorithm is as follows: Function $Generate\_decision\_tree$ (training samples, candidate attribute attribute_list)

{ Set up node N;

If samples are in the same class C then

Return N as leaf node and label it with C;

Record statistical data meeting the conditions on the leaf node;

If attribute_list is empty then

Return to N as the leaf node and label it as the most common class of samples;

Record statistical data meeting the conditions on the leaf node;

Suppose GainMax=max(Gain1, Gain2, … , Gainn)

If GainMax< threshold

Return to N as the leaf node and label it as the most common class of samples;

Choose attribute with the highest information gain of attribute_list;

Label N as test_attribute;

For each a_i of test_attribute, produce a branch from node N meeting the condition of test_attribute = a_i;

Suppose S_i sample set of samples meeting the condition of test_attribute = a_i;

If S_i is empty then Record statistical data meeting the conditions on the leaf node;

Add a leaf and label it as the most common class of samples;

Table 1. Examination score of the students.

Course Code	Whether Re-Learning	Paper Difficulty	Whether Required Course	Score
110101290	no	high	yes	89
H200104088	no	middle	yes	75
H2001 16090	yes	middle	no	80
H120101160	yes	high	yes	65
120101288	yes	middle	yes	70
H200152069	no	low	no	90

Else add a node returned by $Generate\_decision\_tree$ (S_i, attribute_list_test_attribute);

}

4. EMPIRICAL RESEARCH

4.1. Data Cleaning

This paper takes sports course score of some universities for example. Examination score of the students is shown in Table **1**.

Data in Table **1** was not suitable for classification, so we initially performed data cleaning. According to the general course, basic course, professional basic course and specialized course, classify the course into A, B, C, and D. Score is divided into three categories outstanding, medium, and general. Paper difficulty is divided into three categories 1, 2, and 3. Such as:

Update ks set ci_pi='outstanding' where ci_pj>='85'

Update ks set ci_pi=' medium' where ci_pj>='75'and ci_pj<'85'

Update ks set ci_pi='general' where ci_pj>='60'and ci_pj<'75'

Update ks set sjnd='high' where sjnd='1'

Update ks set sjnd=' medium' where sjnd='2'

Update ks set sjnd='low' where sjnd='3'

4.2. Result of ID3 Algorithm

Table **2** contains a training set of student test scores situation information after data cleaning. We classify the samples into three categories. $C_1=$"outstanding", $C_2=$"medium", $C_3=$"general", $s_1=300$, $s_2=1950$, $s_3=880$, $s=3130$. According to formula 1, we obtain

$$I(s_1,s_2,s_3)=(300,1950,880)$$
$$=-(300/3130)\ /\log_2(300/3130)$$
$$-(1950/3130)\log_2(1950/3130)-(880/3130)\log_2(880/3130)$$
$$=1.256003.$$

Entropy of every attribute is calculated as follows: Firstly calculate whether re-learning. For yes, $s_{11}=210$, $s_{21}=950$, $s_{31}=580$.

$$I(s_{11},s_{21},s_{31})=(210,950,580)$$
$$=-(210/1740)\log_2(210/1740)-(950/1740)\log_2(950/1740)$$
$$-(580/1740)\log_2(580/1740)=1.074901$$

For no, $s_{12}=90$, $s_{22}=1000$, $s_{32}=300$.

$$I(s_{12},s_{22},s_{32})=(90,1000,300)$$
$$=-(90/1390)\log_2(90/1390)-(1000/1390)\log_2(1000/1390)$$
$$-(300/1390)\log_2(300/1390)=1.373186.$$

If samples are classified according to re-learning requirement, the expected information is:

$$E(\text{"}whether\quad re\text{-}learning\text{"})=(1740/3130)\cdot I(s_{11},s_{21},s_{31})$$
$$+(1390/3130)\cdot I(s_{12},s_{22},s_{32})$$
$$=0.555911\cdot 1.074901+0.444089\cdot 1.373186=1.240721.$$

So the information gain is:

$$Gain(\text{"}whether\quad re\text{-}learning\text{"})=I(s_1,s_2,s_3)$$
$$-E(\text{"}whether\quad re\text{-}learning\text{"})=0.015282.$$

Secondly, calculate course type, when it is A, $s_{11}=110, s_{21}=200, s_{31}=580$.

$$I(s_{11},s_{21},s_{31})=(110,200,580)$$
$$=-(110/890)\log_2(110/890)-$$
$$(200/890)\log_2(200/890)-(580/890)\log_2(580/890)$$
$$=1.259382.$$

For course type B, $s_{12}=100, s_{22}=400, s_{32}=0$.

$$I(s_{12},s_{22},s_{32})=(100,400,0)$$
$$=-(100/500)\log_2(100/500)$$
$$-(400/500)\log_2(400/500)-0$$
$$=0.721928.$$

For course type C, $s_{13}=0, s_{23}=550, s_{33}=0$.

$I(s_{13}, s_{23}, s_{33}) = (0, 550, 0)$
$= -(0/550)\log_2(0/550)$
$-(550/500)\log_2(550/500) - 0$

$= 1.168009$.

For course type D, $s_{14} = 90, s_{24} = 800, s_{34} = 300$.

$I(s_{14}, s_{24}, s_{34}) = (90, 800, 300)$
$= -(90/1190)\log_2(90/1190) -$
$(800/1190)\log_2(800/1190) - (300/1190)\log_2(300/1190)$

$= 1.168009$.

$E(\text{"}course \quad type\text{"}) = (890/3130)\cdot I(s_{11}, s_{21}, s_{31})$
$+(500/3130)\cdot I(s_{12}, s_{22}, s_{32})$

$+(550/3130)\cdot I(s_{13}, s_{23}, s_{33})$
$+(1190/3130)\cdot I(s_{14}, s_{24}, s_{34})$
$= 0.91749.$

$Gain(\text{"}course \quad type\text{"}) = 1.256003 - 0.91749 = 0.338513$.

Thirdly, calculate the paper difficulty as follows:

For high, $s_{11} = 110, s_{21} = 900, s_{31} = 280$.

$I(s_{11}, s_{21}, s_{31}) = (110, 900, 280)$
$= -(110/1290)\log_2(110/1290) -$
$(900/1290)\log_2(900/1290) - (280/1290)\log_2(280/1290)$

$= 1.14385$.

For medium, $s_{12} = 190, s_{22} = 700, s_{32} = 300$.

$I(s_{12}, s_{22}, s_{32}) = (190, 700, 300)$
$= -(190/1190)\log_2(190/1190) -$
$(700/1190)\log_2(700/1190) - (300/1190)\log_2(300/1190)$
$= 1.374086$.

For low, $s_{13} = 0, s_{23} = 350, s_{33} = 300$.

$I(s_{13}, s_{23}, s_{33}) = (0, 350, 300)$
$= -(0/650)\log_2(0/650) - (350/650)\log_2(350/650)$
$-(300/650)\log_2(300/650) = 0.995727.$
$E(\text{"}paper \quad difficulty\text{"}) = (1290/3130)\cdot I(s_{11}, s_{21}, s_{31})$
$+(1190/3130)\cdot I(s_{12}, s_{22}, s_{32})$
$+(650/3130)\cdot I(s_{13}, s_{23}, s_{33}) = 1.200512.$

$Gain(\text{"}paper \quad difficulty\text{"}) =$
$1.256003 - 1.200512 = 0.55497.$

Fourthly, calculate whether required course. For yes, $s_{11} = 210, s_{21} = 850, s_{31} = 600$

$I(s_{11}, s_{21}, s_{31}) = (210, 850, 600)$
$= -(210/1660)\log_2(210/1660) -$
$(850/1660)\log_2(850/1660) - (600/1660)\log_2(600/1660)$

$= 1.220681.$

For no, $s_{12} = 90, s_{22} = 1100, s_{32} = 280$

$I(s_{12}, s_{22}, s_{32}) = (90, 1100, 280)$
$= -(90/1470)\log_2(90/1470) -$
$(1100/1470)\log_2(1100/1470) - (280/1470)\log_2(280/1470)$

$= 1.015442.$

$E(\text{"}whether \quad required\text{"}) = (1660/3130)\cdot I(s_{11}, s_{21}, s_{31})$
$+(1470/3130)\cdot I(s_{12}, s_{22}, s_{32})$
$= 1.220681.$

$Gain(\text{"}whether \quad required\text{"}) =$
$1.256003 - 1.220681 = 0.035322.$

Table 2. Training set of student test scores.

Course Type	Whether Re-Learning	Paper Difficulty	Whether Required	Score	Statistical Data
D	no	medium	no	outstanding	90
B	yes	medium	yes	outstanding	100
A	yes	high	yes	medium	200
D	no	low	no	medium	350
C	yes	medium	yes	general	300
A	yes	high	no	medium	250
B	no	high	no	medium	300
A	yes	high	yes	outstanding	110
D	yes	medium	yes	medium	500
D	no	low	yes	general	300
A	yes	high	no	general	280
B	no	high	yes	medium	150
C	no	medium	no	medium	200

Table 3. Special case for classification of the sub-tree.

Course Type	Whether Re-Learning	Paper Difficulty	Whether Required	Score	Statistical Data
A	no	high	yes	medium	15
A	no	high	yes	general	20

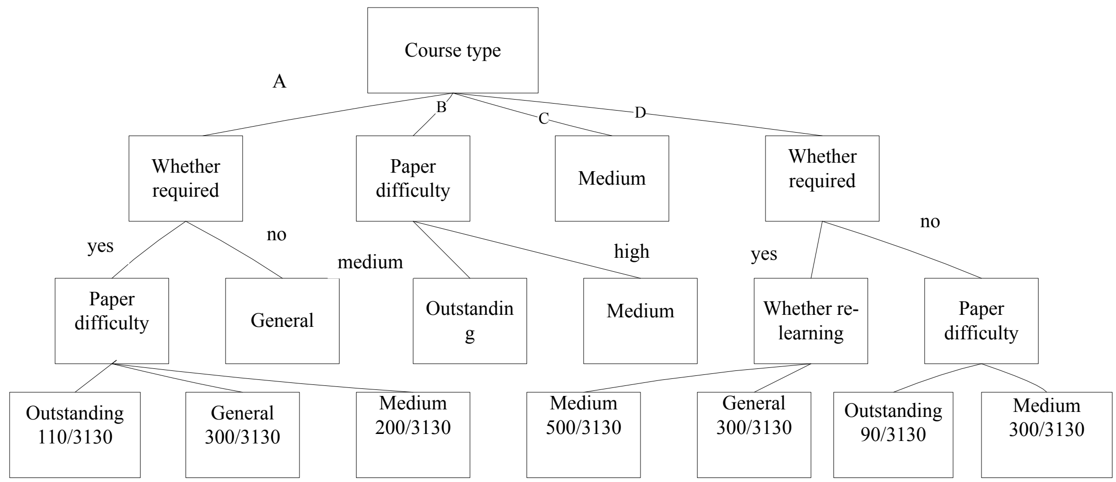

Fig. (2). Decision tree using improved algorithm.

4.3. Result of Improved Algorithm

The original algorithm lacks termination condition. There are only two records for a sub-tree to be classified which is shown in Table **3**.

All Gains calculated are 0.00, and GainMax=0.00 which does not conform to the recursive termination condition of the original algorithm in Table **3**. The tree obtained is not reasonable, so we adopt the improved algorithm, and decision tree using improved algorithm is shown in Fig. (**2**).

CONCLUSION

In this paper, we studied construction of a decision tree and attribute selection measure. Because the original algorithm lacks termination condition, we proposed an improved algorithm. We obtained course score of some universities for example, and we could find the latency factor which impacts the grades.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] W. Bor-tyng, S. Tian-Wei, L. Jung-Chin, T. Jian-Wei, and N. Masatake, “The study of soft computing on the field of sports education: Applying Grey S-P chart in sports writing assessment”, *International Journal of Digital Content Technology and its Applications*, vol. 5, no. 9, pp. 379-388, 2011.

[2] F. Huang, “Research of an algorithm for generating cost-sensitive decision tree based on attribute significance”, *International Journal of Digital Content Technology and its Applications*, vol. 6, no. 12, pp. 308-316, 2012.

[3] J. Guang-xian, “The research of decision tree learning algorithm in technology of data mining classification”, *Journal of Convergence Information Technology*. vol. 7, no. 10, pp. 216-223, 2012.

[4] H. Zhang, “Lazy decision tree method for distributed privacy preserving data mining”, *International Journal of Advancements in Computing Technology*, vol. 4, no. 14, pp. 458-465, 2012.

[5] H. Xin, “Assessment and analysis of hierarchical and progressive bilingual sports education based on neuro-fuzzy approach”, *Advances in Information Sciences and Service Sciences*, vol. 5, no. 1, pp. 269-276, 2013.

[6] C. Hong-chao, Z. Jin-ling, and D. Ya-qiong, “Application of mixed-weighted-association-rules-based data mining technology in college examination grades analysis”, *International Journal of Digital Content Technology and its Applications*, vol. 6, no. 10, pp. 336-344, 2012.

[7] M. HongYan, Y. Wang, and J. Zhou, “Decision rules extraction based on necessary and sufficient strength and classification algorithm”, *Advances in Information Sciences and Service Sciences*, vol. 4, no. 14, pp. 441-449, 2012.

[8] M. Bahrololum, E. Salahi, and M. Khaleghi, “An improved intrusion detection technique based on two strategies using decision tree and neural network”, *Journal of Convergence Information Technology*. vol. 4, no. 4, pp. 96-101, 2009.

[9] M. F. M. Mohsin, M. H. Abd Wahab, M. F. Zaiyadi, and C. F. Hibadullah, “An investigation into influence factor of student programming grade using association rule mining”, *Advances in Information Sciences and Service Sciences*, vol. 2, no. 2, pp. 19-27, 2010.

[10] S. M. Elayidom, S. M. Idikkula, and J. Alexander, “Design and performance analysis of data mining techniques based on decision trees and naive bayes classifier for”, *Journal of Convergence Information Technology*, vol. 6, no. 5, pp. 89-98, 2011.

[11] T. Bai, J. Ji, Z. Wang, and C. Zhou, "Application of a global categorical data clustering method in medical data analysis", *Advances in Information Sciences and Service Sciences*, vol. 4, no. 7, pp. 182-190, 2012.

[12] X. Xu, and C. Lou, "Applying decision tree algorithms in sports vocabulary test item selection", *International Journal of Advancements in Computing Technology*, vol. 4, no. 4, pp. 165-173, 2012.

[13] Z. Xin-hua, Z. Jin-ling, and L. Jiang-tao, "An education decision support system based on data mining technology", *International Journal of Digital Content Technology and its Applications*, vol. 6, no. 23, pp. 354-363, 2012.

[14] Y. Wang, and L. Zheng, "Endocrine hormones association rules mining based on improved apriori algorithm", *Journal of Convergence Information Technology*, vol. 7, no. 7, pp. 72-82, 2012.

[15] Z. Liu, and Y. XianFeng, "An application model of fuzzy clustering analysis and decision tree algorithms in building web mining", *International Journal of Digital Content Technology and its Applications,* vol. 6, no. 23, pp. 492-500, 2012.

Received: May 26, 2015 Revised: July 14, 2015 Accepted: August 10, 2015