Rule Induction Algorithms Applied in Educational Data Set

Walisson Carvalho[1], Cristiane Nobre[2] and Luis Zarate[2]
[1]*Centro Universitario Una, Belo Horizonte, Brazil*
[2]*Pontificia Universidade Catolica de Minas Gerais, Belo Horizonte, Brazil*

Keywords: Rule Induction Algorithms, Educational Data Mining, Classification, OneR, RIPPER, PART.

Abstract: This article presents the application of three induction rules algorithm: OneR, RIPPER and PART in an educational data set aiming to explain the main factors that lead students to be succeed or failure in online course. The dataset used to develop this article was extracted from the log of activities of engineering students that enrolled in a 20 weeks course of Algorithm offered online. The students used Learning Management System, Moodle. The dataset was preprocessed and then it was applied the algorithms into it. As result it was observed that students who begin earlier an assignment improve their probability of succeed.

1 INTRODUCTION

Due to the accessibility of Information Technology, over the last years, the process of teaching and learning had been changing of paradigm. One big change is related to the modality of teaching, day after day the traditional, face-to-face, modality is being replaced by online or blended teaching.

The hybrid teaching, as well as online, makes use of Learning Management System (LMS) or e-learning systems such as Moodle[1], Eliademy[2] and others. These LMS records data of the actions such as reading materials, interactions, chats, assignments and others from students and teachers as server logs. As a result, a big amount of data related to the behavior of the students, teachers are being produced.

Educational Data Mining (EDM) is an interdisciplinary research field of Data Mining that focus on developing methods and analyzing data that come from education sector. The main goal of EDM is to improve teaching and learning process. According to Romero and Ventura (2010), even though in EDM the dataset used are related to education, the methods and techniques are basically the same of the traditional Data Mining (DM).

YACEF (2009) classify EDM in five categories, they are: i) *prediction*; ii) *clustering*; iii) *Relationship mining*; iv) *Distillation of data for human judgment* and v) *Discovery with models*. The first three categories are like most of traditional DM methods, the other two are not so usual in traditional DM.

The first category is splitted in three sub-areas: i) *classification*; ii) *Regression* and iii) *Density estimation*. The focus of this article is in the second sub-area, classification.

Classification is a learning supervised technique that aim at categorizing data from prior information. According to Qin et al. (2009), even though there is a great number of algorithms dealing with classification, classifiers are still a big challenge in machine learning.

Two approaches are used in order to classify an object: black-box and rule-based classifiers. Black-box classifiers such as SVM, Neural Network, and other despite of having good performance, does not explain how an object were classified in one or another class. On the other hand, classifiers based on rules explicit the rules used during the process of classification. Thus, this approach makes it viable to create models that explain why an instance were designated to a specific class (Devasena and Sumathi, 2012).

In this scenario, the main goal of this article is to extract rules from an educational dataset from an online course offered to engineers' students in the face to face modality of a private university in Brazil. From the rules, it is expected to understand the students' behavior that lead them to succeed or fail in the course.

To develop this study the dataset was submitted to three algorithms of rule-base classifiers, they are OneR (Holte, 1993), RIPPER (Cohen, 1995) and

[1]https://moodle.com/
[2]https://eliademy.com/

Carvalho, W., Nobre, C. and Zarate, L.
Rule Induction Algorithms Applied in Educational Data Set.
DOI: 10.5220/0007832506910696
In *Proceedings of the 11th International Conference on Computer Supported Education (CSEDU 2019)*, pages 691-696
ISBN: 978-989-758-367-4

PART (Frank and Witten, 1998).

The remainder of the paper is structured as follow. Section 2 provides an overview of the main concepts covered in the paper. Section 2 describes the preprocessing and application of the algorithm in the data ser. In section 4, ours experiments and results is presented. Finally, section 5, presents some conclusions and future work.

2 RULE-BASED CLASSIFIERS

According to Aggarwal (2014), rule-based classifiers, even thought are related to decisions trees, are more general than decision trees.

Classifiers based on rules can be categorized according to its method as direct, focus of this article, or indirect. Direct algorithms, such as OneR, RIPPER, PART, CN2 and others, extract rules directly from the dataset. On the other hand, indirect algorithms identify rules of induction from classifiers such as Decision Tree, Neural Network and others.

Rule-based approach classifies objects based on a set of rules in the format of *if...then...*. According to Qin et al. (2009), each rule is expressed in the format $r_i : (condition) \rightarrow y_i$, such that $r_i : condition$ is known as rule antecedent and y_i is the rule consequent that represents the label of an instance.

Two important characteristics of rule-based Classifier are: mutually exclusive and exhaustive. Mutually exclusive occurs if no two rules are triggered by the same record, in other words, every object is covered by one rule. Exhaustive rule guarantee that all instances of the data set have at least one rule that cover it.

A rule can be assessed by its Accuracy and its Coverage. Given a rule *R*, we can say that *R* covers an instance *I* when the attributes of *I* observes the rule *R*. Thus, the Coverage of a rule is the number of instances that satisfy the antecedent condition. Coverage of a rule can be expressed as follow:

$$coverage(R) = \frac{n_{covers}}{|D|} \quad (1)$$

where n_{covers} represents the number of instances covered by R and $|D|$ the number of instances in data set.

Qin et al. (2009) define Accuracy of a rule as the fraction of instances that satisfy the antecedent and consequent of a rule, normalized by those satisfying the antecedent. Accuracy can be expressed from Equation 2:

$$accuracy(R) = \frac{n_{correct}}{n_{covers}} \quad (2)$$

where $n_{correct}$ is the number of instances correctly classified by R and n_{covers} instances covered by R.

Qin et al. (2009) highlight that both metrics, Coverage and Accuracy, should be high. One paradigm used by classifiers based on rules is the sequential covering paradigm, this paradigm learns a list of rules sequentially, one at a time, to cover the whole training mining rules with high accuracy and coverage first (Aggarwal, 2014).

2.1 OneR

The One Rule, OneR, algorithm was proposed by Holte (1993). It is one of the simplest algorithm for rule induction and despite its simplicity, it has good accuracy.

OneR creates a single rule for each attribute of data set and then picks up the rule with the minor errors rate. This algorithm learns an one level decision tree, it has four main steps as follow:

1. For each attribute, the algorithm creates one branch for each value of its domain;
2. for each value of its domain, the algorithm identifies the most frequent class;
3. OneR identify the error rate, in other words, the proportion of objects that do not belong to the majority class;
4. pick the attribute with the minor error rate;

Once that OneR uses all domain from each feature of the dataset, it covers all instances of the data set.

2.2 RIPPER

RIPPER, short for *Repeated Incremental Pruning to Produce Error Reduction*, was projected by Cohen (1995).

Rule learning in RIPPER is based on the strategy separate to conquer, this approach concentrates on one class at time disregarding what happens to the other classes. According to Cohen (1995), this algorithm can be described as follow:

1. Divide training set into growing and pruning sets;
2. grow a rule adding conditions;
3. prune rule;
4. go to 2), stopping criteria;
5. optimization of rules.

This algorithm is considered efficient and works well with imbalanced data as well as noisy data.

2.3 PART

PART, short for *Projective Adaptive Resonance Theory*, was projected by Frank and Witten (1998). This algorithm takes advantage of the construction of the tree based on C4.5 algorithm and from separate to conquer rule learning strategy of RIPPER.

According to Frank and Witten (1998), this algorithm has three main steps, they are:

1. Induce a rule from a partial tree;
2. remove all instances that are not covered by the rule;
3. induce new rules from the remaining instances.

Frank and Witten (1998) emphasize that PART, once that combines two paradigms, C4.5 and RIPPER, produces good results without global optimization.

3 METHODS

The dataset used in this article were extracted from Learning Management System Moodle - Modular Object-Oriented Dynamic Learning Environment. LMS recorded the actions of 229 students that enrolled an online course of Algorithm during the second semester of 2016. Initially the dataset had 75,948 instances and 42 features, each feature representing possibles actions performed by one of the 229 students and each instance representing an action taken by one student. Therefore an average of 331.65 (75,948/229) actions per student.

3.1 Pre-processing

After being extracted from LMS, the data was preprocessed. During this stage the data was transform such that each instance represents a student and each attribute a kind of action performed by the students. After the transformation, the dataset was with 229 instances and 42 attributes.

Once that not all students perform all actions, there was many missing data in the dataset. For example, only 8 students performed "chat talk" action, only 1 performed "course report log", and so on. Therefore it was discarded 23 features during the pre-processing stage due to missing data.

It was also added a new dichotomy's feature, Status. This is a variable with the domain 1, student approved, or 0, student failure. Status were computed according to the grade of the student. Students with final grade greater than or equal to 70, it was attributed 1 to Status, approved, otherwise it was attributed 0.

The final dataset is compound of 20 features, being one target, Status, and 19 predictive variables. Those explanatories variables describe actions performed by students such as visualizing tasks, submitting assignments, participating in chats and other actions that are part of the routine of the students. Table 1 describes each variable of the data set, access mean and standard deviation for each action performed by the 229 students at Moodle during the course.

The dataset analyzed was with 229 instance representing the students in which 135 were approved and 94 failure, this means that 41% failure in the course.

3.2 Rule Induction

After being preprocessed, three algorithms were applied to the dataset: OneR, RIPPER and PART. It was used R language and IDE StudioR version 1.0.136. OneR is based on the OneR library and for RIPPER and PART it was used RWeka library (Hornik et al., 2008)

All 20 attributes were discretized through the function *optbin* of the library OneR of the R Language. This function makes discretization of numeric data considering the target variable. Besides that, it is used logit regression to define the number of factors of the discretization.

The model used to train all algorithms was the simplest cross validation method, hold-out (Kuncheva, 2014). The dataset was divided in train, 80% of the data set (183 instances), and test, 20% (46 instances).

4 EXPERIMENTS AND RESULTS

The first algorithm applied into the educational dataset were OneR. Table 2 presents five attributes with the lowest error rate.

Most of the features presented in Table 2 are related to an assignment as described in table 1, only attribute *course.view* is not directly linked to an evaluation activity. Considering the feature *assign.submit*, attribute with minor error rate, OneR identified two rules, they are:

1. $if assign.submit = (-0.014, 5]$ then Status = 0
2. $if assign.submit = (5, 14]$ then Status = 1

According to the contingency table, Table 3, 68 instances are covered by rule number one and 115 by rule number two, implying in 37.3% and 67.2% coverage. Rule number one has an accuracy of 88.2% and number two, 85.2%. Considering both rules the accuracy is 86.34%, 158/183, as shown in Table 2.

Table 1: Actions performed by students.

Id	Action	Meaning	Mean	Standard Deviation
1	assign submit	Student is performing an evaluation activity: user has closed an evaluative activity, that was saved on Moodle to continue later and that was not yet sends for correction.	6.3	3.88
2	assign submit for grading	Student is finishing an evaluation activity: user has finished an evaluative task and sent it for correction.	3.9	2.38
3	assign view	Student is performing an evaluation activity: user has visualized the main page of evaluative task.	58.5	44.72
4	assign view all	Student has clicked on the link that lists all the evaluative tasks of a course.	1.0	2.24
5	assign view feedback	Student accessed the teacher feedback of an evaluative task.	04	1.14
6	assign view submit assignment form	User has viewed an evaluative task that was already submitted to be corrected by teacher. It is not permitted to edit anymore.	8.4	5.05
7	chat report	User has viewed the chat report of all previous conversation.	0.2	1.70
8	chat talk	Student has accessed the chat forum.	0.1	0.69
9	chat view	User has viewed the history of previous conversations on a chat.	0.4	1.15
10	course view	User has viewed the main page of the course to study or preparing to study some content.	127.4	98.18
11	forum view forum	User has viewed the forum main page.	1.7	3.99
12	page view	User has clicked on the page resource link, a custom html page that was displayed by the teacher. Student is studying or preparing to study some content.	24.8	24.9
13	quiz attempt	Student is performing an evaluation activity: user has started an evaluative task, however the results are not yet saved on the Moodle.	10.6	4.56
14	quiz close attempt	Student is performing an evaluation activity: user has finished an evaluative task that was saved on the Moodle.	10.4	4.63
15	quiz continue attempt	A questionnaire can be started and saved so that the student can continue to carry out the activity later. In this case the student is giving up for continuity to the questionnaire that moment.	14.8	7.82
16	quiz review	Student is performing an evaluation activity: user has edited an evaluative task that is saved on the Moodle but not yet finished.	4.6	7.38
17	quiz view	Student is performing or preparing to do an evaluation activity: user has viewed the main screen of an evaluative questionnaire.	35.9	21.17
18	quiz view summary	User has clicked a specific link to see if all questions of an evaluative questionnaire were answered.	12.0	5.69
19	url view	Student is studying or preparing to study some content: user has clicked on an url resource link and was directed to another page out of the Moodle system.	6.8	9.53
20	Status	Identify if student succeed or failure during the course.	NA	NA

Table 2: Top 5 attributes' accuracy.

Attribute	Accuracy (%)
assign.submit	86.34
assign.view	85.25
quiz.attempt	84.7
course.view	84.15
quiz.close.attempt	84.15

Combining the rules inferred from OneR and the description of the attributes presented in Table 1, it is possible to observe that one important factor of being succeed in the process, happens when the students save an assignment before submitting it for assessment. This means that students that reflect or review the activity before send it to be evaluated improves their probabilities of being approved.

Table 3: Contingency table of assign.submit.

Performance	(-0.014,5][3]	(5,14]	sum
Approved	8	98	106
Failure	60	17	77
Sum	68	115	183

The second algorithm applied into the dataset were RIPPER. As shown in confusion matrix, Table 4, the accuracy of RIPPER was 89.07%, 163 instances classified correctly.

Table 4: Confusion Matrix of RIPPER.

	Approved	Failure	Total
Approved	64	8	72
Failure	12	99	111
Sum	76	107	183

From RIPPER algorithm it was identified two rules:

1. *assign.submit* $= (-0.023, 5])$ then Status = 0 (67.0/7.0)
2. (*url.view* = (-0.057,1]) and (*assign.view* = (-0.345,39]) then Status = 0 (5.0/1.0) else Status = 1 (111.0/12.0)

It is important to observe that the antecedent of the first rule identified by RIPPER, *assign.submit*, is the same in OneR rule. The other two attributes that are antecedent in

The numbers in parentheses at the end of each rule represent the number of correctness and errors of the classifier. Therefore the first rule has an coverage of of 40.4%, $74/183$, and accuracy of 90.5%, $67/74$. The coverage of the second rule induced by RIPPER is 70.49% and accuracy of 89.9%.

The third algorithm applied into the dataset had an accuracy of 90.71%, 166 instances correctly classified as presented by confusion matrix, Table 5.

Table 5: Confusion Matrix of PART.

	Approved	Failure	Total
Approved	62	3	65
Failure	14	104	111
Sum	76	107	183

PART identified eight rules as shown bellow. As in the output of the RIPPER algorithm, the numbers in parentheses at the end of the rule represent number of correctness and errors of the classifier. For instance, rule number one covers 57, being 55 classified correct and 2 errors.

1. assign.submit = (-0.023,5] AND course.view = (0.408,97]: 0 (55.0/2.0)
2. quiz.view.summary = (9,27] AND assign.submit = (5,23] AND page.view = (13,123]: 1 (83.0/4.0)
3. quiz.view.summary = (9,27] AND quiz.close.attept = (9,16] AND page.view = (-0.123,13] AND chat.view.all = (-0.004,0] AND course.view = (97,594]: 1 (13.0/3.0)
4. quiz.review = (0,44] AND quiz.continue.attempt = (11,38] AND quiz.close.attept = (9,16] AND assign.submit = (5,23]: 1 (15.0/5.0)
5. quiz.review = (0,44] AND assign.submit.for.grading = (-0.007,4] AND assign.view.submit.assignment.form = (-0.021,6] AND assign.view.all = (-0.013,0]: 1 (5.0/2.0)
6. forum.view.forum = (-0.029,0] AND quiz.review = (0,44] AND assign.submit.for.grading = (-0.007,4]: 0 (4.0/1.0)
7. assign.submit = (-0.023,5]: 0 (4.0)
8. quiz.review = (-0.044,0]: 0 (2.0) : 1 (2.0)

Table 6 presents metrics accuracy and coverage of each rule extracted by PART algorithm.

Table 6: Accuracy and coverage of the rules inducted by PART.

Rule	Accuracy	Coverage
1	96.4%	31.7%
2	95.4%	47.5%
3	81.2%	0.8%
4	75%	10.9%
5	71.4%	3.8%
6	80%	2.7%
7	100%	2.1%
8	100%	2.1%

From the set of rules extracted from the algorithms it is possible to observe that 14 distinct attributes are antecedents. Besides, assign.submit is the only attribute that is present on all sets of rules. The cross-table shown in Table 7 represents all 14 attributes that take part in some rule and its respective algorithm.

Table 7: Antecedent's attributes in the set of rules.

Attribute	OneR	RIPPER	PART
assign.submit	X	X	X
url.view		X	
assign.view		X	
course.view			X
quiz.view.summary			X
page.view			X
quiz.close.attempt			X
chat.view.all			X
quiz.review			X
quiz.continue.attemp			X
assign.submit.for.grading			X
assign.view.submit. assignment.form			X
assign.view.all			X
forum.view.forum			X

From the analysis of the accuracy and coverage of the rules extracted from the three algorithms, it can be observed that the attributes that have the greatest influence on student performance are: *assign.submit*, *assign.view*, *quiz.view.summary*, *url.view*, *course.view* and *page.view*. The first three features are related to assignment and the other three are linked to the preparation of the students.

5 CONCLUSIONS

Considering that the main of this article is to identify the factors that lead a student to be succeed or failure in a teaching and learning process, it was possible to identify 14 attributes that are related to their performance.

It is important to stress the hole of the attribute *assign.submit* that is present in all three algorithms and has good metrics. Therefore, we can conclude that students who begin their activities earlier and reflect on them, increase their chances of being approved.

In spite of using a small data set, it is possible, according to the rules inferred through this work, understand the behavior of students in a Learning Management System and, from this comprehension, propose actions that can improve the process do teaching and learning.

As future work one can apply induction rules in feature selection identifying the relevant attributes to a target variable.

Another possible research is compare the results from this work with another techniques such as causal inference and Formal Concept Analysis.

REFERENCES

Aggarwal, C. C. (2014). *Data Classification: Algorithms and Applications*. Chapman & Hall/CRC, 1st edition.

Cohen, W. W. (1995). Fast effective rule induction. In *In Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann.

Devasena, C. L. and Sumathi, T. (2012). Effectiveness evaluation of rule based classifiers for the classification of iris data set i.

Frank, E. and Witten, I. H. (1998). Generating accurate rule sets without global optimization. pages 144–151. Morgan Kaufmann.

Holte, R. C. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11(1):63–90.

Hornik, K., Buchta, C., and Zeileis, A. (2008). Open-source machine learning: R meets weka. *Comput Stat*, 24(2):225–232.

Kuncheva, L. I. (2014). *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, New York, NY, USA.

Qin, B., Xia, Y., Prabhakar, S., and Tu, Y. (2009). A rule-based classification algorithm for uncertain data. In *2009 IEEE 25th International Conference on Data Engineering*, pages 1633–1640.

Romero, C. and Ventura, S. (2010). Educational data mining: A review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(6):601–618.

YACEF, R. S. B. K. (2009). The state of educational data mining in 2009: A review and future visions. *Journal of Educational Data Mining Publishing Home*, 1(1):3–17.